

머 리 말

국내 경기 변동상황을 종합적으로 측정하여 현실경제의 흐름을 한눈에 파악할 수 있는 경기종합지수(CI)를 통계청에서 작성하기 시작한 지도 어언 20년이 되어 갑니다. 그 동안 경기종합지수의 이론은 물론 지수작성기법도 발전을 거듭하여 이제는 우리 나라 실물경제의 흐름을 진단하는 데 중심적인 지표로서 자리잡은 것으로 평가할 수 있겠습니다.

한편 지방자치제의 정착과 함께 지역별로 경제 및 산업구조 등이 점차 특성화됨에 따라 지역 특성에 맞는 경제정책 수립은 물론 각 지역 기업의 생산, 투자 등 경영에 관한 의사결정의 기초자료의 필요성도 그 만큼 커지고 있습니다. 이와 함께 지역경제상황을 한눈에 볼 수 있는 지표에 대한 요구도 제기되어 왔습니다.

이와 같은 필요성에 부응하여 우리청은 그 동안 축적되어온 경기종합지수의 이론과 실무경험을 바탕으로 5개 시범지역을 선정하여 해당 시·도와 공동작업을 통해 금년에 시범지역의 「지역 경기종합지수」 작성방법을 제시하게 되었습니다. 이 자료는 그 동안 지표개발과정에서 활용된 지역경기종합지수의 이론과 작성요령 및 프로그램활용방법 등을 각 지방자치단체의 경제정책담당자가 쉽게 이해하여 실제 업무에 활용할 수 있도록 작성되었습니다. 이 자료가 지역 경기종합지수를 작성하는데 유익하게 활용되기를 기대합니다.

1999년 11월

산업동향과장 권 오 봉

업무참고자료

지역 경기종합지수 작성요령

1999. 11

통 계 청

총 목 차

1. 계절조정방법	3
2. 경기지표 해설	41
3. EViews 사용방법	65
4. 지역 경기종합지수 작성과정	209

계절조정방법

목 차

1. 계절조정의 개념 및 필요성	7
가. 계절조정의 개념	7
나. 계절조정의 필요성	8
2. 계절조정방법	16
가. 이동평균형 조정법	16
나. 모형조정법	20
3. X-12-ARIMA 방법	22
가. X-12-ARIMA의 개요	22
나. X-12-ARIMA의 특징	23
4. X-12-ARIMA 프로그램 사용법	30
가. INPUT과 OUTPUT	30
나. Flags(명령어)	34
다. 일반적인 입력사항	36
라. X-12-ARIMA 프로그램의 제약	38
마. 프로그램 예제	38

1. 계절조정의 개념 및 필요성

가. 계절조정의 개념

월별 또는 분기별 경제시계열에는 계절의 변화나 사회관습 등에 의해 매년 계절적으로 변동하는 패턴, 즉 계절성을 갖고 있는 것이 많다. 따라서 시계열의 추세적인 움직임이나 순환적인 움직임을 보려는 경우에는 이러한 계절성을 제거할 필요가 있고 이를 위한 방법으로 계절조정방법이 있다.

계절조정방법중 현재 가장 널리 이용되고 있는 것은 미국 상무성 센서스국이 개발한 「센서스국법 X-11」이다. X-11은 1965년에 발표된 이래 사용방법의 편리함으로 인해 일본을 비롯한 세계 각국의 통계기관에서 널리 이용되고 있으며, 실무적인 면에서는 대표적인 계절조정방법이라고 말할 수 있지만 기술적인 면에서 몇가지의 문제점이 지적되고 있다.

예를 들면, 매월 새로운 자료를 입력하면 추가한 자료로 계절조정을 처음부터 다시하여 계절조정계열이 과거로 소급하여 대폭 수정되는 경우가 적지 않다. 또한 백화점판매액, 신차등록대수처럼 월중의 요일구성(토·일요일의 일수 차이 등)의 영향을 강하게 받는 자료는 계절성을 적절하게 제거할 수 없는 경우가 있다. 더욱이 자료에 이상치가 섞여있는 경우에는 그 시기의 전후에 계절조정의 왜곡이 생기기 쉽다.

미국 상무성 센서스국에서는 X-11의 이러한 결점을 보완하기 위해 재검토를 한 결과, 이번에 센서스국법의 최신판으로 X-12-ARIMA를 개발하였다.

X-12-ARIMA의 특징은 실제로 계절조정을 하기 전에 데이터의 이상치나 요일구성의 요인을 추계, 제거하는 「사전조정」이라는 통계적 기법을 사용한다는 점에 있다. 뿐만 아니라 실제로 계절조정을 한 뒤에 계절성이 적절히 제거되었는지를 사후진단하는 기능도 가지고 있다.

미상무성에서는 대외적으로 공표하는 통계에 대하여 X-12-ARIMA의 적용을 이미 시작하였고 연방준비제도이사회(FRB)나, 노동성 노동통계국(BLS) 등 미국의 여러 기관도 현재 도입을 검토하고 있다. 통계청에서는 1999년부터 산업생산지수, 도소매판매액지수, 실업률 등의 계절조정에 X-12-ARIMA방법을 적용하고 있다.

나. 계절조정의 필요성

(1) 경제시계열의 계절성

월별이나 분기별 경제시계열에는 매년 계절적으로 반복되는 1년 주기의 변동이 포함되어 있는 경우가 많다. 이러한 변동을 「계절성」 또는 「계절변동성분」이라고 부른다. 계절변동(일년주기의 변동)이 생기는 배경에 대해 구체적으로 살펴보면 다음과 같은 자연적, 사회적 요인 등을 들 수 있다.

(가) 자연조건

기후나 기온 등 자연조건의 변화는 경제활동에 직접적인 영향을 미친다. 예를 들어 한랭지역에서는 동절기가 되면 농산물과 관련된 각종의 생산활동이 둔화된다.

(나) 월간영업일수의 차이

연말 또는 연초, 여름 휴가, 황금주말(golden week), 명절(설날, 추석) 등을 포함한 달이나 2월 등은 다른 달에 비해 영업(가동)일수가 적다.

(다) 경영환경에 관계되는 제도 및 습관

일반적으로 12월 결산기가 되면 경영성과상의 관점에서 매출액이나 수주액이 늘어나고 자금수요가 증가되는 경향이 뚜렷하다.

(라) 수요면에서의 영향

설날(추석) 명절, 세모(歲暮), 여름휴가 등에는 국민생활에 관계되는 관습에 따라 보너스가 많이 지급되면 개인소비가 급증하고 이에 대응하여 소비재의 생산, 매출, 재고가 계절적으로 변동한다.

(마) 공급면에서의 영향

원료, 자재, 동력 등의 계절적인 제약으로 이를 사용하는 생산활동이 변동한다. 예를 들면 전력다소비형 산업인 철강업 등의 경우 하절기에 생산을 줄인다.

(2) 계절조정의 필요성

경제시계열의 계절성은 때로는 유용한 정보를 제공해 준다. 예를 들면 국제 금융시장의 자금수급을 조절하는 중앙은행에 있어서 연말의 자금수요가 계절성에 의해 어느 정도 증가할 것인가는 시장조절을 하는데 있어서 중요한 정보이다.

그러나 경제시계열의 변동 가운데 계절성 이외의 부분에 관심을 가진 많은 분석가들에게 계절성은 단순히 잡음(noise)에 불과하다는 관점이 많다. 예를 들어 경제시계열을 사용하여 경기변동을 분석하려고 하는 경우 경기는 기본적으로 계절변동성분의 움직임과는 무관하기 때문에 계절변동성분을 제거한 시계열의 변동을 분석하는 것이 좋다. 이러한 경우에 사용되는 절차가 「계절조정」이다.

실제의 계절조정방법에 있어서 원계열(Y_t)은 계절변동성분(S_t) 외에 추세변동성분(T_t), 순환변동성분(C_t), 불규칙변동성분(I_t)으로 구성되어 있다고 하면 원계열(Y_t)과 이들 4성분과의 관계에 대해서는 다음중 어느 하나라고 통상 가정한다.

가법형 모형: $Y_t = T_t + C_t + S_t + I_t$

승법형 모형: $Y_t = T_t \cdot C_t \cdot S_t \cdot I_t$

계절조정을 위의 모형에 따라 해석한다면 원계열 Y_t 로부터 계절변동성분 S_t 를 제거하여 가법형 모형의 경우 $T_t + C_t + I_t$, 승법형 모형의 경우 $T_t \cdot C_t \cdot I_t$ 가 되는 계열(계절조정계열)을 만들어 내는 것에 불과하다. 각 변동성분에 대해서는 다음과 같이 정의된다.

(가) 추세변동(T_t)

경제성장 등에 따라 발생하는 장기적인 상승 또는 하강 경향을 나타내는 변동.

(나) 순환변동(C_t)

경기순환과정에서 확장과 수축기간을 상호 반복하는 주기적인 변동(2년을 초과하는 기간).

(다) 계절변동(S_t)

계절변화 등 1년을 주기로 하는 변동

(라) 불규칙변동(I_t)

위의 세가지 변동이외의 변동으로서 불규칙적으로 발생하는 돌발적인 요인이나 그밖에 원인불명의 교란요인에 의해 일어나는 변동(동맹파업이나 태풍, 지진 등의 자연재해도 여기에 포함), 설과 추석의 월간 또는 분기간 이동에 따른 명절변동, 월 또는 분기별 요일구성의 차이에 따른 요일구성변동 등이 있다.

여기서 명절이동(H_t)과 요일구성요인(D_t)은 계절조정하기전에 파악이 가능한 변동으로 사전조정요인이라 하고 이들 요인들을 원계열(Y_t) 구성요인에 포함하여 분해할 경우에 다음과 같다.

즉, $Y_t = T_t + C_t + S_t + I_t + H_t + D_t$ (가법형 모형)

$Y_t = T_t \cdot C_t \cdot S_t \cdot I_t \cdot H_t \cdot D_t$ (승법형 모형)

(3) 전년비에 의한 계절성의 제거

경제시계열의 분석에 있어서는 앞에서 설명한 계절조정계열 이외에 원계열의 전년동월(기)비를 계산하는 방법으로 계절성을 제거하는 간편법이 자주 사용되고 있다.

이 방법에서는 먼저 「원계열 Y_t (월별자료)는 승법형 모형을 따른다」고 가정할때,

전년동월비는 $\frac{Y_t}{Y_{t-12}} = \frac{TC_t}{TC_{t-12}} \cdot \frac{S_t}{S_{t-12}} \cdot \frac{I_t}{I_{t-12}}$ 로 표시된다.

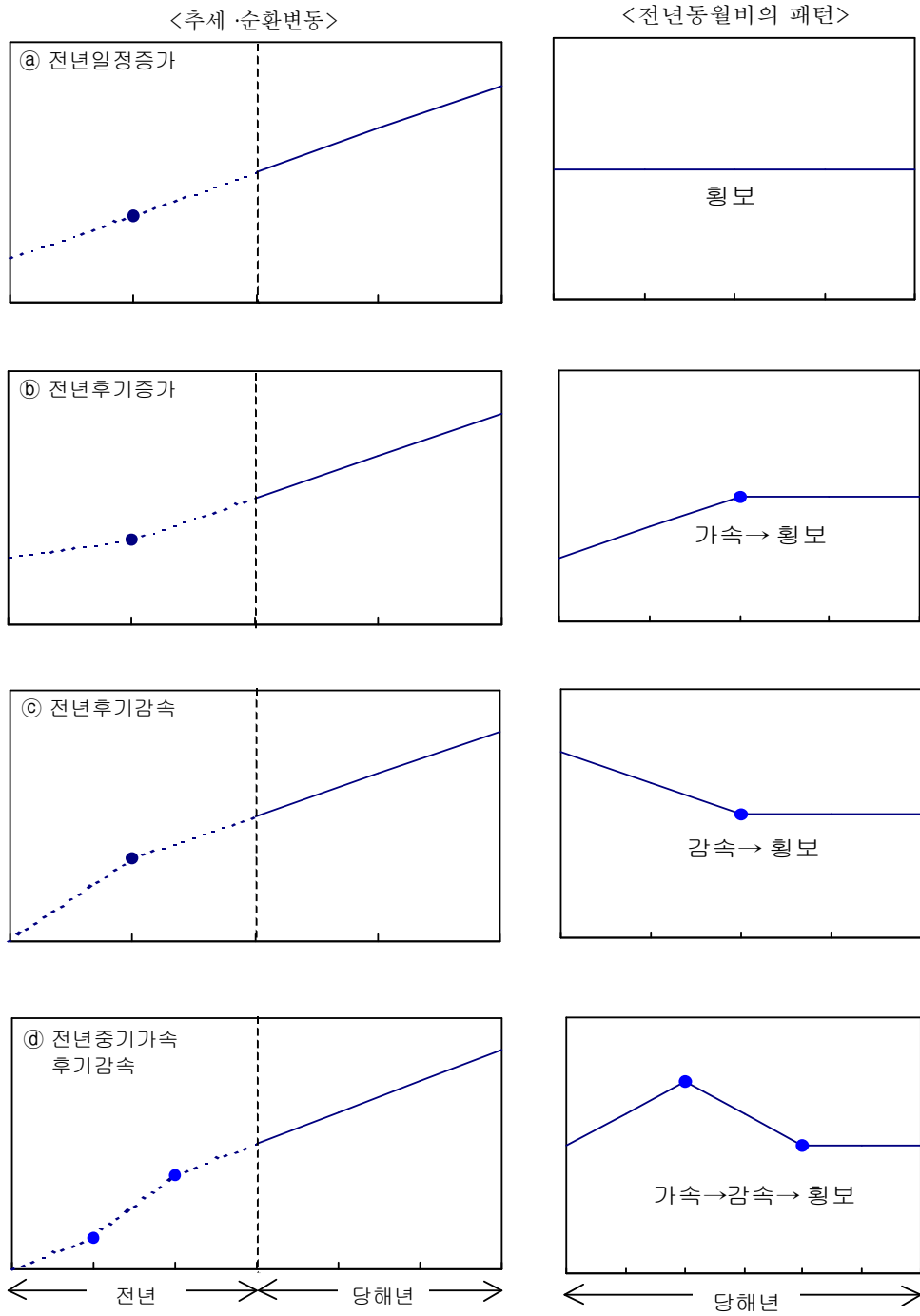
여기에서 「계절변동이 1년주기의 고정적인 패턴을 가진다」고 가정하면 ($S_t = S_{t-12}$), 전년동월비는 $\frac{Y_t}{Y_{t-12}} = \frac{TC_t}{TC_{t-12}} \cdot \frac{I_t}{I_{t-12}}$ 로 표시된다. 여기서는 계절변동성분이 완전히 제거되어 있다.

이와 같이 전년동월비는 계절성을 제거하기 위한 간편한 방법으로 어느 정도 유효하지만 몇 가지 문제점이 있다. 즉, 전년동월비를 대상으로 하는 시계열이 승법형이 아닌 가법형을 따를 경우나, 실제 경제시계열의 계절변동 패턴이 일정하지 않을 경우 ($S_t \neq S_{t-12}$)에는 적당하지 않다. 또한 시계열이 승법형을 따르며 계절변동 패턴이 일정하다고 하더라도 전년동월비는 경제분석상 다음과 같은 문제점이 내포되어 있다.

첫째는 전년도의 움직임이 교란요인이 될 가능성이 있다. 금년의 추세순환변동(TC_t)이 같은 움직임을 하고 있고, 전년이 각각 다른 움직임을 하고 있는 4가지 가상적인 예를 들어 전년동월비의 패턴을 비교하여 보면, (그림1)과 같이 금년의 추세순환변동이 동일(즉, 실제 경기추세가 동일)하더라도 전년의 추세순환변동의 패턴에 따라 전년동월비 움직임은 전혀 다르게 나타나는 것을 알 수 있다.

두번째는 전년동월비는 경기의 전환시점에 대해 잘못된 정보를 줄 가능성이 있다. 예를 들면, 5년주기로 순환하는 sine 곡선형의 추세순환변동(TC_t)과 그 전년동월비(TC_t/TC_{t-12}) 및 전월비(TC_t/TC_{t-1})를 비교하여 보면, (그림2)와 같이 전월비는 추세순환변동의 저점에서 정점까지는 「+」, 정점에서 저점까지는 「-」, 전환점(정점 또는 저점)에서는 「제로(0)」라고 하는 명확한 대응을 나타내고 있는 데 반해 전년동월비의 부호나 추이는 추세순환변동에 어떻게 대응하고 있는지 명확하지 않다. 또한 전년동월비는 전월비에 비해 후행적 성격을 지니고 있기 때문에 경기전환점의 판정을 지연시킬 위험도 있다.

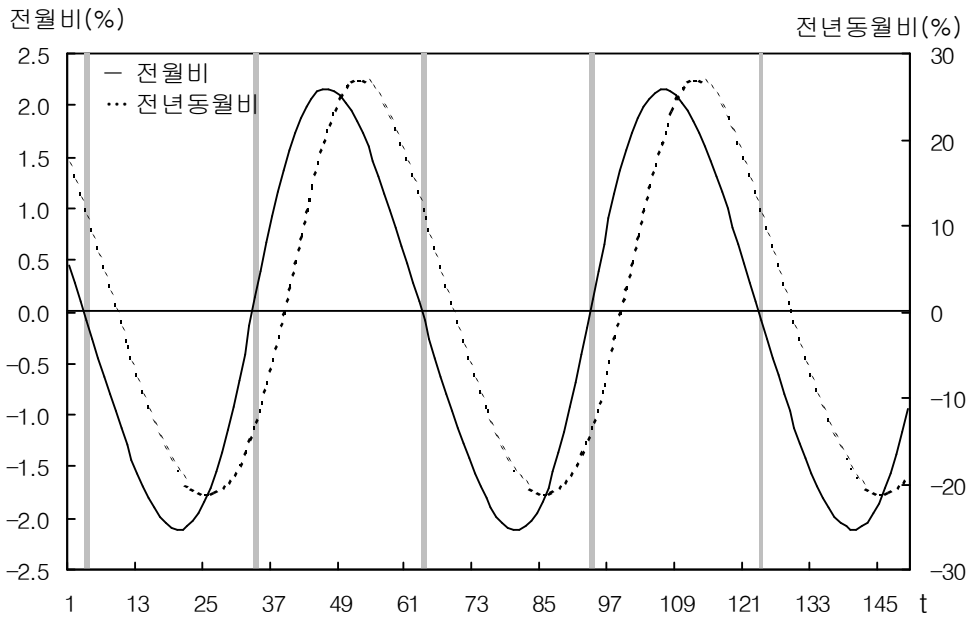
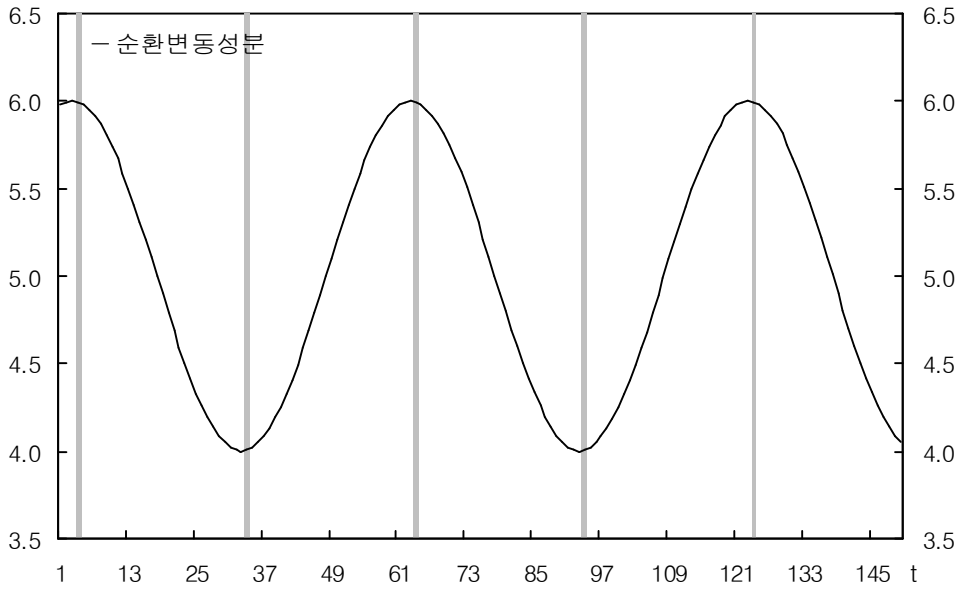
(그림1) 추세·순환변동과 전년동월비의 관계



출처:田原(1983)

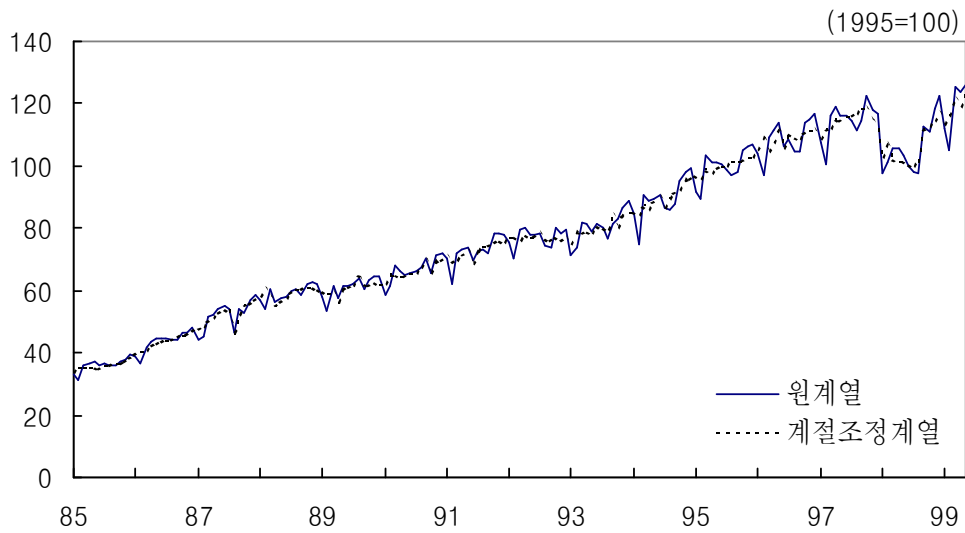
(그림 2) 원계열, 전월비, 전년동월비의 움직임

$$\sin[\pi(t+12)/30]+5$$



이와 같이 전년동월(기)비는 경기의 전환점을 파악함에 있어 잘못 판단하는 결과를 초래할 가능성이 있어 경제시계열의 계절성을 제거할 때에는 X11, X-11-ARIMA, X-12_ARIMA 등 계절조정방법을 사용하는 것이 바람직하다.

<산업생산지수 원계열과 계절조정계열>



2. 계절조정방법

계절조정방법은 「이동평균형 조정법」과 「모델형 조정법」으로 나눌 수 있다. 이동평균형 조정법은 『원계열을 이동평균함으로써 계절성을 평활화(smoothing)할 수 있다』라는 단순한 발상에 근거한 것으로, 계절조정방법의 원형이라 말할 수 있다. 다른 방법인 모델형 조정법은 경제시계열이 어떠한 확률모형으로부터 생성되었다고 명시적으로 가정하는 점에 특징이 있다. 즉, 데이터의 생성과정이 명백해지면 이를 기초로 하여 계절조정의 순서를 이론적으로 도출하는 것이 가능하고, 계절조정계열의 통계이론적인 성질이 분명해진다.

이 2가지 방법은 상당히 다른 방식에 근거하기 때문에 선형적으로 어느 것이 바람직한가는 말할 수 없지만 적어도 실무자들 사이에서는 사용방법의 편리함으로 이동평균형 조정법을 선호하고 있다.

최근에는 X-12-ARIMA와 같이 전적으로 이동평균형 조정법에 의존하기 보다는 모형조정법을 병행 사용함으로써 이동평균형 조정법의 단점을 보완하려는 경향이 나타나고 있다.

가. 이동평균형 조정법

이동평균형 조정법은 「1년치의 원계열을 이동평균하면 1년주기의 계절변동이 제거됨과 동시에 불규칙변동의 영향도 제거되어 결과적으로 추세순환변동만을 추출할 수 있다」는 방식에 근거한다. 이렇게 구한 추세순환변동을 원계열로부터 제거하면 계절변동과 불규칙변동이 남는다. 이 계절변동과 불규칙변동을 포함한 계열에 대

해서 같은 월만 추출하여 12개의 연도별 계열로 분류한 뒤 각각의 계열에 대해 연도간 이동평균하면 계절변동의 추계치를 얻을 수 있다.

구체적인 계산절차(가법형인 경우)는 다음과 같다.

- ① 원계열 Y_t 의 12개월 이동평균을 취함으로써 계절변동성분 S_t 와 불규칙변동성분 I_t 를 제거하여 잠정적인 추세순환변동성분 TC_t 를 산출한다.
- ② Y_t 로부터 잠정적인 TC_t 를 제거하여 잠정적인 S_t+I_t 를 산출한다 [$Y_t-TC_t=S_t+I_t$].
- ③ 잠정적인 S_t+I_t 를 동일 월의 연도별 계열로 분류하고 각각에 대해 수년치를 가중이동평균함으로써 I_t 를 제거하여 잠정적인 S_t 를 산출한다.
- ④ Y_t 로부터 잠정적인 S_t 를 제거하여 잠정적인 계절조정계열 TC_t+I_t 를 산출한다 [$Y_t-S_t=TC_t+I_t$].
- ⑤ 잠정적인 계절조정계열 TC_t+I_t 에 적당한 가중이동평균을 실시함으로써 I_t 를 제거하여 수정된 TC_t 를 산출한다.
- ⑥ 수정된 TC_t 를 사용하여 ②~④의 과정을 반복함으로써 최종적인 3가지 성분(TC_t , S_t , I_t)을 산출한다.

이동평균형 조정법의 대표적인 것은 미국 상무성이 개발한 센서스국법이다. 최초의 버전은 1957년에 센서스국법 X-1으로 발표되었다(X는 Experimental을, 1은 모델 일련번호를 나타내고 있음).

그 후 컴퓨터 처리능력의 향상과 함께 차례로 센서스법의 개량형이 발표되어 왔다. 그것의 기본적인 방식은 모든 경제시계열에 대하여 공통으로 적용할 수 있는 방법을 개발하려는 것으로, 1965년에 발표된 X-11까지의 개량형까지 적용되었다.

미국 상무성 센서스국이 개발한 이동평균형 조정법 X-11은 일변량계절조정방법의 대표적인 방법으로 발표한 이후 오늘날에도 세계 각국의 통계작성기관에서 이용되고 있고 일본에서는 일본은행이나 경제기획청, 총무청에서 공표된 통계의 계절조정에 이용되고 있다. 그러나 X-11은 이러한 실용면에서의 관록이 있는 반면 동 방법에 대한 비판도 적지 않다.

그 이유는 첫째로 X-11은 시계열의 각 변동성분에 대하여 명확한 확률모형을 가정하지 않고 단순히 이동평균을 반복하는 것에 지나지 않기 때문에 산출된 계절조정계열의 통계이론적인 특성이 불명확하다는 비판이 있다. 이러한 것은 이동평균항수 등에 대해 통계적으로 보아 객관성 있는 선택을 하는 것이 곤란하기 때문에 통계작성자의 주관이 깊이 들어갈 여지가 있음을 의미하고 있다.

두번째로는 계열말단부분에 있어 이동평균치가 그 시계열의 특성에 관계없이 일률적으로 후방이동평균으로 산출되고 있는 점이라든지, 구조변화(level-shift)나 이상치(outlier)의 처리가 적절하게 이루어지고 있지 못하다는 문제점으로 인해 경기판단에 있어 가장 중요한 최근시점의 계절조정계열이 추계기간의 변경 등에 의해 큰 폭으로 개정되는 경우가 많다. 이것은 계절조정계열의 「불안정성」이라고 불리어지며 경기판단이나 정책대응을 교란하고 나아가서는 계절조정에 대한 신뢰도를 약화시키는 중대한 문제가 발생한다.

상기 두번째의 문제 즉, 「불안정성」의 문제점을 계절말단부분 처리의 향상을 통해 개선시킬 목적으로 캐나다 통계청에서 개발한 X-11-ARIMA는 현재 캐나다 통계청외에 미국 노동성(실업통계)이나 FRB(통화공급량(money supply) 통계) 등에서 실제 사용되고 있다. 동 방법에서는 ARIMA모형을 이용하여 원계열의 예측치를 1~2년간 추정한 다음 원계열과 예측치를 연결한 계열에 대하여 X-11을 적용하는 방법을 취하고 있다. 즉, 계절말단부분을 일률적으로 후방이동평균으로 처리하지 않고 각각의 시계열이 갖는 정보를 가능한 한 이용하여 처리한다는 점이 특징이다. 그러나 X-11-ARIMA의 성과(performance)에 관한 실증분석에서는 계절조정계열의 안정성이 높아졌다고 하는 의견이 제시된 반면 원계열에 대해서는 그 안정성이 그다지 개선되지 않았다는 분석도 있다. 이것은 계절조정계열의 불안정성을 충분히 줄이기 위해서는 계절말단처리의 개선뿐만 아니라 이상치나 구조변화의 처리개선이 불가피한 것을 시사하고 있다.

이러한 문제를 해결하기 위해서 개발된 것이 X-12-ARIMA로 센서스국에 의해 1996년초 일반에게 공개되었다. 동 방법에서는 REGARIMA라고 불리는 사전조정부분에서 먼저 원계열을 ARIMA모형으로 표현되는 부분과 구조변화나 이상치로 설명되는 부분으로 분해한다(구조변화나 이상치를 추출하기 위한 연구로서 더미변수가 사용되고 있고 그 구체적인 설정 등에 관해서는 통계이론적으로 객관적인 선택이 가능). 이와 같이 원계열이 먼저 2개로 분해되고 다음으로 ARIMA모형으로 표현되는 부분(사전조정계열)에 대해 X-11에 의한 이동평균을 한다. 즉, X-12-ARIMA는 구조변화나 이상치를 사전에 제거한 계열에 대해 계절조정을 하는 방법이다. 미국의 경제시계열에 대해서 X-12-ARIMA를 적용한 실증분석에서는 동 방법이 X-11이나 X-11-ARIMA에 의한 방법보다도 안정성이 높은 계절조정계열을 생산해 내는 것이 확인되고 있다.

나. 모형조정법

모형조정법이란 현실의 경제시계열이 어떠한 확률모형으로부터 생성되고 있는가를 명확하게 가정하여 계절조정의 절차를 투명하게 함으로써 산출된 계절조정계열의 특성을 명백하게 나타내려고 하는 것이다. 이러한 방법에는 회귀분석방법(regression approach), 신호추출방법(signal extraction approach), Bayes형 조정법, 상태공간모형(state space model, SSM) 등이 있다.

회귀분석방법(regression approach)이란 각 변동성분에 대하여 적당한 함수를 설정하여(예를 들면 추세·순환변동에 대해서는 시간에 관한 다항식) 동 함수의 모수(parameter)를 최소자승법으로 추정하는 방법이다. 그러나 이 경우 함수형태의 선택이 다분히 선형적 또는 자의적이고 그 타당성은 주로 회귀식의 설명력에 의해 판단되기 때문에 계절조정계열의 신뢰성은 보장되지 않는다.

신호추출방법(signal extraction approach)은 원계열, 비계절변동성분(=추세·순환·불규칙변동성분), 계절변동성분의 각각에 대하여 ARIMA모형을 적용한 후 비계절변동성분의 ARIMA모형과 계절변동성분의 ARIMA모형의 합은 원계열의 ARIMA모형과 일치한다는 제약을 충족하는 비계절변동성분의 추계치(즉 계절조정계열)를 공학적 방법을 이용하여 도출하는 방법이다. 신호추출방법에 의한 계절조정에서는 ARIMA모형이라고 하는 신축적인 모델을 계열에 가정하고 있다는 점에서 우수하지만 그 잡음(noise)분포의 분산의 설정이 자의적이라고 하는 문제가 지적되고 있다.

따라서 회귀분석방법과 신호추출방법은 시계열의 배후에 존재하는 모형을 명확하게 가정하지 않고 각각의 함수선택이나 잡음(noise)분포의 분산의 결정에 관한 통계이론적인 객관성이 결여됨으

로써 분석자의 자의가 깊이 개입될 소지가 있다는 점에서 X-11과 동일한 문제점을 내포하고 있다.

Bayes형 조정법과 상태공간모형(SSM)은 이러한 문제를 개선한 방법인데 이들 방법의 최대 특징은 모형을 설정(specification)하는데 필요한 여러 가지 가정(시차변수의 차수라든지 비계절변동성분의 평활화 등)에 대해 AIC(Akaike Information Criterion)라는 통계량을 이용하여 통계이론적인 면에서 객관성이 있는 선택을 할 수 있도록 되어 있다는 점이다. 특히 상태공간모형(SSM)은 시계열의 각 변동성분을 확률차분방정식으로 표시하고 모형전체를 상태공간 표현으로 규정하여 각 변동성분의 상태나 잡음(noise)분포 등에 관한 가정에 범용성을 갖도록 한 계절조정방법이다. 예를 들면 잡음(noise)분포를 비가우스분포(non-Gaussian, 정규분포에 비하여 분포의 꼬리(아래)부분이 무거운 Quasi분포 등)로 가정함으로써 시계열의 이상치나 구조변화가 존재하는 경우에도 적절한 계절조정계열을 추계할 수 있도록 되어 있다.

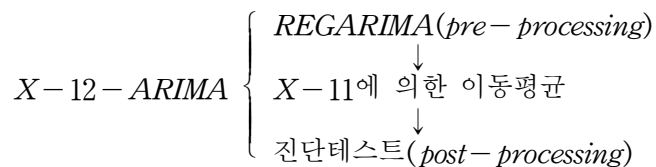
Bayes형 조정법이나 상태공간모형(SSM)은 통계이론적으로 본 객관성이란 관점에서 다른 모델형 조정법이나 X-11 등의 이동평균형 조정법에 비해 우수하다고 말할 수 있다. 그러나 「통계이론적으로 본 객관성」이란 ①데이타의 배후에 어떤 확률모형을 가정하고 있는가가 명확하고 ②시차다항식의 차수 등 확률모형의 설정(specification)에 필요한 선택적 정보의 선택이 통계이론에 입각한 기준에 의거하여 이루어져야 한다는 두가지 점을 의미하는데, ①에서 가정한 모형이 계절조정모형으로 최선이라는 것까지 의미하는 것은 아니다. 따라서 여기서 말하는 「통계이론적으로 본 객관성」을 갖춘 방법이라 해서 예를 들어 상태공간모형(SSM)이 X-12-ARIMA에 비하여 우월하다고 선택적으로 판정할 수 없다. 그래서 여러 가지 평가기준을 설계하고 각 계절조정방법에 대해 비교·검정할 필요가 있다.

3. X-12-ARIMA 방법

가. X-12-ARIMA의 개요

미 상무성에서는 1996년초 그 동안의 계절조정에 대한 연구성과를 바탕으로 이동평균법에 모형접근법을 가미한 새로운 계절조정방법 X-12-ARIMA를 제안하였다.

X-12-ARIMA의 작성순서는 ① REGARIMA에 의한 원계열의 사전조정부분 ② 종래의 X-11에 의한 이동평균부분 ③ 사후진단부분 등 3부분으로 구성되어 있다.



각 부분을 상세하게 살펴보면, 첫번째 부분은 REGARIMA를 이용하여 원계열을 ARIMA모형으로 표현할 수 있는 부분과 이상치나 요일변동 등의 회귀부분으로 분해한다. 그런후에 ARIMA모형에서 표현할 수 있는 부분과 그 ARIMA모형을 이용하여 추계한 예측치를 연결해 합쳐진 「사전조정계열」을 추계한다. 예를 들면, 이상치에 대해서는 REGARIMA에서 그 크기를 추계하여 이상치 발생시 원계열에서 그 추계치를 제거함으로써 이상치가 있어도 발생하지 않았던 것처럼 사전조정계열에 관해서 그 예측치를 포함하여 작성한다.

두번째 부분에서는 이렇게 해서 얻어진 사전조정계열에 대해서 종래의 X-11에 의한 계절조정을 실시한다. 여기에서는 데이터의 말단에서도 선행의 예측치를 사용한 중심이동평균이 가능하고 또한 이상치나 요일변동의 교란을 받지 않기 때문에 이동평균에 의해 계절변동을 적절하게 추출할 수 있다.

세번째 사후진단부분에서는 계절성이 완전히 제거되었는지를 통계 이론적인 관점에서 점검함과 동시에 계절조정계열의 안정성에 관한 진단을 수행한다. 진단의 결과에 따라 REGARIMA에 있어서의 모형화의 방법이나 X-11에 있어서의 이동평균항수 등을 변경한다.

나. X-12-ARIMA의 특징

기존에 광범위하게 이용되어온 X-11-ARIMA, X-11에 대한 주요 개선내용으로는 ① X-11의 이동평균 기능의 보강, ② 새로운 객관적 진단방법을 도입, ③ RegARIMA모형으로 사전조정 및 예측모형 설정 등이 있다.

(1) X-11법의 이동평균 기능 보강

X-12-ARIMA에서는 주기가 1년미만인 계절변동요인과 변동성이 큰 불규칙 변동을 제거할 수 있는 보다 장기의 이동평균을 이용함으로써 기존의 X-11보다 평활화의 효과가 큰 추세순환변동을 추출할 수 있다.

X-11에서는 3×3 , 3×5 , 3×9 의 3가지 필터가 제안되었으나 X-12-ARIMA는 이보다 긴 3×15 의 필터도 이용하고, X-12-ARIMA에서 이용되는 Henderson필터는 1년미만의 변동요인을 충분히 제거하기 때문에 주기가 1년이상의 변동요인만을 보존하는 특성을 갖도록 설계되었다. X-11에서 이용되는 Henderson필터는 주파수 0 외에도 0.1 정도의 주파수의 순환변동을 제대로 제거하지 못한다.

시계열에 대해 중심화 이동평균 적용시 양극단의 경우 자료부족

으로 중심화 필터 대신 전방 또는 후방 필터를 이용하게 되는데 이때 X-12-ARIMA에서는 Musgrave의 방법을 이용하는데, 이 방법은 평방개정오차(mean square revision error)를 최소화하는 필터를 찾는 방법이다.

(2) 새로운 진단방법의 도입

계절요인은 관측불가능하기 때문에 계절조정的评价기준으로 조정계열의 안정성, 직교성 등이 이용되고 있는데, 안정성이란 계절조정 대상기간변경 또는 신규자료추가에 따라 계절조정계열의 변화폭이 크지 않아야 하고, 직교성이란 계절요인이 원계열에서 완벽히 제거되었는지 여부를 판단하는 방법이다.

안정성 검정시 X-11 ARIMA에서는 M1-M11과 이를 가중평균한 Q통계량으로 계절성이 적절히 제거되었는지를 평가하는데, 3개월간격의 원계열, 추세순환 제거한 원계열에 대한 불규칙요인의 상대적 기여도, 불규칙요인에 대한 추세순환요인의 전월비 등 계절조정의 각 요인에 대한 검정결과를 제공한다.

Lothian · Morry(1978)는 캐나다 421개 계열을 바탕으로 M1-M11에 대한 가중치를 산출하여 Q통계량을 작성하였는데, M1-M11, Q통계량은 이론적인 기반이 취약한 간접적인 평가이다.

X-12-ARIMA에서는 Q통계량보다 객관적인 Sliding span방법과 Revision history방법을 제공하고 있고, Sliding span 분석은 계절조정산출기간을 k개의 구간(span)으로 나눈 후 동일시점의 계절변동 성분, 계절조정계열의 전월(기)비 등이 얼마나 안정적으로 변하는지

분석하는 방법으로 그 변화가 적은 경우 계절조정방법은 안정적이라고 할 수 있다. 예를 들면 계절조정산출기간(1982년 1월(분기)...1998년 12월(분기))을 4개의 구간으로 나누고 k 번째 구간 i 년 j 월(분기)의 계절요소를 $S_{ij}(k)$ 로 정의함. 각 i 년 j 월(분기)에 대해 MPD(maximum percentage difference)를 구할 수 있다.

$$SP_{ij} = \frac{\text{Max}|S_{ij}(k)| - \text{Min}|S_{ij}(k)|}{\text{Min}|S_{ij}(k)|}$$

또한 위와 같은 방법으로 계절조정계열의 전월(기)비에 대해 MPD(MMP_{ij})를 구할 수 있는데, SP_{ij}와 MMP_{ij}가 3%이상인 월(분기) 비율(각각 S%, M-M%)을 이용하여 안정성을 분석할 수 있다.

Sliding span 분석을 통하여 추가적으로 직접법과 간접법의 우수성과, 이동평균 필터 길이를 선정하기 위한 판단에 이용된다.

여기서 직접법과 간접법이란 산업생산지수(총), GDP와 같이 여러 항목의 합으로 구성된 경제변수에 대한 계절조정시 하부항목(산업생산지수 소분류, GDP계열 등)을 직접 계절조정하는 방법을 직접법, 각 구성항목을 계절조정한 후 합하는 방법을 간접법이라 한다.

X-12-ARIMA에서는 안정성에 대한 또다른 진단방법으로 한 시점에서 초기의 조정계열과 최종 조정계열의 차이에 바탕을 둔 Revision history 분석이 있는데, 이를 계산하기 위하여 다음의 MAPR(Mean Absolute Percentage Revision)을 이용한다. MAPR이 작을수록 안정성이 높은 계절조정법으로 평가 한다.

$$MAPR = \frac{100}{T - t_0 + 1} \sum_{t=t_0}^T \left| \frac{A_{t/T} - A_{t/t}}{A_{t/t}} \right|$$

여기서 $A_{t/T}$ 는 T기까지의 원계열($Y_1 \dots Y_T$)를 이용하여 구한 t시점의 계절조정계열이고, $A_{t/t}$ 는 t기까지의 원계열($Y_1 \dots Y_t$)를 이용하여 구한 t시점의 계절조정계열로 동반조정(concurrent adjustment) 계열이다.

직교성 검정시 X-12-ARIMA에서는 계절조정 후 원계열과 계절조정계열에 대해 스펙트럼을 추정하여 해당 주기에 대한 유의한 변동이 존재하는지를 평가하는데, 계절조정이 적절하게 된 경우 원계열의 계절 및 요일구성주기 스펙트럼은 유의하게 나타난 반면 계절조정계열의 계절 및 요일구성주기 스펙트럼은 유의하지 않게 나타난다.

여기서 계절빈도는 월별자료의 경우 $k/12$, $k=1, \dots, 6$ 이며 분기자료의 경우 $1/4$, $1/2$ 이며 요일변동의 빈도는 .348 주기/월, .432주기/분기이고, 스펙트럼은 시계열의 주기적 순환변동을 파악하는 기법으로 X-12-ARIMA에서는 최근의 8년에 대해서 AR추정치가 사용된다.

(3) RegARIMA모형과 사전조정

RegARIMA모형은 구조변화, 특이항 및 요일변동 등 더미변수를 계절ARIMA모형에 결합한 시계열모형이다.

시계열 y_t 에 대해 r 개의 더미변수 x_{it} 와 계절ARIMA모형 $(p,d,q)(P,D,Q)_s$ 이 결합된 RegARIMA모형은 아래와 같이 구성된다.

$$\Phi_p(B)\Phi_p(B^s)(1-B)^d(1-B^s)^D(y_t - \sum_{i=1}^r \beta_i x_{it}) = \Theta_q(B)\Theta_q(B^s)a_t$$

여기서 B 는 후방연산자($y_{t-k}=B^k y_t$)이며 $\phi_p(z), \Phi_p(z), \theta_q(z), \Theta_q(z)$ 는 각각 상수항이 1인 다항식이고, 또한 a_t 는 $i.i.d. N(0, \sigma^2)$ 분포를 따른다고 가정한다.

RegARIMA모형의 모수들은 반복일반화최소자승법(IGLS)에 의해 추정할 수 있는데, AR 및 MA모수를 고정하여 더미변수의 계수(β_i)를 먼저 추정한 후 이를 이용하여 다시 AR, MA 그리고 더미변수의 계수를 최우법(maximum likelihood)으로 추정한다.

사전조정에 이용되는 더미변수로는 명절, 공휴일수, 요일구성변동, 특이항 및 구조변화 등을 고려할 수 있다.

(가) RegARIMA모형을 이용한 명절효과의 조정

우리나라의 태음력에 기초하는 명절인 설과 추석은 각각 1, 2월과 9, 10월에 걸쳐서 나타나기 때문에 월 또는 분기로 작성되는 시계열의 경우 계절조정시 이들 요인으로 인하여 왜곡현상이 발생할 수 있다.

1980년부터 1999년까지 우리나라의 설 및 추석일자를 살펴보면 설은 1월에 6회, 2월에 14회, 추석은 9월에 4회, 10월에 16회 나타났다.

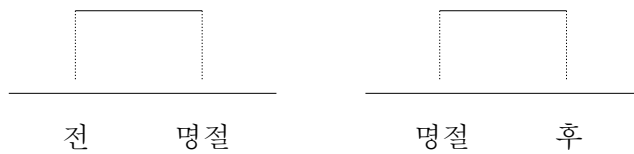
<우리나라의 설, 추석일자>

년도	설	추석
1980	2.16	9.23
1981	2. 5	9.12
1982	1.25	10. 1
1983	2.13	9.21
1984	2. 2	9.10
1985	2.20	9.29
1986	2. 9	9.18
1987	1.29	10. 7
1988	2.18	9.25
1989	2. 6	9.14
1990	1.27	10. 3
1991	2.15	9.22
1992	2. 4	9.11
1993	1.23	9.30
1994	2.10	9.20
1995	1.31	9. 9
1996	2.19	9.27
1997	2. 8	9.16
1998	1.28	10. 5
1999	2.16	9.24
2000	2. 5	9.12

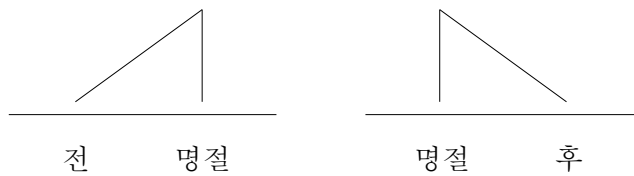
일반적으로 설 또는 추석 등 명절은 공휴일이 3일간 지속되기 때문에 생산 관련지표가 감소하는 한편 일시적인 자금수용 증대로 소비를 증가시키는 등 경제시계열의 교란요인으로 작용한다.

각종 경제변수의 기초적 변화를 파악하기 위하여 계절조정시 명절 효과를 제거할 필요가 있으며 이러한 명절효과를 제거하는 방법으로 미국 상무성의 Bell·Hillmer방법과 캐나다 통계청의 Dagum방법이 사용되는 데, 통계청에서는 Dagum방법을 사용하고 있다.

Bell·Hillmer(1983)는 부활절 이전과 지속기간에 동일한 가중치를 부여하는 방법이다



Dagum(1988)은 명절의 파급기간(k)동안 명절효과가 점차 감소하는 가중치(i/k , $k=1,2,3\cdots k$)를 부여하는 방법이다



현재 통계청에서 사용하는 방법은 명절 전의 경제시계열의 행태가 다른 점을 고려하여 명절 전에 대하여 더미변수를 이용하여 명절효과를 추정하고 있다

예) 2월 4일에 설이 있었다면 1월 명절효과는 $(1+2+3+4+5+6)/55=21/55$
 이고 2월 명절효과는 $(7+8+9+10)/55=34/55$ 이다

4. X-12-ARIMA 프로그램 사용법

X-12-ARIMA 프로그램을 실행하기 위해서는 입력파일 이름을 filename.spc라는 파일로 만들어야 되며, 실행 결과는 filename.out, 실행 중 에러는 filename.err 이름으로 생성된다

가. INPUT과 OUTPUT

X-12-ARIMA를 실행하기 위해서는 파일 확장명이 *.spc인 파일을 만들어야 한다. 실행 파일에는 사용자정의 회귀변수 및 사전조정요인, 자동모형 선택과정 등 여러 가지 옵션들로 구성할 수 있다. 실행결과는 *filename.out*이 생성된다, 실행 결과 중에 분석에 필요한 부분별로 *save*할 수 있다. 예를 들면 계절요인만 산출할 경우에 확장명이 *.D10인 파일을 만들 수 있다. 산출 결과의 파일명을 별도로 지정하지 않으면 *input filename*에 확장명이 *.out인 파일로 저장되고, *input filename*과 다른 *output filename*을 지정할 때는 다음과 같이 실행하면 된다.

```
c:\x12a\x12a path\input-filename path\output-filename
```

여기서 *input-filename*은 *filename.spc*이고 *output-filename*은 *filename.out*가 생성된다.

(예1) 입력파일은 *test.spc*이며 실행 결과를 *test.out*로 출력하고자 할 때 *c:\x12a\x12a test*로 지정하면 된다.

(예2) 입력파일이 *test.spc*이나 실행 결과를 *output.out*로 출력하고자 할 때 *c:\x12a\x12a test output.out*로 지정하면 된다

하나 이상의 시계열을 실행시키고자 할 경우와 여러개의 시계열을 동시에 실행하고자 할 경우는 X-12-ARIMA를 다음과 같이 이용한다.

- ① *multi-spc mode*: 여러개의 시계열에 대하여 각각의 입력명령문을 작성
- ② *single-spc mode*: 여러개의 시계열을 한 개의 입력 명령문으로 처리

②번 명령을 실행하기 위해서는 X-12-ARIMA를 실행하기 전에 *metafile*을 만들어야 한다. 그리고 *multi-spc mode*를 실행하고자 할 때 *metafile* 안에 X-12-ARIMA에서 실행하고자하는 입력명령 파일이름이 *TEXT*형식으로 입력되어야 한다.

*multi-spc mode*의 경우, *xuu1.spc xuu2.spc xuu3.spc*의 개별 명령문을 작성한 후 일괄처리를 위해서는 “*.mta”라는 확장자(extension)를 갖는 다음과 같은 *metafile*을 작성해야 한다.

*metafile.mta*의 입력 형태: · *xuu1*
 · *xuu2*
 · *xuu3*

이 때 X-12-ARIMA의 실행방법은 다음과 같다.

`c:\x12a\x12a -m metafile`

실행 결과는 *xuu1.out, xuu2.out, xuu3.out*으로 각각 출력된다. 한편 *x12a.sum*에는 실행된 *metafile*의 모든 내용이 저장된다.

*single-spc mode*의 경우, 하나의 명령문에 의해서 여러개의 시계열을 계절조정하기 위해서는 먼저, “*.dat”라는 확장자를 갖는 *data*

에 관한 *metafile*을 생성시켜야 한다. *data metafile*의 한 줄에는 주어진 시계열에 대한 파일명과 출력 파일명을 쓸 수 있다. 출력 파일명을 지정하지 않으면 입력 *data* 파일명과 경로명이 출력 파일명이 된다. *data* 파일이 현재의 디렉토리에 있지 않다면 반드시 경로명을 써야 한다. 개별 *data metafile*에 나타나는 순서대로 실행된다. *data metafile*에 서술된 모든 *data* 파일은 같은 형태(*format*)로 입력되어 있어야 한다.

xuu1, *xuu2*, *xuu3*라는 *data*를 계절조정한다면 *data metafile*의 구성은 다음과 같다.

*metafile.dat*의 입력형태 :

· *xuu1.dat* · *xuu2.dat* · *xuu3.dat*

*data metafile*을 갖는 경우의 X-12-ARIMA의 실행방법은 다음과 같다.

```
c:\x12a\x12a specfile -d metafile
```

여기서 ‘-d’는 *data metafile*이 있음을 나타내며, ‘*specfile*’은 계절조정을 수행하기 위한 입력명령 파일인 *specfile.spc*이며 *metafile*은 *data*들을 지정한 *data metafile*이다.

(예3) [*multi-spc mode*] 1980년 1월부터 1998년 12월까지의 광업, 제조업, 전기·가스·수도업의 생산지수를 계절조정 하고자 한다. 자료 파일이름은 각각 *PC.dat*, *PD.dat*, *PE.dat*, 입력명령문을 *PC.spc*, *PD.spc*, *PE.spc*이라고 할 때 실행순서는 다음과 같다.

- ① 개별 시계열에 대한 계절조정 입력명령문 작성
(*PC.spc*)

```

series{ title="Mineral Production Index"
        name="MPI"
        start=1980.01
        period=12
        span=(1980.01, 1998.12)
        file="PC.dat"
        format="(8x,12f8.1)"
      }

```

② 개별 시계열의 입력명령문에 대한 *metafile* 구성(*test.mta*)

```

PC.spc    PD.spc    PE.spc

```

③ DOS mode에서 실행한다.

```

c:\x12a\x12a -m test

```

④ 실행의 결과는 *PC.out*, *PD.out*, *PE.out*로 생성된다.

(예4) [*single-spc mode*] 1980년 1월부터 1998년 12월까지의 광업, 제조업, 전기·가스·수도업의 생산지수를 계절조정 하고자 한다. 자료는 모두 (8x,12f8.1) 형식으로 입력되어 있다. 자료 파일이름은 각각 *PC.dat*, *PD.dat*, *PE.dat*이며 이 때 결과는 *output*이라는 경로에 *out1*, *out2*, *out3*로 출력하고자 할 경우 다음과 같이 실행한다.

① *data metafile* 구성(*meta.dat*)

```

PC.dat c:\output\out1

```

```

PD.dat c:\output\out2

```

```

PE.dat c:\output\out3

```

② 입력 명령문 작성(*pcom.spc*) 한다.

```

series{ title="Industrial Production Index"
        start=1980.01
        period=12

```



```

span=(198001, 1996.12)
format="(8x,12f8.1)"
}

```

③ DOS mode에서 실행한다.

```
c:\x12a\x12a pcom -d meta
```

나. Flags(명령어)

프로그램을 실행 시키고자할 때 DOS mode에서 사용되는 명령어는 다음과 같이 지정한다

```
c:\x12a\x12a arg1 arg2 ... agrN
```

여기서 *arg1 arg2 ... agrN*은 flag이거나 파일이름이다.

<X-12-ARIMA에서 사용되는 flag>

flag	내 용
-i filename	입력파일의 파일이름
-o filename	프로그램이 실행되는 동안에 생성되는 모든 출력파일의 파일 이름
-m filename	입력 metafile의 파일이름
-d filename	data metafile의 파일이름
-n	입력 명령문에서 요구한 표만 인쇄
-w	출력파일의 한 줄이 132자로 인쇄
-p	출력파일의 page 표시를 없앴
-s	계절조정의 진단결과를 파일속에 저장
-c	복합 계절조정의 경우 개별 요인의 합을 계산
-v	입력 명령문의 error check

다음은 입력파일이 *test1*이고 출력파일은 *test2*인 경우 X-12-ARIMA의 실행방법이다.

```
c:\x12a\x12a test1 test2
```

```
c:\x12a\x12a -i test1 -o test2
```

```
c:\x12a\x12a -o test2 -i test1
```

flag(명령어)의 내용을 좀 더 상술하면 다음과 같다.

- **-n** : X-12-ARIMA 결과의 출력시 표가 많이 있기 때문에 출력 파일에 나타나는 표 번호를 지정하기 위해 사용한다.
- **-s** : 계절조정의 가장 중요한 진단결과를 출력파일 "**.out*"과 분리하여 "**.xdg*"파일로 저장한다.

(예5) 출력결과는 *test.out*이고 *test.xdg*에 저장되며 *test.xdg*에는 계절조정의 주요한 진단결과가 저장된다.

```
c:\x12a\x12a test -s
```

(예6) 출력결과는 *testout.out*이고 *testout.xdg*에 저장된다.

```
c:\x12a\x12a test -s -o testout
```

- **-c** : 입력 *metafile* (*-m*)로 실행되는 합성 계절조정시 사용한다. 합성 계절조정에 있어서 X-12-ARIMA는 개별 시계열요인을 계절조정한다. 입력 명령문에서는 개별 시계열에 대한 내용이 필요하다. 이 옵션을 사용할 때 개별 시계열에 대한 입력 명령문의 계절조정과 모형옵션은 무시되며, 개별 시계열은 시계열을 구성하기 위해서만 사용된다. 이 옵션은 시계열을 구성하기 위한 REGARIMA 모형을 사용할 때 유용하다.

다. 일반적인 입력사항

자료의 값은 항상 같은 형태(*format*)로 입력하며 월(분기)별 자료의 날짜는 “연도.월(분기)”로 표시한다. 연도를 표시할 때 “99”는 1999년을 의미한다. 월은 1~12의 정수 또는 3자의 영문을 사용할 수 있다. (jan, feb, mar, ...) 따라서 1998.12, 98.12, 98.dec는 모두 같은 의미이다. 날짜는 시계열의 시작 시점과 분석하고자 하는 시계열의 기간 및 특이치에 대한 회귀변수를 정의할 때 사용한다. 예를 들어 1982년 4월의 가법 특이치와 1998년 1월을 시작으로 하는 수준변동에 대한 회귀변수는 다음과 같이 사용한다.

```
regression { variables=(ao1982.apr ls1998.jan) }
```

“ ” 안에 있는 문자를 제외한 문자, 숫자, 공란, tab, =, ., {}, (), [], +, -, # 등을 사용할 수 있다. 또한 “ ” 안에 !, %, ‘, *, /, ;, ;, <>, ?, @, \, _, |, ~, ^ 등을 사용할 수 있다.

{}, (), []는 다른 기능을 갖고 있다. {}는 명령문에서 옵션을 포함하며, ()는 하나의 옵션에 여러 개의 값을 포함할 때 이용한다. []는 다음과 같은 부활절 회귀계수와 같은 특별한 옵션이나 결측 시차를 갖는 ARIMA 모형의 시차를 사용할 때 이용한다.

```
regression { variables=(td easter[14]) }
```

```
arima { model=(0 1 [1,3])}
```

주석(comment)은 # 다음에 쓰고, 주석이 한 줄 이상일 때는 각 줄의 첫 번째 열에 #을 해 주어야 한다. 시계열 이름을 나타내는 *title*은 한 줄에 79자 이내에 ‘*title*’ 또는 “*title*”로 기술한다.

명령화일의 한 줄(line)은 132자로 제한되어 있다. 명령화일의 한 줄이 132자를 넘는 경우 자료입력시 자료가 잘려지는 형태로 입력

이 된다. 자료가 file 옵션에 의해 별도의 파일에 의해서 읽혀지는 경우 한 줄이 132자 이내라는 제한은 적용되지 않는다. 이 경우 FORTRAN 자료 입력 길이의 제한에 따른다.

여러 개의 옵션값은 다음과 같이 괄호에 의해서 함께 묶여져야 한다.

```
variables=(td seasonal const)
```

하나의 옵션에 하나의 값 또는 여러개의 값을 갖고 있으나 하나의 값만이 주어진 경우에는 괄호를 다음과 같이 선택사항으로 사용할 수 있다.

```
variables=td, variables=(td)
start=1998.4, start=(1998.4)
```

숫자는 다음과 같이 지수(exponential) 형태 또는 free format 형태로 지정할 수 있다.

```
400, 400.0, 400., 4.e+2
```

정수인 경우에는 2.0 혹은 2.e+0은 사용할 수 없으며 정수 표현양식 2와 같이 사용해야 한다.

명령문 중 series명령은 항상 맨 처음에 있어야 하며, 명령문 내에 있는 옵션의 순서에는 제약이 없다.

blank, space, tab, blank line을 분리 연산자로 사용할 수 있다. 여러 개의 옵션값이 있는 경우, 콤마에 의해서 각 옵션값을 다음과 같이 분리할 수 있다.

```
data=(0, 1, 2, 3, 4, 5)
```

콤마는 기본값(default)이 대체되는 결측 옵션값을 나타내기도 한다. span=(1967.4,)의 경우 1967.4 다음의 콤마는 두 번째 기간(span)이 결측 옵션값으로 자료의 마지막 관측치를 취하게 된다.

라. X-12-ARIMA 프로그램의 제약

X-12-ARIMA에서는 시계열의 최대 길이, 모형의 최대 회귀변수의 개수 등에 대한 제약을 두고 있다. 그러나 이러한 제약은 FORTRAN source 파일의 model.prm과 srslen.prm을 수정한 후 다시 실행하면 해결할 수 있다. 현재 X-12-ARIMA에서는 최대 입력시계열 길이는 600이다. 전방과 후방을 예측하여 연장한 최대 시계열 길이는 60년이다. 최대 전방 예측길이는 60, 모형에 사용되는 최대 회귀변수는 48개, 사용자가 정의하는 최대 회귀변수 개수는 20개이다. AR과 MA모수에 대한 최대 시차는 24, ARIMA요소에 대한 최대 차분은 3이다.

마. 프로그램 예제

```
series{
  file="c:\x12a\data\mpi.dat"
  start=1980.jan
  format="(6x,12f7.1)"
  span=(1980.1,1998.12)
  modelspan=(1980.1,1998.12)
  name="Industrial production Index(mfg)"
  title = "MPI- adjust holidays for Sul and Chusuk"
  decimals=1
  save=(a1)
}

transform{ function=log
  # file="c:\x12a\data\mpibeapr.dat"
  # format="(8x,12f6.1)"
  # start=1980.jan
  # type=permanent
}
```

```

automdl { }
#arima { model=(0 1 1)(0 1 1) }
identify { diff=(0,1) sdiff=(0,1) }
estimate { }
forecast { }

regression { variables=(tc1987.aug ls1998.jan ls1998.sep)
              user=(sul chosuk)
              usertype=holiday
              file="c:\x12a\data\dagum10a.dat"
              format="(9x,2f10.9)"
              save=(hol)
            }
x11{ save=(b1 d10 d11)
      final=(user)
      appendfcst=yes
      mode=mult | add | logadd | pseudoadd
      seasonalma=s3x5
      trendma=13
    }
outlier{ types=all
          critical=(3.98, 3.98, 3.98)
        }
check { }
slidingspans { }
history { estimates=(sadj sadjchng) target = final start = 1990.jan
          endtable=1998.dec }

```

경기지표 해설

목 차

1. 경기지표의 종류	45
가. 개요	45
나. 개별경제지표	46
다. 종합경기지표	47
(1) 경기종합지수(CI)	47
(가) 개념	47
(나) 구성지표	48
(다) 작성방법	50
(라) 동행지수 순환변동치와 선행지수 전년동월비	53
(마) 해석방법	54
(2) 경기확산지수(DI)	56
(가) 개념 및 작성방법	56
(나) 해석방법	57
라. 기업경기실사지수	59
마. 계량경제모형	63
2. 경기변동 종합판단	63

1. 경기지표의 종류

가. 개요

경기(景氣)란 국민경제의 총체적인 활동수준을 의미하는 것으로 경제활동별로 보면 소비, 투자, 고용 등의 실물부문과 화폐의 수요와 공급 등 금융부문, 그리고 수출과 수입 등 대외부문의 활동을 망라한 거시경제변수들의 움직임이 종합된 것이라고 할 수 있다.

통상적으로 경기가 좋다는 것은 생산, 투자, 소비 등의 경제활동이 대부분의 부문에서 기대하는 평균 수준 이상으로 활발히 나타나고, 경기가 나쁘다는 것은 이의 반대현상인 경우를 의미한다. 장기적인 관점에서 볼 때 경기는 안정되어 있는 것이 아니라 확장→후퇴→수축→회복과정을 반복하면서 끊임없이 변동한다.

이러한 경기의 순환과정은 확장과 수축이 교차하면서 발생하는 반복성, 다수의 경제활동을 포괄하는 다양성, 확장 또는 수축양상이 시차를 두고 경제 각 부문에 전달되는 파급성, 그리고 일정한 방향으로 계속 확대해 나가는 누적성 등의 특징을 갖고 있다.

국민경제가 안정적인 성장을 지속하도록 하기 위해서는 경기의 움직임을 사전에 보다 빠르고 정확하게 예측하여 알맞은 조절정책을 실시하는 것이 대단히 중요하다. 그래서 현재의 경기동향을 판단하거나 미래의 경기를 예측하는 데에는 여러가지 방법들이 사용되고 있다.

첫째, 산업생산지수, 도소매판매액지수, 실업률 등과 같이 경기와 관련성이 높은 개별 경제지표들의 움직임을 분석·판단하는 방법이 있다.

둘째, 경기반영도가 높은 개별 경제지표들을 선정한 후 적절한 통계적 분석방법에 의해 가공하여 합성한 종합경기지표를 사용하는 방법을 들 수 있는데 현재 주요국에서 널리 이용되고 있는 종합경기지표로는 경기종합지수(CI), 경기확산지수(DI) 등이 있다.

셋째, 경제주체(기업가나 소비자)들의 경기에 대한 판단과 전망 등을 조사하여 전반적인 경기동향을 판단하는 설문조사방법이 있는데 기업경기실사지수(BSI)와 소비자태도지수(CSI)가 이에 해당된다.

끝으로 각종 경제변수들의 상호의존관계를 계량경제학적인 측면에서 규명하여 경기를 분석하는 계량경제모형에 의한 분석방법도 사용되고 있다.

나. 개별경제지표

경기의 움직임을 파악하는 개별 경제지표중 가장 대표적인 것으로는 GNI(Gross National Income, 국민총소득)통계가 있다.

GNI란 한 나라의 모든 경제주체가 일정기간 동안에 생산한 재화와 서비스의 부가가치를 금액으로 환산하여 합계한 것으로 각 부문의 생산활동은 물론 소비, 투자, 수출 등 수요동향과 생산활동을 통하여 획득한 소득의 구매력을 살펴볼 수 있는 종합적인 지표라고 할 수 있다. 그런데 GNI통계는 당해년도 또는 분기가 끝난 후 상당기간이 경과한 후에야 추계가 가능하기 때문에 이를 통하여 신속히 현재의 경기상황을 판단하거나 장래의 경기흐름을 예측하기는 어렵다.

따라서 경기동향을 보다 신속히 파악하기 위해서는 가능하면 월별로 발표되는 각종 경제지표들을 이용하여 수요 및 생산의 움직임을 주의깊게 분석·판단해야 한다.

월간 수요의 움직임을 나타내는 경제지표들을 부문별로 살펴보면, 우선 소비활동과 관련되는 지표로서 도소매판매액지수, 소비재출하지수, 소비재수입액 등이 있으며, 투자활동 관련지표로는 건설활동을 나타내는 건축허가면적, 국내건설공사수주액, 건설용중간재출하지수, 시멘트출하량 등과 설비투자동향을 나타내는 국내기계수주액, 설비용기계류내수출하지수, 기계류수입액 등이 있다. 마지막으로 수출입 동향을 나타내는 지표로는 수출액, 수출신용장(L/C)내도액, 수입액 등이 있다.

다. 종합경기지표

산업생산지수, 도소매판매액지수, 실업률 등의 개별 경제지표들은 경제활동의 한 측면만을 나타내고 있기 때문에 국민경제 전체의 경기동향을 파악하기 위해서는 각종 지표들을 종합하여 분석할 필요가 있다. 이에 따라 선진국에서는 경기동향을 민감하게 반영하는 주요 개별경제지표들을 선정한 후 이들 지표를 가공·종합한 종합경기지표를 개발하여 활용하고 있다.

종합경기지표로 경기를 판단한다는 것은 동 지표가 지금까지 어떠한 패턴으로 움직여 왔는가를 측정하고 이러한 움직임으로부터 유의적인 규칙성이나 유사성을 찾아내어 현재의 경기상태를 판단하는 한편 가까운 장래의 경기를 유추하여 예측하는 것을 의미한다.

이와 같이 종합경기지표가 과거의 경험적인 사실에 근거하고 있기 때문에 구조조정의 진전 등으로 전반적인 경제구조가 변화할 경우 경기지표의 경기반영도가 약화되므로 적절한 시기에 구성지표나 합성방법의 변경 등을 통해 경기지표를 개편해야 한다.

종합경기지표는 작성방법에 따라 여러 형태로 분류될 수 있는데 현재 널리 이용되고 있는 것으로는 경기종합지수(Composite Index; CI)와 경기확산지수(Diffusion Index; DI)가 있다.

(1) 경기종합지수(CI)

(가) 개념

경기종합지수는 경제부문의 움직임을 잘 반영하는 경제지표들을 통계적으로 분석·가공·종합하여 작성한다. 경기종합지수는 경기확산지수와 근본적으로 같은 이론적 배경에서 출발하고 있으나 DI

가 각 지표의 변동방향만을 추출하여 작성하는데 비하여, CI는 각 지표의 변화량, 즉 각 개별지표의 전월비 증감률을 통계적으로 종합 가공하여 산출한다. 따라서 CI는 경기변동의 방향, 국면, 전환점 뿐만 아니라 속도와 진폭까지도 파악이 가능한 경기지표이다.

경기종합지수는 1979년의 경기침체를 계기로 경제기획원 조사통계국(현 통계청)과 KDI가 공동으로 연구개발을 시작하여 1981년 3월에 19개 구성지표로 이루어진 CI를 최초로 작성·발표한 이래, 다섯 번의 개편을 거쳐 현재에 이르고 있다.

(나) 구성지표

현행 CI는 국내의 경기에 민감한 지표 중에서 26개의 대표 계열을 선정하여 일반경기(기준순환, reference cycle)와의 시차성에 따라 각각 10개의 선행지표와 동행지표, 그리고 6개의 후행지표로 구성되어 있다.

선행지표계열은 국민경제 전체의 경기동향에 앞서 변동하는 지표를 지칭한다. 예를 들어, 건축허가면적이나 기계수주액 등이 선행지표계열이며, 이들의 움직임을 종합한 지수가 선행종합지수(Leading CI)이다.

동행지표계열은 국민경제 전체의 경기변동과 동시에 변동하거나 국민경제의 변동 자체를 나타내는 지표이며 산업생산지수, 도소매 판매액지수 등이 여기에 포함된다. 이들 계열들을 이용하여 종합적으로 작성된 것이 동행종합지수(Coincident CI)이다.

후행지표계열은 국민경제 전체의 경기변동이 나타난 다음에 변동하는 지표로서 생산자제품재고지수, 비농가실업률 등이 여기에 속한다. 이들 계열로 작성된 지수가 후행종합지수(Lagging CI)이다.

〈경기종합지수 구성지표〉

경제부문	선행지수(10개)	동행지수(10개)	후행지수(6개)
고 용	· 입이직자비율	· 노동투입량(전산업)	· 비농가실업률 · 상용근로자수(전산업)
생 산	· 중간재 출하지수	· 산업생산지수 · 제조업가동률지수 · 생산자출하지수 · 전력사용량(제조업)	
소 비	· 내구소비재출하지수	· 도소매판매액지수 · 비내구소비재출하지수	· 도시가계소비지출
투 자	· 건축허가면적 · 건설용중간재생산지수 · 기계수주액	· 시멘트소비량	· 기계류수입액
재 고	· 재고순환지표		· 생산자제품재고지수
통 화	· 총유동성		· 회사채유통수익률
무 역	· 수출신용장내도액 · 수출용원자재수입액	· 수출액 · 수입액	

(다) 작성방법

선정된 구성지표를 사용하여 선행·동행·후행 경기종합지수를 산출하는데 세부적인 내용은 다음과 같다.

1) 구성지표의 비경기적 변동요인제거

경제시계열(원계열)은 추세변동, 순환변동, 계절변동, 그리고 불규칙변동 등 일반적으로 4가지 요인으로 구성되어 있는데, 이들 요인 중 추세변동과 순환변동을 경기적 변동, 계절변동과 불규칙 변동을 비경기적 변동이라고 부른다.

경기동향을 분석하기 위해서 순수한 경기적 요인만을 추출하기 위해서는 우선 원계열에서 비경기적 요인인 계절변동요인과 불규칙변동요인을 제거해야 한다. 이때 계절요인을 산출하기 위하여 X-12-ARIMA방법을 이용하고, 불규칙요인을 제거하기 위해서는 3개월 이동평균을 사용한다.

2) 개별지표별 전월비증감률 산출

계절·불규칙요인이 제거된 개별지표들의 전월비증감률을 산출한다.

$$\text{금월증감률} = 200 \times \frac{\text{금월치} - \text{전월치}}{\text{금월치} + \text{전월치}} \quad (\text{단, 비율지표는 금월치} - \text{전월치})$$

3) 개별지표 증감률의 진폭표준화

각 구성지표(개별지표)들의 증감률은 진폭에 차이가 있어서 큰 진폭을 갖는 특정개별지표가 종합지수의 움직임을 왜곡시킬 수 있기 때문에 각 구성지표의 증감률을 표준화하여야 한다.

따라서 구성지표들의 전기간 절대치평균이 같은 수준인 “1” 이 되도록 표준화한다.

4) 종합증감률 산출 및 진폭표준화

표준화된 각 구성지표의 증감률을 종합하여 선행, 동행, 후행 종합증감률을 산출한다. 각 지수의 진폭이 서로 다르기 때문에 증감률의 상호비교가 불가능하여 선행, 동행, 후행 종합증감률을 동행 종합증감률 크기로 조정하여 진폭을 표준화한다.

5) 원지수 산출

표준화된 종합증감률을 누적하여 원지수를 산출한다.

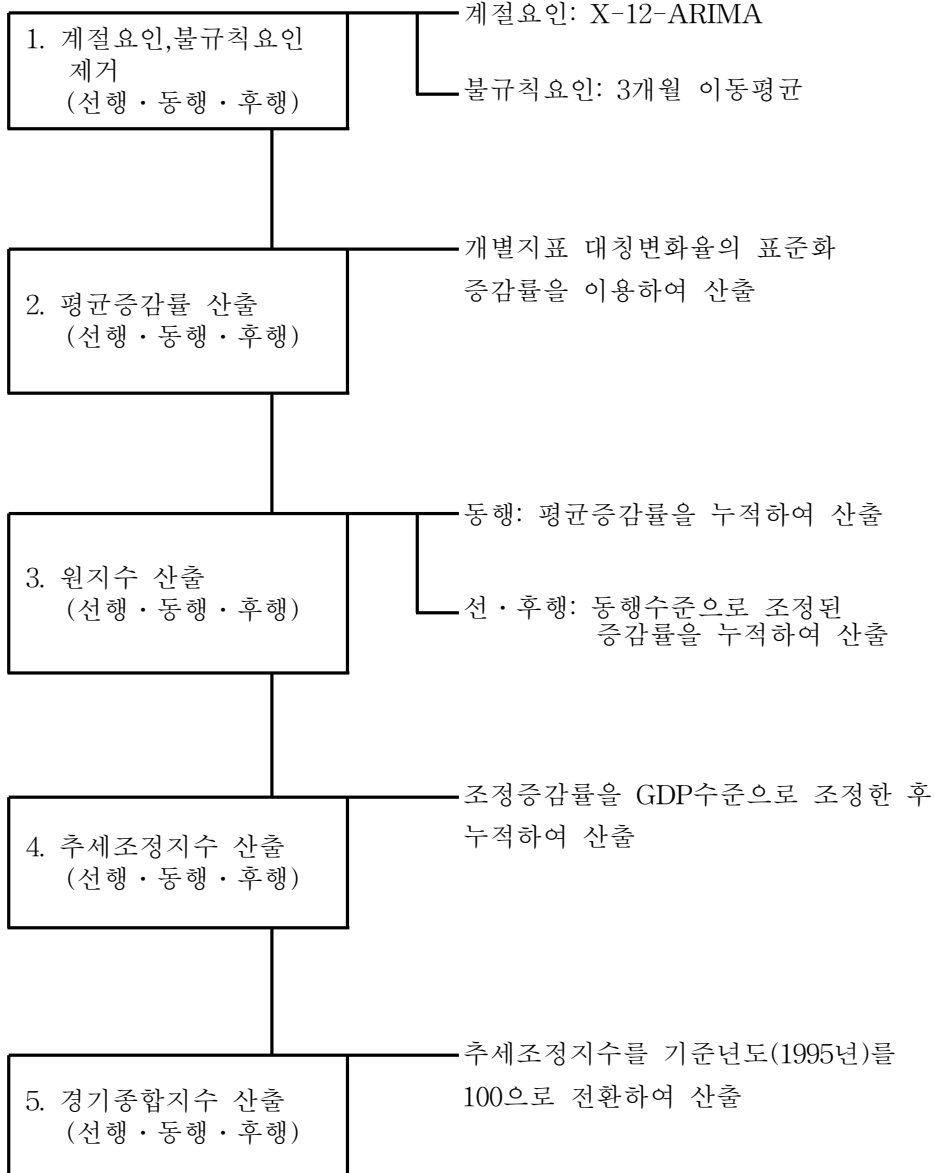
6) 추세조정지수 산출

산출된 선행, 동행, 후행 원지수들의 추세치가 각각 다르기 때문에 국민경제의 장기추세인 GDP추세치에 맞추게 되는데, 원지수 추세치 월평균값과 GDP 월평균값과의 차를 원지수 산출시에 사용한 표준화 종합증감률을 합한 후 지수화하여 추세조정값을 산출한다.

7) 최종 경기종합지수 산출

추세조정값의 기준년도(1995년)를 100으로 한 최종 경기종합지수를 작성한다.

<경기종합지수 작성방법>



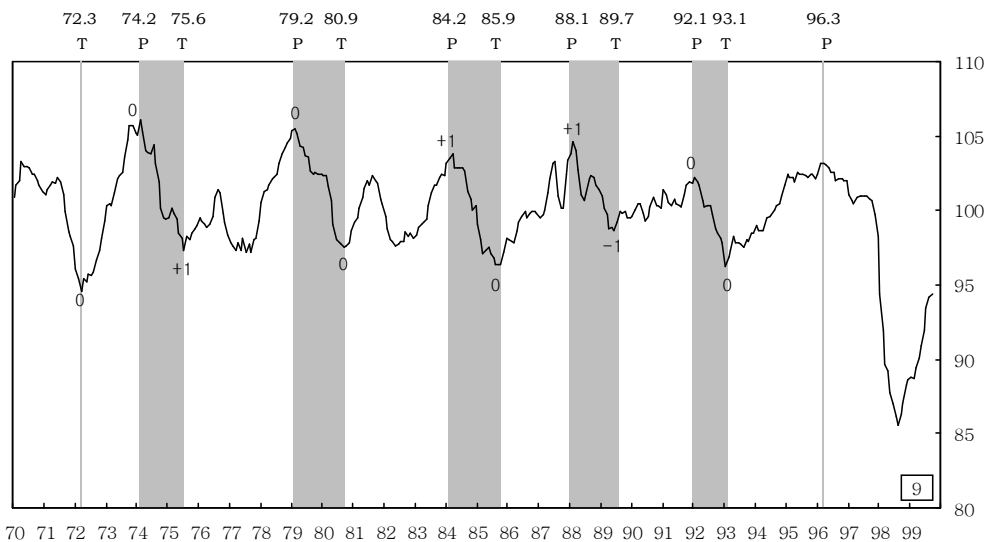
(라) 동행종합지수 순환변동치 및 선행종합지수 전년동월비

동행종합지수 순환변동치는 동행종합지수의 보조지표로서 현재의 경기국면과 전환점을 보다 명확하게 측정·판단하기 위하여 작성한다.

동행종합지수는 경제성장에 따라 증가하는 움직임과 경기의 상승, 하강 움직임을 동시에 포함하고 있는데 그 중 증가추세를 제거하고 경기의 상승, 하강 움직임을 추출하여 순환변동치를 산출한다.

동행종합지수 순환변동치를 해석하는 방법으로는 순환변동치의 절대적 크기 또는 증감차 보다는 그 움직이는 방향(상승 또는 하강)에 더 큰 의미를 둔다.

〈동행지수 순환변동치〉



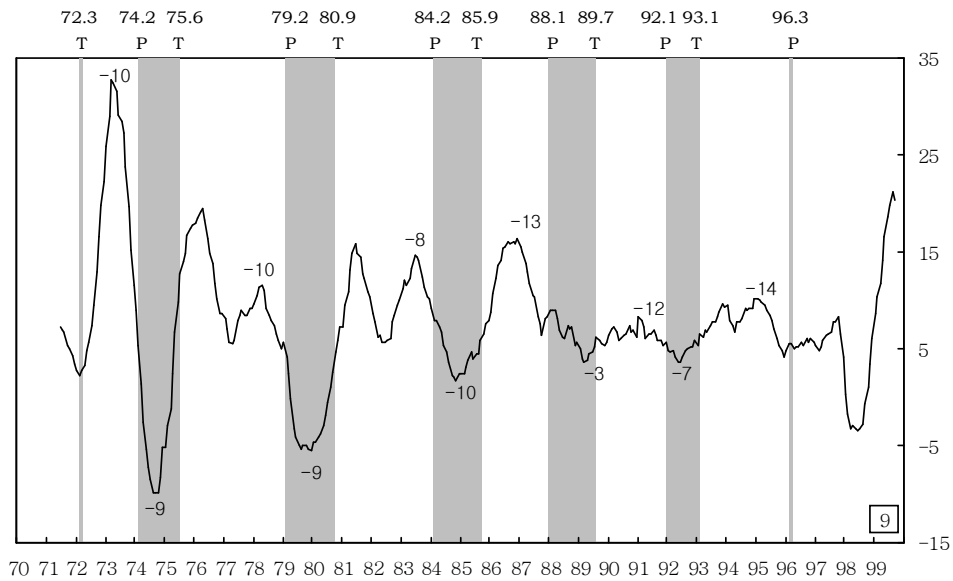
선행종합지수는 앞으로의 경기를 예측하기 위하여 작성하고 있는데 현행 선행종합지수의 선행기간은 약 3~6개월이다. 그러나 선행시차를 더욱 확대함으로써 경기동향을 좀 더 일찍 예측하기 위하여 선행지수 전년동월비를 보조지표로 사용하고 있다.

선행지수 전년동월비도 동행종합지수 순환변동치와 마찬가지로

비율의 크기 또는 진폭보다는 그 움직이는 방향에 더 큰 의미가 있다.

선행지수 전년동월비는 선행지수보다 경기전환점이 더 빨리 나타나는 경향이 있으므로 경기전환점을 예측하는 보조지표로 이용되고 있다. 선행지수 전년동월비의 움직임을 살펴보면 과거 순환주기상 경기정점에서 평균 10.6개월, 저점에서 평균 7.6개월의 선행시차를 보였다.

〈선행지수 전년동월비〉



(마) 해석방법

경기변동의 상황을 판정할 수 있는 명백한 기준설정이 어렵기 때문에 경기종합지수를 해석하는 데에는 약간의 어려움이 있다. 일정한 틀에 의한 해석방법이 있는 것은 아니고 일반적으로는 다음과 같은 방법에 따라 종합적으로 판단, 해석한다.

경기의 변화방향은 경기종합지수의 월별 변화방향을 이용하여 판단하는데 이를 위하여 지수의 전월비를 산출한 후 증감률의 크기를

통해 해석한다. 일반적으로 우리나라의 경우 고성장 국가이므로 동행지수의 증가율이 마이너스를 보이는 경우는 매우 드물다. 따라서 동행지수의 증가율이 장기적으로 0.6%수준(연율7.4%)이상이면 GDP 잠재성장률을 상회하는 성장세 지속, 0.6%수준 미만이면 잠재성장률 이하의 성장세를 시현하고 있다고 볼 수 있다. 그러나 CI와 GDP는 구성지표 및 작성방법에 있어서 차이가 있으므로 항상 일치하지는 않음을 주의해야 한다.

그리고 경기종합지수 해석에 있어서 전반적인 변동추이와 확산도를 동시에 고려하여야 한다. 그 이유는 경기종합지수의 움직임을 월별로 보면 확장(상승)국면내에서도 감소가 나타날 수 있고 반대로 수축(하강)국면 내에서도 일시적으로 증가가 나타날 수 있기 때문이다. 또한 지수의 움직임이 증가(또는 감소)하였다고 하더라도 구성지표 중 몇 개의 지표가 증가(또는 감소) 하였는지가 실제적으로 더 중요할 수 있기 때문이다.

한편 경기의 상승과 하강을 나타내는 순환변동치는 추세치가 포함된 지수와는 해석방법이 다소 다르다. 경기종합지수에서는 지수 수준과 증감률의 크기로 경기의 상승수준이나 속도를 판단할 수 있으나 순환변동치에서는 그렇지 못하다. 왜냐하면 순환변동치의 주된 작성목적은 경기가 움직이는 방향과 국면 및 전환점을 보다 용이하게 판단하는데 있으며 추세치를 산출하는 방법과 시기, 그리고 대상기간에 따라 순환변동치의 수준과 증감률이 달라질 수 있기 때문이다. 따라서 순환변동치의 수준 자체를 이용한 경기판단은 오류를 범할 수 있다.

예를 들어 특정월의 순환변동치가 103이라고 할 때 그 수준자체는 상당히 높다고 할 수 있으나 그 의미는 주변상황에 따라 달라지게 된다. 즉 확장국면에서의 103이라는 의미와 수축국면에서의 의미는 전혀 다를 수 있으며, 어느 한 국면에서 103을 기록했다고 해

서 그것이 다른 국면에서의 99보다 반드시 더 좋은 경기를 나타내고 있다고 단정하기는 어렵다.

(2) 경기확산지수(DI)

(가) 개념 및 작성방법

경기확산지수는 반복성, 파급성, 누적성의 특성을 갖는 경기변동이 경제의 각 부문, 즉 고용·생산·투자·소비 등의 특정부문에서 시작하여 점차 타 부문으로 확산 파급되어 가는 과정을 파악하기 위하여 각 개별지표가 지니고 있는 경제적 의미의 중요성이나 지표간의 상호관계 등을 고려하지 않고 개별지표의 변화방향만을 종합하여 지표화한 것이다.

즉, 경제 각 부문의 변동을 나타내는 경제통계지표 중 전체경제의 변동과 밀접한 대응성을 가지고 움직이는 지표를 선정한 후 이를 선행·동행·후행지표군으로 구분하여 선행지표군의 변동으로 경기를 예측하고 동행지표군의 변동으로 현재의 상황을 측정하며 후행지표군의 변동으로 확인하는 방법이다.

여기서 말하는 대응성과 선행, 동행, 후행이란 과거 전체경제의 움직임과 개별지표간에 나타났던 관계를 토대로 결정되는 것이며, 이와 같은 관계가 앞으로도 계속될 것이라는 전제에 입각하고 있다. 경기확산지수는 경험적이고 징후적인 접근방법으로서 경기변동의 방향과 전환점의 파악이 용이하고 경기변동의 단기예측이 가능하다.

경기확산지수는 1936년 Geoffrey Moore 박사에 의해 장기간에 걸친 통계자료 분석의 부산물로 개발되어 현재 미국, 일본 등 세계

각국에서 작성되고 있으며 특히 일본에서는 경기변동의 측정과 예측을 위한 주요지표로 활용되고 있다.

DI의 종류에는 경기확산지수(DI), 누적확산지수(CDI), 역사적확산지수(HDI) 등이 있으나 오늘날에는 보통 DI와 CDI를 주로 사용하고 있다.

경기확산지수(DI)의 산식은 다음과 같다.

$$DI = \frac{\text{증가지표수}(\text{보합지표수} \times 0.5)}{\text{구성지표수}} \times 100$$

(나) 해석방법

경기확산지수는 50을 기준으로 0~100의 범위 내에서 움직이게 되는데 지수수준이 50을 초과하면 경기확장국면, 50미만이면 경기수축국면, 50이면 경기전환점을 나타낸다고 할 수 있다. 일반적으로 경기국면의 정점에서는 50선을 위에서 아래로 끊고, 반대로 저점에서는 50선을 아래서 위로 끊게 된다. 따라서 선행확산지수가 50선을 위로 끊게 되면 가까운 장래에 경기가 상승할 것임을, 50선을 아래로 끊게 되면 경기가 하강할 것임을 예시한다.

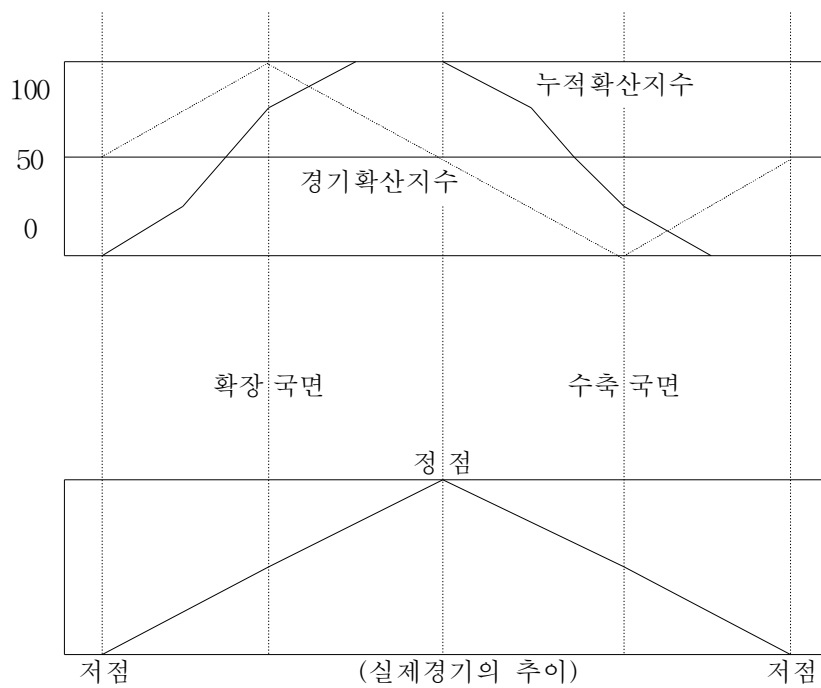
경기확산지수는 월별 변동의 등락이 심한 경우에는 경기전환점의 확인이 어렵게 되므로 누적확산지수(CDI)를 산출하여 보완적으로 사용한다.

$$\text{금월 CDI} = \text{전월 CDI} + (\text{금월 DI} - 50)$$

하지만 경기확산지수는 구성지표의 확산비율이 불규칙하게 증감하는 경우 경기판단에 어려움이 발생하게 된다. 그리고 개별 경제지표들의 변화방향만을 종합한 경기지표이므로 경기확산지수 자체가 경기의 변동속도를 정확히 나타내는 것은 아니다.

〈경기확산지수의 작성 例 및 변동추이〉

	1월	2월	3월	4월	5월	6월	7월	8월	9월
구 성 지 표 수	20	20	20	20	20	20	20	20	20
증 가 지 표 수	10	15	20	15	10	5	0	5	10
경기 확산 지수	50	75	100	75	50	25	0	25	50
누적 확산 지수	0	25	75	100	100	75	25	0	0



라. 기업경기실사지수

(1) 개념

기업경기실사지수(Business Survey Index, BSI)는 기업활동의 실적, 계획 및 경기동향에 대한 기업가의 판단을 설문조사를 통하여 경기동향을 파악하고자 하는 단기경기예측방법 중의 하나이다.

기업가는 기업활동을 위해서 전반적인 국내외 경기동향과 업계의 영업상태 등을 면밀히 분석, 예측하고 이를 토대로 경영계획을 수립하게 된다. 따라서 기업실사지수는 시장경제의 주체인 기업가의 경기에 대한 판단이나 전망이 정확할 뿐만 아니라 전체 경기에 중대한 영향을 미친다는 사실에 근거를 둔 것이다.

이와 같은 기업경기실사조사는 1920년대에 미국, 일본 등에서 시작되어 현재는 독일, 호주, 대만 등 대부분의 국가에서 작성하고 있으며 지수작성이 용이하므로 경기종합지수 및 경기확산지수와 함께 경기변동을 측정하는 방법의 하나로 널리 이용되고 있다.

우리 나라에서는 1964년 한국생산성본부가 처음으로 기업경기전망조사를 실시한 이래 현재 한국산업은행, 한국은행, 대한상공회의소, 중소기업은행, 전경련, 노동부 등 6개 기관에서 월별 또는 분기별로 실적치와 전망치를 작성하고 있다.

〈기관별 기업실사조사 현황〉

(1999. 10월 현재)

통계명칭	KDB산업 경기전망조사	중소제조업 경기전망조사	기업 경기조사	기업경기 전망조사	기업경기 동향조사	고용동향 전망조사
작성기관	한국산업은행	중소기업은행	한국은행	대한상공회의소	전경련	노동부
작성주기	분기	분기	분기	분기	월	분기
조사대상	종업원 200인 이상인 1,218개 제조업체	종업원 5~300인 미만의 1,606개 중소제조업체	2,893개 기업체 (제조업 : 1,721 비제조업 : 1,172)	상시종업원 20인 이상의 1,993개 광공업체	업종별 매출액 상위500개 광공업체	종업원 5인 이상의 4,909개 전산업
조사항목	· 전분기 실적, 당분기 잠정 향후 1분기 전망 · 판단조사 : 21 계수조사 : 5	· 전분기 실적 및 향후 2분기 전망 · 판단조사 : 10	· 전분기 실적 및 향후 1분기 전망 · 판단조사 : 10	· 전분기 실적 및 향후 1분기 전망 · 판단조사 : 13	· 전분기 실적 및 향후 1개월 전망 · 판단조사 : 13	· 전분기 실적, 당분기 잠정 향후 1분기 전망 · 기타조사 : 4
특성	①제조업 BSI 조사 ②실물경제지표 에 대해 회귀 분석에 의한 예측치 추정	중소제조업체 ┌중화학 ┌중기업 └경공업 └소기업 ┌조립가공형 └기초소재형 └생활관련형	┌제조업 └비제조업	광공업 ┌광업 └제조업 5대광역시 발표	광공업 ┌광업 └제조업 ┌중화학 └경공업	고용 BSI 및 고용대처방안

(2) 작성방법

기업경기실사지수(BSI)는 전체 응답업체 중에서 증가로 응답한 업체수와 감소로 응답한 업체수의 차이를 계산하고 이것을 백분비로 표시한 것이다. 현재 우리 나라에서 일반적으로 사용하는 산식은 다음과 같다.

$$BSI = \left(\frac{\text{증가응답업체수} - \text{감소응답업체수}}{\text{전체응답업체수}} \right) \times 100 + 100$$

(3) 해석방법

기업경기실사지수는 100을 기준으로 0~200의 범위 내에서 움직이게 된다. 즉 모든 응답업체가 증가로 응답을 하면 지수는 200이 되고, 반대로 전부 감소로 응답을 하면 지수는 0이 되며 증가와 감소로 응답한 업체가 동일한 경우에는 100이 된다. 따라서 지수가 100을 초과하면 경기가 호전, 100미만이면 악화, 100이면 전기와 동일할 것으로 판단한다.

하지만 기업경기실사지수는 작성방법이 간단하여 신속히 경기동향을 파악할 수 있는 장점이 있으나 응답자가 조사시점의 여건 변화에 지나치게 민감한 반응을 나타낼 가능성이 있고 경기변동의 속도 및 전환점 측정이 불가능하여 지수자체의 의미가 약하다는 단점도 있다.

<참고> 경기종합지수, 경기확산지수 및 기업경기실사지수 비교

	경기종합지수(CI)	경기확산지수(DI)	기업경기실사지수(BSI)
작성목적	<ul style="list-style-type: none"> - 경기전환점의 예측, 판단 - 경기국면의 판단 - 경기변동의 크기, 속도 측정 	<ul style="list-style-type: none"> - 경기전환점의 예측, 판단 - 경기국면의 판단 	<ul style="list-style-type: none"> - 기업가의 경기에 대한 판단, 전망을 조사하여 경기동향분석자료로 활용
작성방법	<ul style="list-style-type: none"> - 개개의 변화율을 구하여 변화폭을 표준화한 후 합성, 누적하여 지수화시킴 	<ul style="list-style-type: none"> - 총구성지표수에서 증가한 계열의 비율(%) 	<ul style="list-style-type: none"> - 전체 응답업체 중 증가업체수가 차지하는 비율로 100미만이면 경기악화, 이상이면 경기호전
특징	<ul style="list-style-type: none"> - 선행, 동행, 후행계열내의 경기과급도, 침투도는 나타나지 않음 - CI의 움직임은 경기변동의 상대적인 크기를 나타냄 	<ul style="list-style-type: none"> - 경제 각부문으로 과급, 침투도를 선행, 동행, 후행계열별로 파악 가능 - DI수준의 변화는 경기변동 크기의 변화를 나타내지 않음 	<ul style="list-style-type: none"> - BSI는 조사응답의 수준 및 조사시점의 여건변화에 민감하고, 경기변동속도 및 전환점 측정이 불가능함.

마. 계량경제모형

계량경제모형(Econometric Model)이란 한나라의 전체적인 경제흐름을 설명할 수 있도록 각 경제부문별로 경제변수들간의 인과관계를 구조방정식(structural equations) 체계로 모형화(specification)한 것으로, 이를 이용하여 경기변동을 분석·예측할 뿐만 아니라 각종 경제정책의 효과분석도 가능하다. 계량경제모형을 작성하는 방법은 상당히 복잡하며, 다양한 경제·통계이론과 추정기법이 이용된다.

계량경제모형에 의한 방법은 경기변동의 속도는 측정할 수 있으나 경기변동의 방향이나 전환점 측정이 어렵고, 또한 외생변수의 설정에 있어서 개인적인 판단이 개재될 수 있다는 단점이 있다. 그러나 이 방법은 이론적 합리성, 성장속도 측정, 국제비교 등에 있어서 유리하기 때문에 오늘날에는 중장기 예측에 널리 쓰이고 있다.

2. 경기변동 종합판단

위에서 살펴본 바와 같이 경기변동을 측정하고 예측하기 위한 경기지표나 방법에는 여러 가지가 있다. 그러나 어느 한 두가지의 경기지표나 경기측정 방법으로 급격히 변화하는 경기변동상황을 정확히 측정하고 예측하는 데에는 한계가 있다. 왜냐하면 경기변동에는 여러 가지 요인들이 복합되어 있으며 이러한 요인들이 경제 여러 부문에서 서로 다른 시차를 가지고 지속적, 누적적으로 변화하면서 나타나기 때문이다. 또한 현재 이용하고 있는 경기지표나 경기변동 측정수단은 제각기 기술상의 한계점을 갖고 있기 때문이다.

경기확산지수와 기업경기실사지수는 경기변동의 변화 방향을 측정·판단하는 데에는 유용하지만 변동속도를 측정할 수 없는 반면, 계량경제모형은 변화속도의 측정과 중장기 예측이 가능하다는 이점은 있으나 경기변동의 방향이나 전환점의 측정이 불가능하다는 단점이 있다.

경기종합지수는 경기변동의 방향, 속도, 전환점 및 국면을 동시에 측정할 수 있고 단기예측이 가능하다는 점에서 현재까지 개발된 종합적인 경기지표로서는 가장 유용한 방법이라고 할 수 있다. 그러나 경기종합지수는 개별구성지표가 과거와 동일한 패턴을 보이면서 변해간다는 전제조건하에서 작성되었기 때문에 과거와는 비교할 수 없을 정도로 급격히 변화해 가는 경기를 정확하고 완전하게 측정한다는 것은 어려운 일이라고 할 수 있다.

따라서 경기변동을 측정할 때에는 어느 특정한 경기지표만을 이용하기보다는 여러 경기지표나 방법 및 기초통계 등을 상호보완적으로 이용하는 것이 효과적이라고 할 수 있다. 경기동향판단 및 경기예측의 제고를 위해서는 경기종합지수의 움직임 외에도 GDP, 산업생산지수, 실업률, 환율, 기업경기실사지수(BSI) 등 주요경제지표들의 움직임, 당시의 경제여건 및 경제정책의 방향 등을 종합적으로 활용할 필요가 있다.

EViews 사용방법

목 차

1. EViews 입문	69
1.1 EViews란 무엇인가?	69
1.2 EViews 설치	69
1.3 EViews 시작하기	70
1.4 EViews 창	71
1.5 EViews 종료	74
1.6 도움말 보는 방법	74
2. EViews 기초	77
2.1 작업파일(Workfile)	77
2.2 개체(Object)	85
2.3 명령어	99
3. 기본적인 데이터 처리	100
3.1 데이터 읽기(Importing Data)	100
3.2 데이터 내보내기(Exporting Data)	117
3.3 데이터 개체(Data Objects)	119
3.4 표본(Samples)	127
4. 그래프(Graphs), 표(Tables)	133
4.1 그래프(Graphs)	133
4.2 다중 그래프 개체(Multiple Graph Objects)	146

4.3	그래프 출력	148
4.4	다른 프로그램에 그래프 복사하기	149
4.5	표(Tables)	150
4.6	다른 프로그램에 표 복사하기	151
5.	데이터 작업(Working with data)	153
5.1	식 이용하기(Using Expressions)	153
5.2	계열 작업(Working with series)	159
5.3	자동계열의 작업(Working with Auto-series)	163
5.4	그룹 계열로 작업하기(Working with Groups of series)	167
5.5	스칼라 작업(Working with Scalars)	168
5.6	계절 조정(Seasonal Adjustment)	169
5.7	추세치산출(Hodrick-Prescott 필터)	172
6.	EViews 데이터베이스	179
6.1	개요	179
6.2	데이터베이스 기초(Database Basics)	180
6.3	데이터베이스에서의 개체 작업(Working with Objects in Database)	184
6.4	데이터베이스 자동계열(Database Auto-Series)	192
6.5	데이터베이스 등록기(Database Registry)	194
6.6	데이터베이스 검색기(Querying the Database)	197
6.7	데이터베이스 관리	204
6.8	명령어	207

1. EViews 입문

1.1 EViews란 무엇인가?

EViews(Econometric Views)는 윈도우 환경에서 쉽게 사용할 수 있고 복잡한 데이터 분석, 회귀분석 및 예측을 할 수 있는 기능과 프로그래밍이 동시에 가능한 응용 프로그램이다.

EViews는 대형 컴퓨터에서 다루는 시계열 처리 소프트웨어에서 발전된 것으로 시계열 데이터를 처리하는 기능들이 모여 있는 도구라고 할 수 있다. EViews는 1981년에 발표된 MicroTSP에서 발전된 것이다.

EViews는 키보드로 데이터를 입력하거나 디스켓 등에서 불러오는 방식, 기존의 파일에 계열을 추가하고 데이터를 화면으로 보거나 프린트하는 방식, 계열들간의 관계를 통계적으로 분석하는 일 등을 아주 편리하게 수행 할 수 있다.

EViews는 윈도우 환경에서 작업할 수 있어서 윈도우 메뉴와 창에서 원하는 작업을 마우스로 선택하여 사용할 수 있다. 또한 윈도우의 메뉴들을 이용하여 작업할 수도 있고 EViews 자체의 명령어를 직접 입력하여 작업을 수행할 수도 있다. 또한 데이터를 처리하기 위해 명령어로 프로그램을 만들어 이용할 수도 있다.

1.2 EViews 설치

EViews를 설치하기 전에 설치하고자 하는 컴퓨터가 다음의 윈도우 환경에 맞는지를 확인해야 한다.

1. 386, 486, 펜티엄, Microsoft의 윈도우 3.1, 윈도우 95(98), 또는 윈도우 NT
2. 윈도우 3.1의 RAM 4Mbyte 이상, 8Mbyte이상이면 더욱 좋다. 윈도우 95(98)나 NT일 경우에는 대용량의 RAM일수록 좋다.
3. 하드디스크 공간은 10Mbyte 이상이면 충분하다.

설치에 필요한 EViews 디스켓은 버전 3.1기준으로 모두 4장이며 설치 방법은 다음과 같다.

1. 윈도우가 실행되면 다른 프로그램이 EViews 설치를 방해하지 않도록 EViews를 설치하기 전에 다른 모든 프로그램을 닫아야 한다.
2. 설치 디스크(디스크 1)를 드라이브에 넣는다.(일반적으로 A:)
3. 윈도우 95(98)에서는 '시작'을 선택한 후 '실행(R)'을 선택한다.

4. 'A:SETUP'을 입력하고 엔터키를 누른다.(드라이브가 A가 아닐 경우 해당 드라이브 이름을 쓴다)

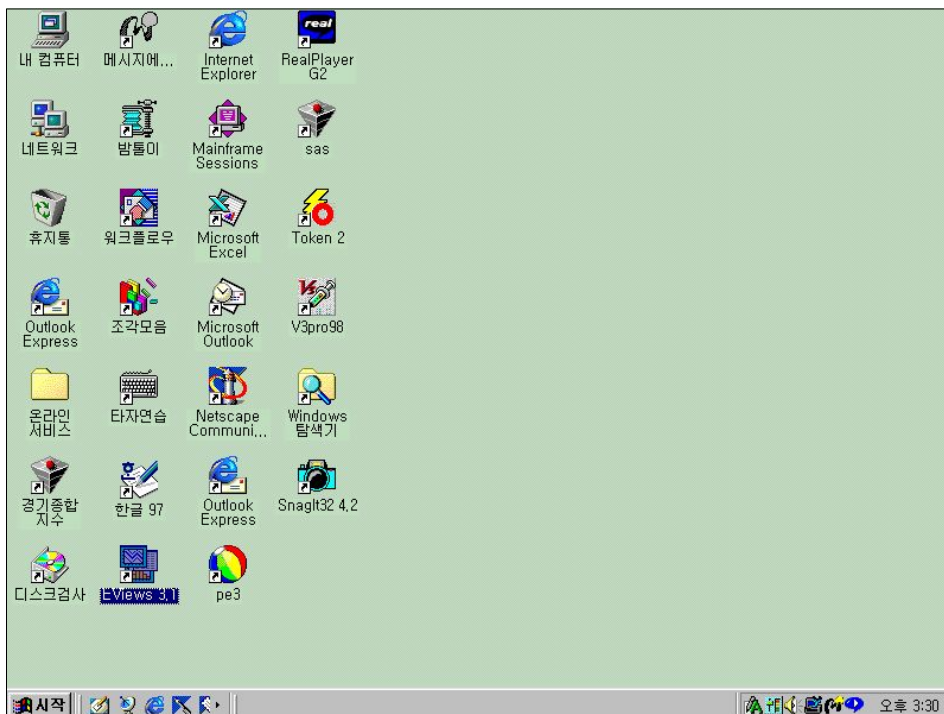
화면에 나타나는 지시에 따라 차례대로 디스켓을 넣어 주고 모두 끝나면 1번 디스켓을 다시 넣으라는 메시지가 나온다. 설치가 완료되면 설치가 성공적으로 완료되었다는 메시지가 나온다.

1.3 EViews 시작하기

윈도우 95(98) 환경에서 EViews를 시작하는 방법은 몇 가지가 있다.

- 작업표시줄에 있는 '시작'을 클릭하고 프로그램에서 EViews3 디렉토리 중에서 EViews3.1 프로그램 아이콘을 선택한다.
- 윈도우 탐색기나 바탕화면의 내 컴퓨터에서 EViews 디렉토리를 선택하여 EViews3.1 프로그램 아이콘을 더블클릭한다.
- 윈도우 바탕화면에 EViews3.1 단축아이콘을 만들어 이 아이콘을 더블클릭한다.
- EViews 작업파일이나 데이터베이스 아이콘을 더블클릭한다.

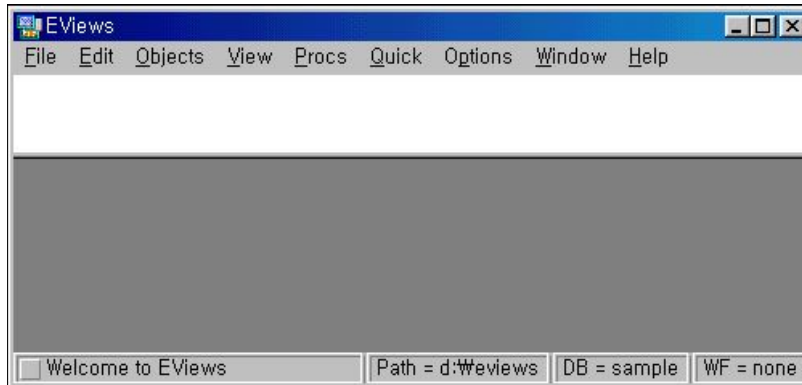
<그림 1.1> 윈도우 바탕화면에서의 EViews3.1 아이콘



1.4 EViews 창

프로그램이 실행되면 윈도우에서 다음과 같은 EViews 창이 나타난다.

<그림 1.2> EViews 초기화면



EViews 창에는 다음과 같은 것들이 있다.

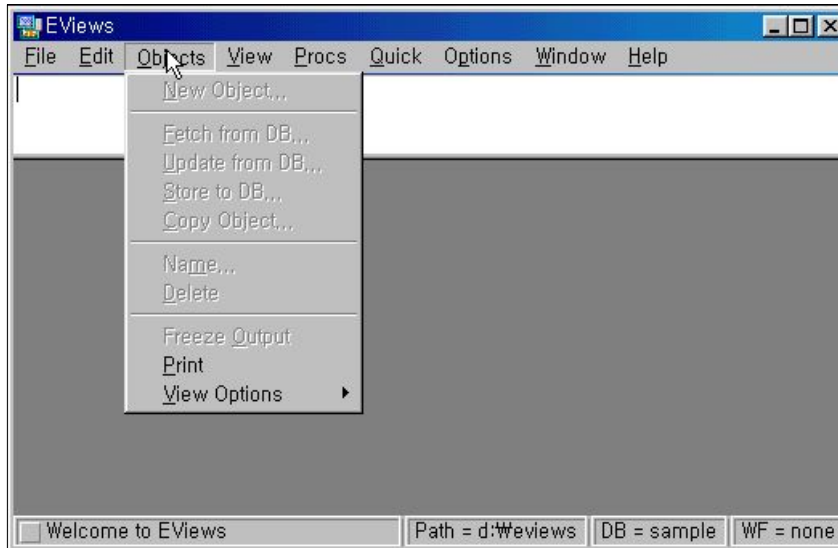
1) 제목 표시줄(Title Bar)

제목 표시줄은 창의 맨 위 부분에 EViews라고 쓰여져 있는 부분이다. EViews 프로그램이 실행되어 창이 활성화되면 다른 비활성창과 달리 제목 표시줄의 색깔이나 명암이 다르다.(보통 비활성창은 제목표시줄의 색깔이나 명암이 흐리다). 다른 프로그램이 활성화되면 EViews 제목 표시줄은 흐려지고 다시 EViews를 활성화시키기 위해서는 EViews의 아무 곳을 클릭하거나 EViews가 활성화 될 때까지 Alt와 Tab을 동시에 누르면 된다.

2) 메인 메뉴(Main Menu)

제목표시줄 아래에 있는 것들이 메인 메뉴(main menu)이다. 메인 메뉴 중 어느 하나를 클릭하면 그에 따른 부메뉴(drop-down menu)가 나타난다. 부메뉴 중 어느 하나를 선택하면 그 항목의 색깔은 진하게 표시된다.

<그림 1.3> EViews 메인 메뉴

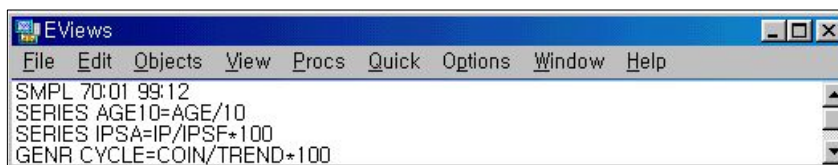


예를 들어, 메인 메뉴에서 'Objects' 항목을 클릭하여 부메뉴들이 나타나게 해 보자. 부메뉴는 진한 색과 흐리게 되어 있는 것이 나타나는 것을 볼 수 있다. 진한 색으로 되어 있는 메뉴들은 실행할 수 있지만 흐리게 되어 있는 것은 실행할 수 없다. 위의 그림에서는 'New Object' 나 'Store' 등은 실행시킬 수 없는 메뉴들이지만 'Print'나 'View Option'은 실행시킬 수 있는 메뉴이다.

3) 명령어 창(Command Window)

다음 그림에서 메인 메뉴 밑에 있는 부분을 명령어 창이라고 부른다. EViews 명령어는 여기서 입력할 수 있다. 명령어를 입력하고 엔터키 (enter key)를 누르면 그 명령이 실행된다.

<그림 1.4> EViews 명령어 창



명령어 창 안에 있는 세로막대의 커서는 키보드 상에서 명령어를 입력하는 위치에 놓이게 된다. 일반 문서 편집기에서처럼 마우스로 원하는 위치를 클릭하면 커서를 움직일 수 있다. 이 커서가 보이지 않을 때는 명령어 창이 활성화되어 있지 않은 상태이다. 이 때는 명령어 창의 아무 곳이나 클릭하여 원하는 곳에 명령어를 입력할 수 있다.

이미 실행된 명령어일지라도 다시 커서를 옮겨 수정한 후 엔터를 누르면 수정된 명령어를 실행시킬 수 있다.

명령어 창에서는 복사하여 붙이기 기능이 있어 EViews의 다른 창에 있는 문자나 또 다른 프로그램에 있는 문자들을 복사하여 쉽게 편집할 수 있다. 또 나중에 다시 사용할 수 있도록 명령어 창에 있는 내용들을 텍스트 파일로 저장할 수도 있는데 이것은 명령어 창이 활성화된 상태에서 메인 메뉴의 File/Save As...를 이용하면 된다.

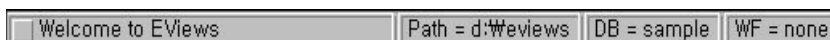
명령어 창에 입력한 내용이 너무 많을 때에는 스크롤 바에 있는 상, 하, 좌, 우의 화살표를 움직여서 내용을 볼 수 있다.

기본으로 설정되어 있는 명령어 창은 창의 밑 부분을 마우스로 누른 상태에서 끌어 내리거나 올려서 원하는 크기로 다시 조절하면 된다.

4) 상태 표시줄(Status Line)

윈도우 창의 맨 밑부분에는 상태 표시줄이 있는데 이것은 몇 개의 부분으로 나뉘어져 있다.

<그림 1.5> EViews 상태 표시줄



상태 표시줄의 제일 왼쪽에 있는 것은 EViews를 처음 시작할 때 나타나는 메시지이다. 이것은 상태 표시줄의 제일 왼쪽에 있는 작은 상자를 클릭하면 없어진다. 그 다음 부분은 EViews가 데이터와 프로그램을 이용하는 기본 디렉토리가 표시되어 있다. 그리고 마지막 두 개 부분에는 기본 데이터베이스와 현재의 작업파일 이름이 표시된다.

5) 작업영역(Work Area)

창의 중간부분에 있는 것을 작업 영역이라고 하는데 여기에는 EViews가 실행하는 많은 작업들이 표시된다. 이런 창들은 우리가 책상에서 일을 할 때 여러 장의 종이를 펼쳐놓고 작업을 하는 것과 비슷하다. 여러 창들 중에서 현재 작업하고 있는 창이 맨 위에 위치하여 서로 겹쳐지게 되는데 활성화 된 창의 제목 표시줄만 색깔이 밝아진다.

창들이 서로 겹쳐져 있을 때는 밑에 있는 창의 제목 표시줄 또는 보이는 부분을 아무 곳이나 클릭하면 그 창이 맨 위로 오게 된다. 또 F6 키나 Ctrl+Tab을 이용하면 여러 개의 창들이 순서대로 위로 올라오게 된다.

또 메인 메뉴의 창('Window')을 클릭하여 원하는 창의 이름을 선택하면 곧바로 그 창으로 이동할 수도 있다.

창의 제목 표시줄을 클릭하여 마우스로 끌어서 원하는 위치로 이동시킬 수도 있고 창의 오른쪽 하단의 모서리를 끌어서 창의 크기를 조절할 수도 있다.

1.5 EViews 종료

EViews를 종료할 때도 몇 가지 방법이 있다. 메인 메뉴의 File/Exit를 선택하거나 Alt+F4를 눌러서 종료시킨다. 또한 윈도우에서 실행중일 때에는 EViews 창의 오른쪽 상단에 있는 종료 버튼을 누르거나 왼쪽 상단에 있는 EViews 아이콘을 더블 클릭하면 종료된다.

EViews 종료시에는 항상 저장되지 않은 파일에 대한 저장여부를 묻는 메시지가 나온다.

1.6 도움말 보는 방법

1) EViews 메뉴얼

'User's Guide' 는 EViews를 수행하기 위해 필요한 여러 가지를 설명하고 있다. 앞부분에서는 기본적인 운용 방법에 대한 설명이었고 중간부분에서는 기본적인 계량경제를 다루는 방법들에 대해서 다루고 있다. 또 뒤 부분에서는 좀 더 깊이 있는 방법들을 다루고 있다.

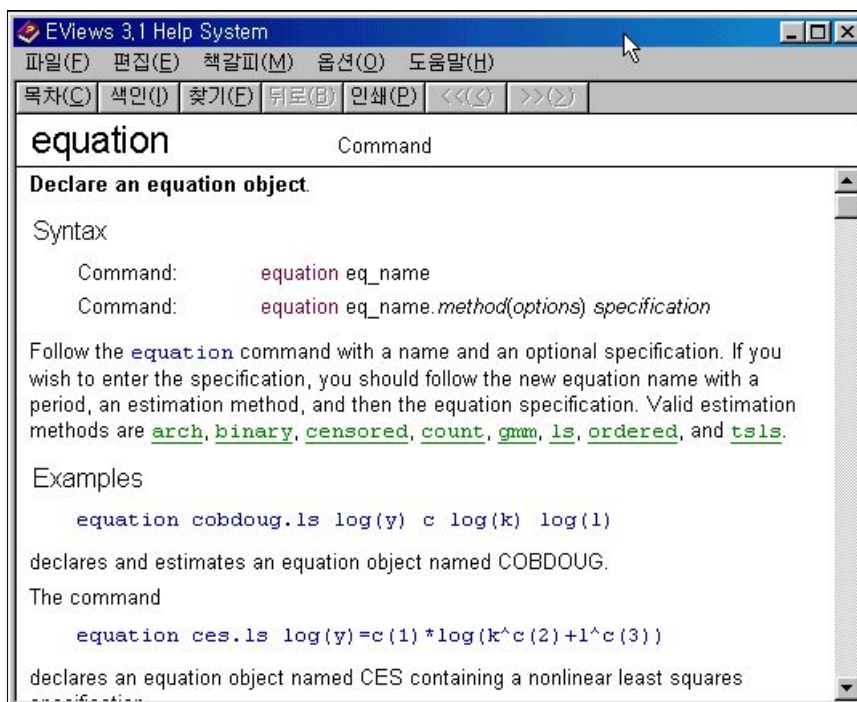
EViews의 다른 메뉴얼인 ‘*Command and Programming Reference*’에는 EViews 명령어에 대하여 구체적이고 체계적으로 설명을 해 놓았다. 이 메뉴얼의 뒤 부분은 EViews 언어로 프로그램을 작성하고자 사람들을 위한 것이다. 이 부분에서는 EViews를 대화형식으로 작업하지 않고 명령어를 이용해서 모든 작업을 할 수 있다.

2) 도움말(Help)

도움말 기능을 이용해서 EViews 내에 내장되어 있는 EViews 각각의 기능에 대한 해설을 찾아 볼 수가 있다. 도움말을 보기 위해서는 메인 메뉴의 도움말(‘Help’)을 선택하면 된다.

예를 들어 EViews의 도움말에서 “equation”이란 항목을 보기 위해서는 EViews의 메인 메뉴에서 Help/EViews Help Topics...를 클릭하면 도움말 항목이 나타나는데 여기서 색인부분의 첫줄에 “equation”를 입력하면 다음과 같은 도움말이 나타난다.

<그림 1.6> EViews 도움말(예: equation)



3) The World Wide Web

EViews 메뉴얼과 도움말에서 제공되는 정보들 이외에도 인터넷의 웹사이트상에서 EViews의 홈페이지(<http://www.eviews.com>)에 접속하면 EViews의 가격, 구매방법, 기능 등을 알아 볼 수 있다. 또한 온라인 상에서 EViews 3.0을 업데이트 할 수 있고 표본 데이터와 프로그램 이 외에도 많은 것들을 제공받을 수 있다. Eviews를 구입하게 되면 동봉된 프로그램과 해설서 이외에도 많은 것들을 제공하고 있는데 현재 사용하는 EViews가 약간의 보정이나 변화가 있을 때는 웹사이트상에서 구매자들이 다운로드 할 수 있도록 연락을 해 주고 있다. QMS(Quantitative Micro Software)를 구매한 고객들은 현재의 버전을 언제든지 자유롭게 다운로드하여 업데이트 할 수 있다.

2. EViews 기초

하나의 작업을 처리할 때 매우 많은 일들을 수행하게 되는데 이런 많은 작업들을 처리하는 것은 복잡하고 시간이 많이 걸리는 것도 있다. 다행히 EViews는 체계적인 일 처리로 작업을 할 수 있도록 프로그램을 설계하였다.

EViews는 하나의 개체(object) 개념이다. 즉, 사용하기 쉬운 기능들을 서로 묶어 관련있는 정보와 연관되어 작동할 수 있도록 한 개체들의 집합이다. 실제로 EViews에서 수행하는 작업들은 다양한 개체들을 이용하여 다루는 것이다.

EViews는 개체 포함기 속에 이런 모든 개체들을 포함하고 있다. 개체 포함기는 서류함이나 작업할 때 사용하는 많은 개체들의 조직 같은 것이다. 처음에는 이 개체의 개념을 외부에서 찾으려고 할 지도 모르겠으나 기본적인 개념은 다루기 쉽고 EViews에서 대부분 작업의 기본이 되는 것들이다. 그러나 처음부터 이 모든 개념들을 이해할 필요는 없다. 작업을 하나 하나 해 나가면서 이런 개체들이나 작업파일을 이해하면서 EViews를 수행시켜 나가면 된다.

2.1 작업파일(Workfile)

EViews의 모든 개체는 개체 포함기에서 열어야 한다. EViews에서 행하는 대부분의 작업은 개체를 포함하고 있는데 이 개체는 작업파일에 들어 있어서 작업의 첫 단계는 새로운 작업파일을 생성하거나 이미 저장되어 있는 작업파일을 불러오는 것이다.

작업파일은 두 가지 중요한 성질을 가지고 있다. 첫째, 작업파일안에 있는 개체들을 빠르게 접속할 수 있도록 RAM에 저장되어 있다. 둘째, 이 작업파일들은 주기(Frequency)와 범위(Range)에 따라 특징지어 진다.

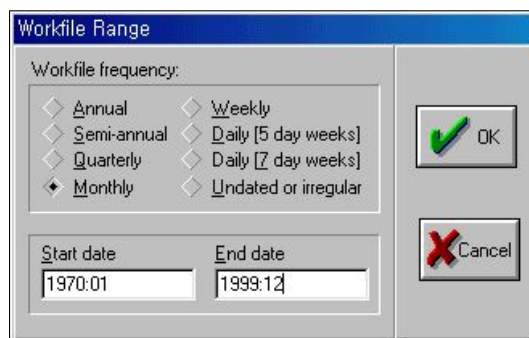
작업파일을 생성할 때는 데이터의 주기를 지정해 주어야 한다. EViews는 연간(Annual), 반년간(Semi-annual), 분기(Quarterly), 월(Monthly), 주(Weekly) 그리고 일일(5일 또는 7일) 데이터를 다룰 수 있는 주기 작업파일이 있다.

이런 주기 작업파일들은 데이터를 만들고 처리할 때는 가능한 한 월자료를 많이 이용한다.

1) 작업파일 생성(Creating a Workfile)

EViews에서 첫 번째 단계는 작업파일을 생성하는 것이다. 작업파일을 생성하는 방법은 File/New/Workfile을 선택하여 다음과 같은 대화상자에 필요한 정보를 입력하면 된다.

<그림 2.1> 작업파일(Workfile) 범위(Range)의 설정



적절한 주기를 선택하고 작업파일의 범위를 입력한다. Start date는 작업에서 이용할 제일 처음의 데이터나 관측치의 날짜이고 End date는 마지막 데이터나 관측치에 관한 것이다. 정확한 시작과 마지막 데이터를 몰라도 걱정할 필요는 없다. 만일 지정한 작업파일의 범위가 너무 작다면 나중에 확장시킬 수가 있다.

날짜를 입력하는 데는 다음과 같은 간단한 규칙이 있다.

- **연간(Annual):** 연도를 지정한다. 예를 들어, 1981, 1999, 81, 99 등이다. 21세기 이후의 연도는 연도의 전체 자리수를 모두 입력해야 한다.(즉, 2001, 2020 ...) 20세기의 연도는 2자리나 4자리 단위(즉, 99 또는 1999) 모두 이용이 가능하다
- **분기(Quarterly):** 연도 뒤에 콜론(:), 세미콜론(;), 마침표(.) 또는 쉼표(,)를 쓴 후 분기 숫자를 입력한다. 예를 들면 1992:1, 99:4과 같이 입력하면 된다.

- 월(Monthly): 분기와 마찬가지로 연도 뒤에 콜론, 세미콜론, 마침표 또는 쉼표를 쓰고 월 숫자를 입력한다. 예를 들어, 1970:1, 1999:11과 같다.
- 주와 일일(Weekly and daily): 기본적으로는 <월 : 일 : 연도>순으로 입력한다.

분기, 월, 주, 일일 데이터에서도 연도만 입력해도 EViews에서는 자동으로 처음과 마지막 관측치를 결정하여 준다.

작업파일에 필요한 형식을 모두 입력한 후 OK를 클릭하면 다음과 같은 작업파일 창을 볼 수 있는데, <그림 2.2>는 월 데이터로서 1970년 1월부터 1999년 12월까지를 지정했음을 알 수 있다. 아직까지 작업파일을 저장하지 않았기 때문에 작업파일의 이름은 없다.

<그림 2.2> 작업파일(Workfile)의 생성

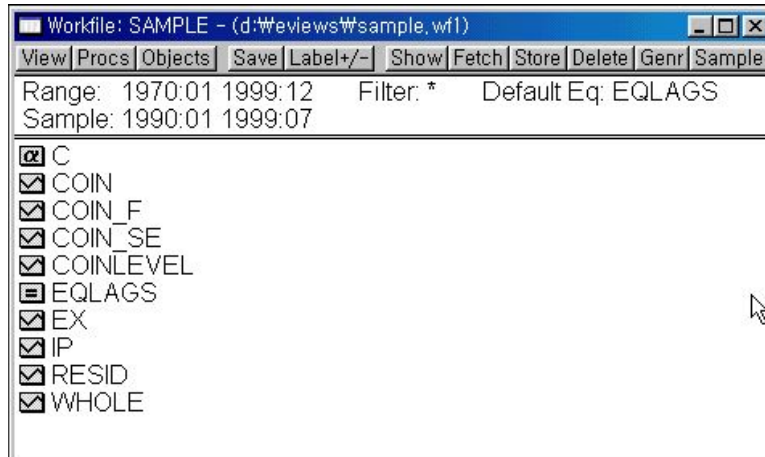


새로 생긴 작업파일에는 두 개의 아이콘이 있다. 이 아이콘은 모든 작업파일에 포함되어 있는 개체로서 C는 벡터 계수를 RESID는 잔차 계열을 의미하는데 왼쪽의 작은 아이콘은 개체의 형태를 나타내는 것으로 벡터 계수는 α 로 잔차계열은 시계열의 작은 그래프로 표시된다.

2) 작업파일 창(The Workfile Window)

작업파일과 개체를 생성한 후의 작업파일 창은 다음과 같다.

<그림 2.3> 작업파일의 생성후의 작업파일 창



작업파일 창의 제목 줄에는 ‘Workfile:’이라고 되어 있고 그 옆에 작업파일의 이름이 나온다. 만일 작업파일이 저장되지 않았다면 ‘UNTITLED’라고 되어 있을 것이다. 작업파일이 디스크에 저장이 되었다면 그 작업파일의 이름과 경로가 표시된다.(SAMPLE-(d:\weviews\sample.wf1))

제목 줄의 바로 아래에는 몇 개의 버튼으로 구성된 툴바가 있다. 이 버튼을 이용하면 작업파일의 기능들을 쉽게 사용할 수 있다.

툴바의 아래에 두 줄의 상태 표시가 되어 있는데 이것은 작업파일의 범위와 현재의 작업파일 표본(계산 및 통계적 응용을 하는데 이용되는 데이터의 범위), 필터(작업파일 창에 표시하기 위한 부분개체들을 선택할 때 이용), 그리고 기본식(최근의 추정식이나 사용식)이 표시되어 있다.

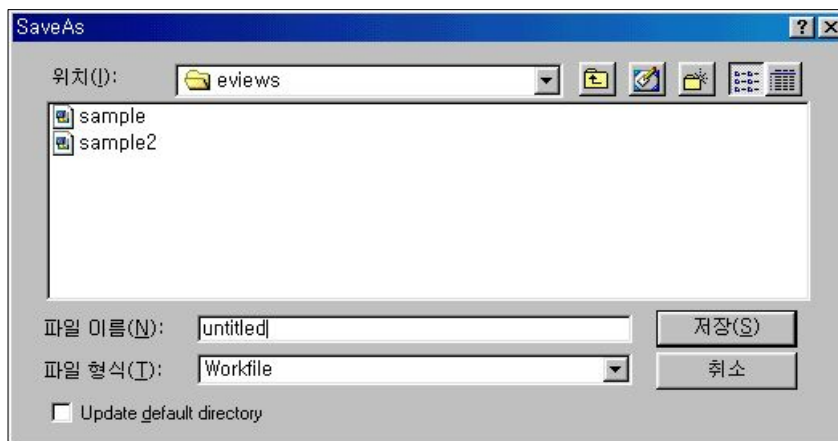
이런 것들은 그 표시들을 두 번 클릭하고 관련정보를 대화상자에 입력하면 범위, 표본 그리고 필터를 바꿀 수 있다. 등식 라벨을 두 번 클릭하면 그 등식이 열린다.

마지막으로, 작업파일 내용을 볼 수 있다. 개체의 이름과 아이콘이 이 화면에 모두 나열되어 있다.

3) 작업파일 저장하기(Saving Workfiles)

작업파일을 후에 다시 사용하기 위해서는 이름을 지정하고 저장해야 한다. 작업파일의 툴바에 있는 저장(Save)을 눌러 작업파일을 디스크에 저장한다. 또한 메인 메뉴에서 File/Save as... 또는 File/Save...를 선택하여 저장할 수도 있다. 저장할 때의 윈도우 대화상자는 다음과 같다.

<그림 2.4> 작업파일 저장



앞의 그림에서 ‘위치(I)’이라고 표시된 곳에 작업파일을 저장할 디렉토리를 지정하여 준다. 원하는 디렉토리를 찾기 위해서는 콤보 박스를 눌러서 찾으면 된다. 이 화면에서 디렉토리 이름을 클릭하면 그 디렉토리아에 있는 서브디렉토리나 파일들이 모두 나열된다. 만일 위의 화면의 디렉토리에 저장을 하고 싶다면 ‘파일 이름(N)’ 이라는 곳에 작업파일의 이름을 입력한 후 오른쪽의 ‘저장(S)’을 누른다. 이제 작업파일은 지정한 이름으로 저장되고 확장자는 .WF1이 된다.

일단 작업파일을 저장하면 툴바의 Save나 메인메뉴의 ‘File/Save...’를 이용해서 최신 정보나 수정된 것을 계속해서 저장할 수 있다. EViews는 작업파일을 저장한 이름으로 사용하다가 종료하면 만일 작업파일의 내용이 바뀌면 바뀐 내용을 디스크에 저장할 것인지의 여부를 묻게 된다. 다른 윈도우 소프트웨어에서처럼 File/Save As...를 이용해서 새로운 이름으로 저장할 수 있다.

일반적으로 버전 3에서 저장된 파일은 그 이전의 버전으로 읽을 수가

있다. 그러나 버전 3에서 새롭게 만들어진 개체들이 있을 때는 작업파일에서 제거된다. 그러므로 EViews의 구 버전으로 작업하거나 저장할 때는 주의를 해야만 한다.

4) 작업파일 불러오기(Loading Workfiles)

전에 작업했던 파일을 불러올 때는 File/Open/Workfile...을 이용한다.

File/Open/Workfile...을 선택하면 파일을 불러오는 대화상자가 열린다. 불러올 파일이 있는 디렉토리를 찾아서 그 파일이름을 두 번 클릭하면 된다. 그러면 작업파일 창이 열리고 그 파일 안에 있는 모든 개체들을 사용할 수 있게 된다.

작업파일을 불러올 때 편리하도록 EViews는 가장 최근에 작업한 10개의 목록을 파일 메뉴의 제일 아래쪽에 보여준다. 여기서 원하는 것을 클릭하면 그 파일이 열린다.

버전 3에서는 구 버전의 파일들을 모두 읽을 수가 있다.

5) 저장과 불러오기의 선택사항(Save and Load Options)

File/Open...과 File/Save As...의 대화상자를 입력하는 데에는 추가적으로 몇 가지의 선택사항이 있다.

① 기본 디렉토리 설정

모든 EViews 파일 대화상자는 기본 디렉토리의 내용을 보여주는 것으로 시작한다. 항상 EViews의 상태 줄에 보여지는 것에서 기본 디렉토리를 설정할 수 있다. 기본 디렉토리는 처음에는 EViews 프로그램이 들어 있는 디렉토리가 설정되어 있지만 이것은 이용자가 언제든지 바꿀 수 있다.

기본 디렉토리를 바꿀때는 File/Open...을 이용하거나 메인 메뉴의 File/Save As...를 이용하여 새로운 디렉토리를 찾은 후 대화상자의 기본 디렉토리 변경 상자(Updated default directory)에 표시를 하면 된다. 이렇게 한 후 파일을 저장하거나 불러오기를 하면 새로 지정한 디렉토리가 처음에 열리게 된다.

② MicroTSP 파일 이용하기

MicroTSP 형식의 작업파일도 읽고 쓰기를 할 수가 있다. 파일 형식 (Files of Type)과 저장할 형식(Save as Type)을 이용하여 DOS와 매킨토시 MicroTSP파일 작업을 할 수가 있다. 콤보 박스의 화살표를 클릭하고 Old Dos Workfile 이나 Old Mac Workfile 중에서 적절한 것을 선택한다. 그러나 만일 MicroTSP 형식으로 저장하기를 선택한다면 기본적으로 계열 데이터만 저장될 것이다.

6) 작업파일 범위 조정(Resizing Workfiles)

작업파일에 데이터를 추가하거나 시작 데이터 전이나 마지막 데이터 이후에 데이터를 예측하고자 할 때가 있다. 또 작업파일의 시작이나 마지막 데이터에서 필요없는 데이터를 제거해야 할 경우도 있다.

작업파일의 크기를 바꾸기 위해서 Procs/Change workfile Range...를 선택하고 대화상자에서 작업파일의 start date와 end date를 입력한다. 만일 입력한 데이터가 원래의 작업파일의 범위를 모두 포함하고 있다면 EViews는 추가 메시지 없이 작업파일을 확장시킨다. 만일 작업파일을 모두 포함하지 않는 범위를 입력하면 EViews는 데이터를 잃을지도 모른다는 메시지를 나타내고 그 실행을 할 것인지를 확인하게 한다.

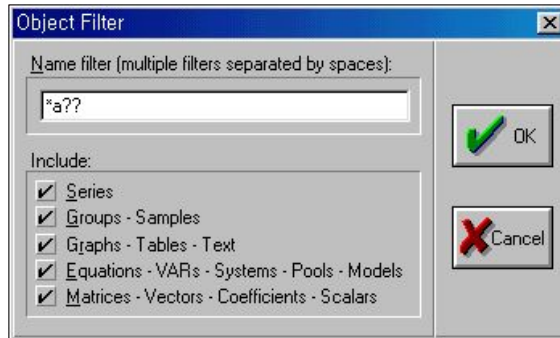
7) 작업파일 화면표시 바꾸기

① 화면 필터(Display Filter)

개체가 많이 있는 작업파일에서 작업할 때는 작업파일 창에서 특별히 원하는 개체로 이동하기가 쉽지 않다. 이것은 화면 필터를 이용하면 작업파일 창에서 부분개체들만을 표시해 주도록 할 수 있다. 이 부분개체들의 집합은 개체의 형태나 이름에 따라 정의할 수 있다.

View/Display Filter...를 선택하거나 작업파일 창에서 필터 부분을 더블 클릭하면 다음의 화면이 나타난다.

<그림 2.5> 화면 필터 대화상자



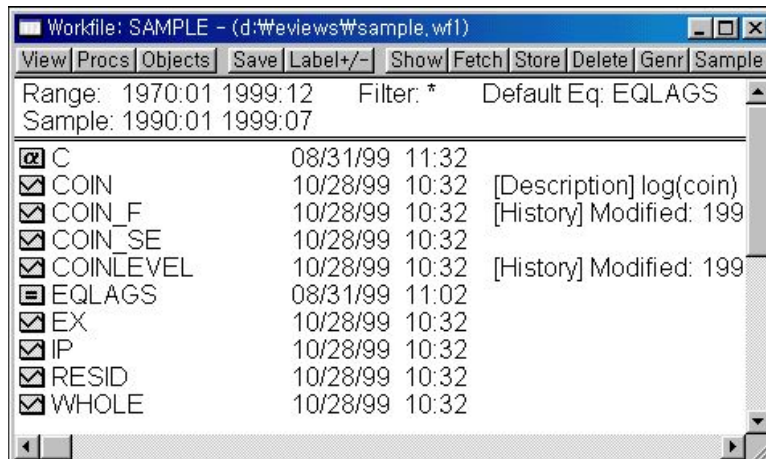
위의 대화상자는 두 부분으로 구성되어 있는데 편집상자(공백부분)에서는 하나 이상의 문자를 입력할 수 있는데 이 때 범용 문자인 '*'(여러 개 문자와 대응)나 '?'(하나의 문자만 대응)를 포함해서 사용할 수 있다. 편집상자 아래에는 EViews 개체들의 다양한 형태에 대응되는 체크 박스들이 있다. 편집상태에 입력된 것과 대응되는 이름을 가진 지정한 형태의 개체들만을 보여주게 된다.

기본 문자열이 '*'이면 지정한 형태의 모든 개체들을 표시한다. 만일 'x*'를 입력했다면 x로 시작하는 이름을 가진 개체들이 작업파일에 표시된다. 또 'x?y'를 입력했다면 문자 x로 시작하고 y로 끝나며 중간에 하나의 문자가 들어 있는 형태로 된 이름의 개체들만 표시된다. 'x* y* *z'라고 입력하면 x, y로 시작되는 이름을 가진 모든 개체와 z로 끝나는 이름을 가진 모든 개체를 표시하게 된다. 좀 더 복잡한 표현도 있는데 '??y* *z*'와 같이 입력한다면 앞에 두 개의 문자다음에 y로 시작하는 모든 이름의 개체와 이름 중에 z를 포함하고 있는 모든 개체들이 표시된다.

② 화면표시 주석(Display Comments)

현재의 개체들에 대한 추가정보를 보기 위하여 기본 작업파일의 표시를 바꿀 수 있다. View/Display Comments(Label+-)를 선택하면 표준작업파일 형식과 개체에 대한 설명을 한 라벨 뿐만 아니라 개체가 생성, 수정된 날짜에 관한 정보들을 보여준다.

<그림 2.6> 화면표시 주석



③ 문자형식 표시(Display Letter Format)

작업파일의 툴바에서 View/Name Display...를 선택하면 문자를 작업파일에서 표시할 때 대문자(Uppercase) 또는 소문자(Lowercase)의 사용을 지정할 수 있다. 기본값은 소문자이다.

2.2 개체(Object)

EViews에서 정보는 개체에 저장된다. 각각의 개체는 특별한 분석분야에 관련된 정보들의 집합이다. 특별한 개념과 연관된 개체들은 이름이 있는데 이 이름은 분석주제를 구분하는데 사용된다. 예를 들면, 계열개체는 특별한 변수들이 관측치와 관련된 정보의 집합이다. 등식 개체는 변수들의 집합사이의 관계와 관련된 정보의 집합이다.

각 형태의 개체들을 연결하는 것은 보기(view)와 절차(procedure)인데 이것은 개체에 포함된 정보를 서로 연결해 주는데 사용된다. 개체에 포함된 데이터 형태에 대한 이런 보기와 절차들의 연결은 EViews에서 'object oriented'라고 한다.

이것은 어떤 작업을 할 때 정보들을 체계화함으로써 EViews에서의 작업을 쉽게 해 준다. 등식 개체는 추정에 관련된 모든 정보를 포함하고 있기 때문에 다양한 등식들을 다른 등식 개체를 선택하여 옮길 수도 있다.

결과를 검증하고 가설을 세워 그에 대한 검정을 하고 또 어떤 시점을 예측할 수가 있다. 하나의 개체가 데이터와 결과를 전체적 집합으로 작업할 때 이용되므로 작업이 훨씬 쉬워진다.

1) 개체 데이터(Object Data)

각 개체는 많은 정보를 포함하고 있다. 예를 들어 계열, 행렬, 벡터 그리고 스칼라 개체들은 대부분 수치 정보를 갖고 있다. 반대로 등식이나 방정식들은 추정에 이용된 기본적 데이터들에 대한 참조뿐만 아니라 등식과 방정식의 설정 그리고 추정 결과에 대한 완전한 정보를 갖고 있다. 그래프와 표는 수치, 문자 그리고 정형화된 정보를 갖는다.

개체들은 많은 종류의 데이터를 포함하고 있기 때문에 다른 개체를 가지고 다른 방식으로 작업을 하고자 할 때도 있는데 예를 들어 한 계열에서 데이터들의 합을 계산하거나 등식의 결과에 근거하여 예측을 하고자 할 때도 있다. EViews에서는 보기(views)와 절차(procedures)라는 일반적 도구를 이용하여 이러한 차이들을 모두 인식하고 개체의 데이터로 작업을 한다.

2) 개체 보기(Object Views)

개체에서 데이터를 보는 데에는 여러 가지 방법이 있다. 보기(View)는 개체에서 데이터를 보는 여러 가지 방법을 제공하는 표와 그래프 창이다.

예를 들면, 계열개체는 스프레드시트 형태로 보여지는데 여기서 데이터, 선그래프, 막대그래프, 히스트그램과 통계량, 상관도 등도 알 수 있고 간단한 가설검정과 통계량도 계산할 수 있다.

등식 개체는 지정한 등식, 추정결과를 포함한 산출결과, 실제값-예측값-잔차 그래프, 추정계수의 공분산 행렬을 포함한 공분산, 많은 설정과 모수 검정에 대한 것이 있다.

개체보기는 개체 창에 표시된다. 한 개체에 대해서는 하나의 창만이 열리고 하나의 창에는 한 번에 하나의 개체보기만을 표시한다. 개체 창의 툴바에 있는 View 메뉴나 메인 메뉴를 이용하여 개체보기를 바꿀 수 있다.

이 개체보기에서 기억해야 할 가장 중요한 것은 개체보기를 변경한다고 해서 개체에 있는 데이터가 변하지는 않는다는 것이다. 개체에 남아있는 원래의 데이터는 똑같고 다만 데이터를 표시하는 방법만 변하는 것이다.

3) 개체 절차(Object Procedures)

대부분의 EViews 개체는 절차(Procedures or procs)를 가진다. 개체보기와 같이 절차도 개체 창에 표와 그래프를 나타내 준다. 하지만 개체보기와는 다르게 개체내에 있는 데이터 자체나 다른 개체에 있는 데이터가 변하게 된다.

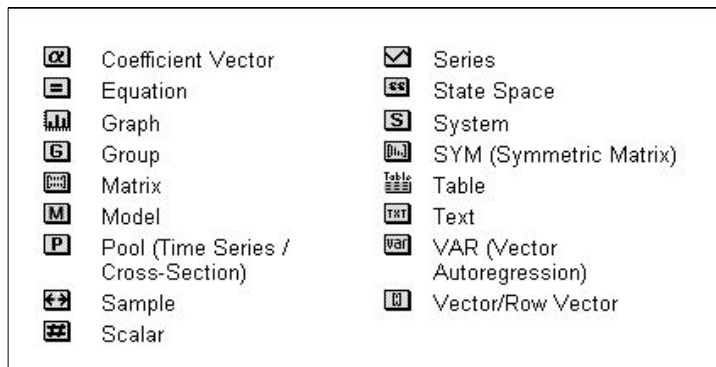
절차(Procedures)를 이용하면 새로운 개체를 생성할 수 있다. 예로서, 계열개체는 평활화나 시계열을 계절조정하는 절차가 있어서 평활화된 또는 조정된 데이터의 새로운 계열을 생성한다. 등식 개체는 잔차, 예측값 추정식에서 얻은 예측치를 포함하는 새로운 계열을 생성하는 절차를 포함하고 있다.

이런 절차(Procedures)들은 EViews 메인 메뉴나 개체 툴바의 Procs 메뉴에서 선택할 수 있다.

4) 개체 형태(Object Types)

EViews에서 가장 기본적인 개체는 계열과 등식 개체이다. 그러나 다른 형태의 개체가 많이 있고 각 개체들은 각자의 기능이 있다. 대부분의 개체들은 개체 상자 창에 각각의 특징적인 아이콘으로 표시된다.

<그림 2.7> 개체 형태



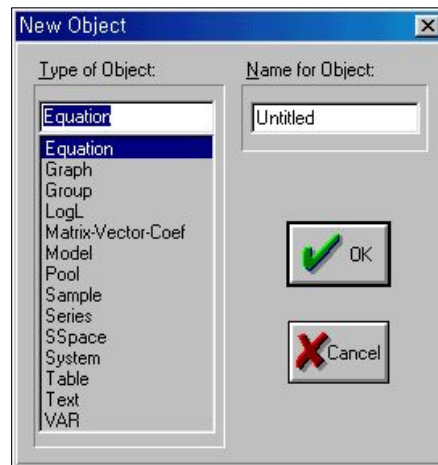
위의 아이콘들은 모두 개체인데 하지만 개체 상자는 그 자체를 다른 개체 상자 안에 넣을 수 없기 때문에 아이콘이 없다. 또 작업파일과 데이터베이스도 다른 작업파일이나 데이터베이스 안에 넣을 수 없기 때문에 아이콘이 없다.

5) 개체 생성, 선택과 열기

① 개체 생성(Creating Objects)

개체를 생성할 때는 먼저 작업파일 상자를 열어 작업파일 창이 활성화되어 있어야 한다. 다음으로 메인 메뉴에서 Objects/New Object...를 선택한다. 작업파일을 만들거나 불러오기 전에는 이러한 선택이 필요없다. 메뉴항목에서 클릭하면 다음과 같은 대화상자를 볼 수 있다.

<그림 2.8> 새로운 개체 생성



여기서 원하는 개체를 선택할 수 있고 경우에 따라 이름을 지정하고 OK를 클릭해야 한다. 어떤 개체들은 선택한 개체를 좀 더 자세히 설명하기 위하여 또 다른 대화상자가 열리기도 한다. 그러나 대부분의 개체들은 개체창이 바로 열린다.

예를 들어, 등식개체(Equation Object)를 선택하면 추가정보를 위한 대화상자가 열리고, 또 계열(Series)을 선택하고 OK를 클릭하면 이름이 없는 스프레드시트를 보여주는 개체 창(계열 창)이 열리게 된다.

<그림 2.9> 새로운 계열 선택

	1	2	3	4	5
Last updated: 10/28/99 - 10:58					
1970:01	NA	NA	NA	NA	NA
1970:07	NA	NA	NA	NA	NA
1971:01	NA	NA	NA	NA	NA
1971:07	NA	NA	NA	NA	NA
1972:01	NA	NA	NA	NA	NA
1972:07	NA	NA	NA	NA	NA
1973:01	NA	NA	NA	NA	NA
1973:07	NA	NA	NA	NA	NA
1974:01	NA	NA	NA	NA	NA
1974:07	NA	NA	NA	NA	NA
1975:01	NA	NA	NA	NA	NA
1975:07	NA	NA	NA	NA	NA
1976:01	NA	NA	NA	NA	NA
1976:07					

② 개체 선택(Selecting Objects)

항상 새로운 개체를 생성해야 하는 것은 아니고 이미 있던 개체를 가지고 작업을 할 때도 있다. EViews에서 기본적인 운용 중에 하나는 작업파일 내용에서 하나 이상의 개체를 선택하는 것이다.

개체를 선택하는 가장 쉬운 방법은 표준 윈도우 창에서 필요시에 원하는 항목을 찾아서 선택하면 된다. 여러 개의 항목을 선택해야 한다면 항목들을 선택하기 전에 화면 필터기를 이용하는 것이 더 편리할 것이다.

작업파일의 툴바에 있는 View는 편리한 단축키를 제공한다.

- **선택 모두해제(Deselect All)**: 현재 선택한 것을 해제
- **모두 선택(Select All(except C-RESID))**: 작업파일에서 계수 C와 잔차 RESID만 빼고 모든 개체를 선택
- **필터로 선택(Select By Filter...)**: 앞에서 설명했던 화면필터기와 같은 방식으로 개체의 이름과 형태의 다양한 조합으로 개체를 선택할 수 있다.

③ 개체 열기(Opening Objects)

일단 개체를 선택하면 그 개체를 열거나 선택한 개체가 포함된 새로운 개체를 만들고자 할 때는 선택된 개체의 아무 곳에서도 더블 클릭을 하면 된다.

또 다중 그래프나 계열을 선택하고 더블 클릭하면 새로운 개체(그룹, 등식, VAR, 그래프)를 생성하거나 열 수 있거나 또는 그 자체의 창에 각각의 선택한 개체를 표시해 주는 부메뉴가 나타난다.

다중 그래프를 선택하고 더블 클릭하거나 View/Open as One Window 를 선택하면 여러 개의 그래프가 하나로 묶여져서 하나의 창에 나타난다.

④ 개체 보기(Showing Objects)

개체를 선택하거나 여는 또 다른 방법으로는 보기(show) 항목을 이용하는 것이다. 툴바에서 Show를 누르거나 메뉴에서 Quick/Show...를 선택하고 개체의 이름을 입력한다.

Show는 개체를 처음 선택했을 때와 같이 실행되어 선택한 개체를 열어준다. 만일 대화상자에 하나의 개체이름을 입력하면 EViews는 개체이름을 더블 클릭한 것과 같이 그 개체를 연다. 또 여러 개의 이름을 입력하면 그 결과들을 보여주기 위해 필요시에는 새로운 개체를 생성하여 하나의 창에 열어 준다.

6) 개체 창(Object Window)

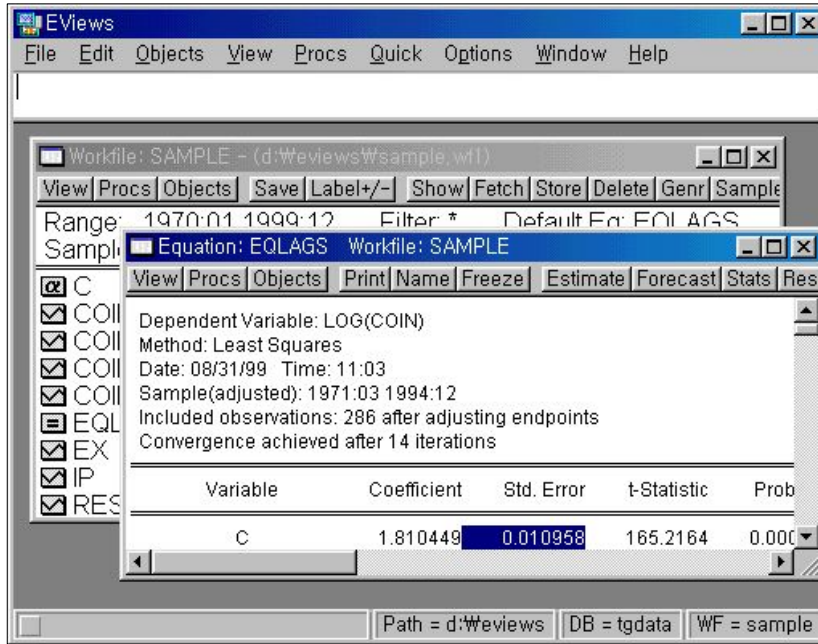
이제까지는 개체 창을 이용한 개체의 생성이나 열기에 관한 절차에 대해서 알아보았다. 개체 창은 개체나 개체상자를 열 때 나타나는 창이다. 개체 창은 개체보거나 개체에 대한 절차에 대한 결과도 포함한다.

EViews의 중요한 성질 중에 하나는 동시에 여러 개의 개체 창을 나타낼 수 있다는 것이다. 이런 개체 창들을 다루는 것은 책상에서 여러 장의 종이를 흩어놓고서 작업하는 것과 같다.

① 개체 창의 구성요소

다음의 전형적인 개체 창을 보자.

<그림 2.10> 개체 창



여기서 EQLAGS의 등식 창을 볼 수 있다. 첫째 이것은 종료, 크기 재 조절, 최소화, 최대화 그리고 가로와 세로방향으로 움직여서 이동할 수 있는 표준 창이다. 다른 윈도우 응용프로그램에서와 마찬가지로 제목 줄 위나 창안의 아무 곳을 클릭함으로써 개체 창을 활성화시킨다. 개체 창을 활성화시킨다는 것은 그 창에서 작업을 하겠다는 것이다.

둘째 개체 창의 제목 줄은 개체 형태, 이름 그리고 개체 상자(이 경우에는 SAMPLE 작업파일)로 구분된다.

마지막으로 창의 제일 위에는 몇 개의 버튼으로 구성된 툴바가 있다. 이 버튼은 자주 사용하는 메뉴 항목을 쉽게 이용할 수 있다. 이런 툴바는 개체의 형태에 따라 바뀐다. 즉 계열 개체는 등식, 그룹이나 VAR 개체와는 다른 툴바를 가진다.

그러나 다음의 몇 가지 버튼은 모든 개체의 툴바에서 볼 수 있는 것들이다.

- **View:** 개체 창에 표시할 보기(View)를 바꿀 수 있다. 선택은 개체의 형태에 따라 다르다.
- **Procs:** 메뉴에서 개체에 대해 가능한 절차(Procedure)를 실행할 수 있다.
- **Object:** 개체들을 다룰 수 있다. 개체를 디스크에 저장, 이름 지정, 삭제, 복사 또는 출력을 할 수 있다.
- **Print:** 현재 보이는 개체를 출력한다.(창 내용)
- **Name:** 개체에 이름을 지정하거나 새이름으로 바꿀 수 있다.
- **Freeze:** 새로운 개체 그래프, 표, 현재 보기의 텍스트 개체를 고정한다.

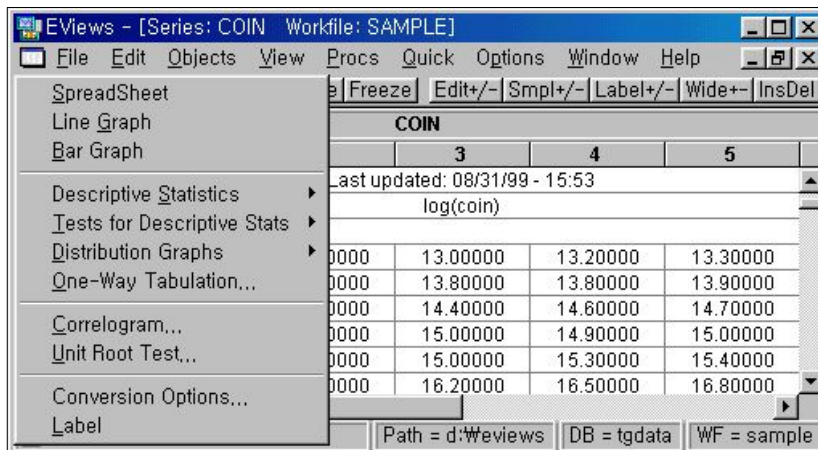
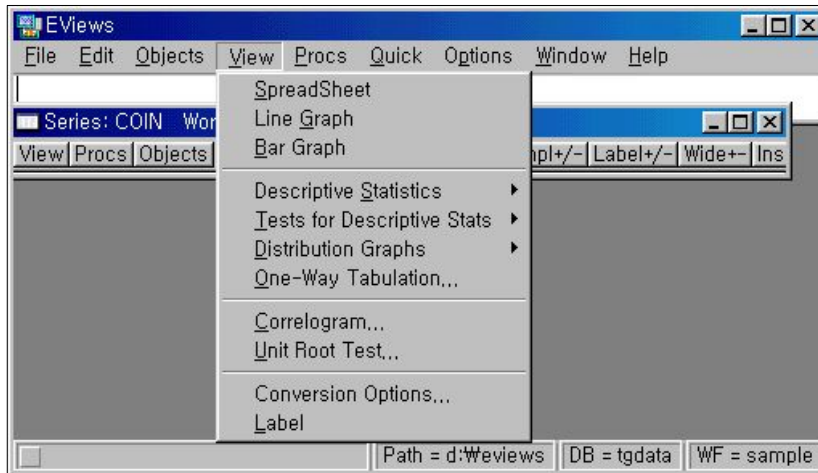
② 메뉴와 개체 툴바

위에서 보았듯이 툴바는 메뉴 명령을 실행할 수 있는 단축키를 제공한다. 다음은 이것과 관련해서 특히 강조할 만한 점들이다.

- 툴바는 메뉴 항목의 단축키를 제공하는 것이므로 메뉴에서도 이 툴바의 명령을 찾을 수 있다.
이것은 만일 창이 툴바에 있는 모든 버튼을 나타낼 만큼 충분히 크지 않다면 메뉴에서 찾으려 한다는 것을 의미한다. 이 때 버튼이 모두 보이게 창을 키우든지 아니면 메뉴에서 직접 명령어를 실행시킬 수 있다.
- 툴바와 메뉴는 모두 개체 형태에 따라 다르다. 특히, View와 Procs 메뉴의 내용은 개체 형태(계열, 등식, 그룹 등)를 반영하기 위해 항상 변한다.

툴바와 메뉴는 개체에 따라 아주 다양하다. 예를 들어, View와 Procs의 부메뉴는 모든 개체 형태에 따라 다르다. 계열 창이 활성화되어 있는 경우 메뉴는 계열보기와 계열절차를 실행할 수 있다. 반대로, 그룹 창이 활성화되어 있을 때의 View와 Procs는 그룹개체와 관련 있는 실행을 할 수 있는 항목들이 나타나게 된다.

<그림 2.11> 그룹 창이 활성화되어 있을 때의 View메뉴



위의 그림은 계열 창이 활성화되어 있는 경우의 View 툴바와 View 메뉴를 비교한 것이다. 윗쪽은 EViews 창이 있고 메인 메뉴에 있는 View를 클릭 했을 때의 그림이다.(COIN 계열 창이 활성화되어 있다.) 아래쪽은 계열 창에서 계열 툴바에 있는 View를 눌렀을 때 나타나는 그림이다. 두 개의 실행은 같기 때문에 실행시 나타나는 부메뉴의 내용도 같다.

View와 Procs 메뉴와 달리 보통 개체 메뉴는 개체에 따라 다르지 않다. 그러나 개체상자 창(작업파일 또는 데이터베이스 창)이 활성화되어 있는 경우는 예외이다. 이 경우에는 툴바에 있는 Objects를 클릭하거나 메뉴에서 Objects를 선택하면 상자에서 개체를 다룰 수 있는 메뉴항목을 볼 수 있다.

7) 개체 작업(Working with Objects)

① 개체 이름 지정하기(Naming Objects)

개체에는 이름이 있을 수도 있고 없을 수도 있다. 개체에 이름을 정해 줄 때는 작업파일의 디렉토리가 표시되고 그 개체는 작업파일이 저장될 때 그 작업파일의 부분으로서 저장된다.

만일 개체의 결과를 유지하고 싶다면 그 개체에 이름을 정해주어야 한다. 개체에 이름을 정해주지 않으면 그것은 제목 없음(UNTITLED)으로 표시된다. 이름이 정해지지 않은 개체는 작업파일과 함께 저장되지 않기 때문에 작업파일을 종료하면 메모리에서 그 내용이 없어져서 삭제된다.

개체에 이름 또는 새이름을 정해 줄 때는 먼저 개체 창을 열어서 그 개체의 아이콘을 더블클릭하거나 작업파일 툴바에 있는 Show를 클릭한다. 그리고 개체 이름을 입력한다. 다음에는 개체 창의 Name을 클릭하고 이름을 입력한다.(16자까지 가능) 선택적으로 표시되는 이름은 표나 그래프에서 참고할 때도 이용된다. 만일 표시한 이름이 없는 경우에는 개체이름을 이용한다.

또 작업파일 창에서 Objects/Rename Selected...를 선택하면 개체를 새 이름으로 지정할 수 있다. 이 방법은 개체를 열 때 처음 저장했던 것을 유지할 때 이용한다.

다음은 예약어로서 계열의 이름으로 사용할 수 없다.

ABS, ACOS, AR, ASIN, C, CON, CNORM, COEF, COS, D, DLOG, DNORM, ELSE, ENDIF, EXP, LOG, LOGIT, LPT1, LPT2, MA, NA, NAND, PLD, RESID, RND, SAR, SIN, SMA, SQR, THEN.

EViews에서는 계열이나 다른 개체에 이름을 정할 때 대문자와 소문자를 모두 사용할 수는 있지만 소문자로 입력한 경우도 모두 대문자로 인식된다. 그러므로 'SALES', 'sales', 'sAles'가 EViews에서는 모두 같은 개체로 취급된다.

그러나 그래프의 범례나 라벨 등과 같은 문자 정보를 입력할 때는 대문자와 소문자를 구분하여 사용할 수 있다.

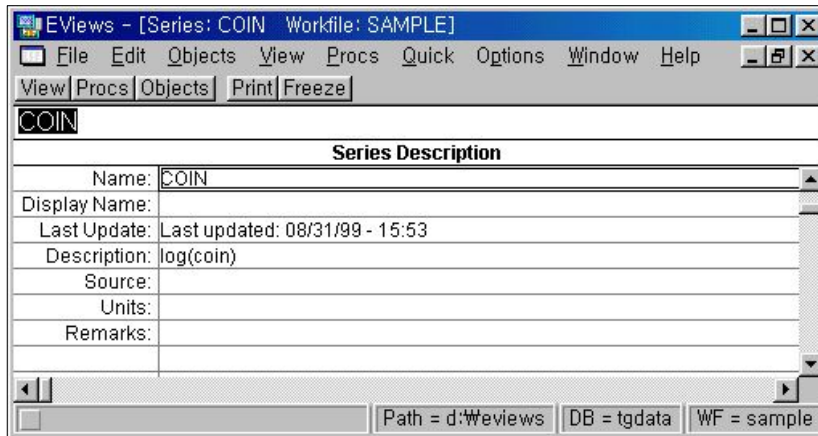
기본적으로 EViews에서 이름이 없는 개체는 하나만 존재할 수 있다. (하나의 계열, 하나의 등식 등) 만일 이미 이름 없는 개체가 존재하는데 또 이름이 없는 새로운 개체를 생성하면 원래의 개체는 이름을 지정해 주

어야 한다. 원래의 개체에 이름을 정해주지 않으면 EViews는 원래의 이름이 없는 개체를 새로운 개체로 대체시킨다. 그러면 원래의 개체는 저장되지 않는다. 이름을 지정하지 않는 방법을 이용하고 싶으면 작업하는 도중 내내 이름이 없는 개체들을 유지시킬 수 있도록 하는 기능을 이용할 수도 있지만 그래도 작업파일과 함께 저장하고 싶은 것들은 이름을 지정해 주어야 한다.

② 개체 라벨(Labeling Objects)

앞에서 설명한 이름 지정하기 외에도 EViews에서는 개체에 대한 설명이나 주석을 표시할 수 있는 라벨 필드가 있다. 이 필드를 보려면 개체 창에서 View/Label을 선택한다.

<그림 2.12> 개체 라벨



위의 그림은 수정되지 않은 개체의 라벨 보기이다. 개체를 수정할 때마다 라벨 표시의 맨 밑에 있는 History 필드에 수정내용이 추가로 기록된다.

위에서 Last Update 필드를 제외한 다른 모든 곳은 수정할 수 있다. 수정할 때는 원하는 필드의 내용이 있는 부분을 클릭하면 된다. 모든 필드는 행이 하나만 있고 Remarks와 History 필드는 여러 개의 행이 있다. 이 두 개의 필드에 새로운 행을 추가하고자 할 때는 엔터를 누르면 된다.

필드 설명란은 EViews 데이터베이스에서 저장된 개체를 찾고자 할 때 가장 유용하다. 데이터베이스에서 이 필드에 있는 문자를 이용해서 찾을 수 있다.

③ 개체 복사(Copying Objects)

개체의 정보를 똑같이 만드는 것으로는 복사와 고정외의 두 가지 방법이 있다.

활성화된 작업파일에서 개체를 열고 개체 툴바 또는 메인 메뉴에서 Objects/Copy Object...를 선택하면 원래의 개체와 똑같은 이름없는 (UNTITLED) 개체를 생성한다. 새로 생성된 개체는 원래의 개체 보기 (View)와 절차(Procedure)에 대한 정보를 모두 갖고 있고 분석에도 이용할 수 있다.

또한 작업파일 창에서도 개체를 복사할 수 있다. Objects/Copy Selected...를 클릭하여 source와 destination개체의 이름을 지정하면 된다.

여기서 복사(Copy)란 다른 많은 기능처럼 매우 일반적이고 유용한 기능으로 작업파일과 데이터베이스 상호간에 복사를 할 수 있다.

④ 복사하여 붙이기(Copy-and-Pasting Objects)

복사 명령은 같은 작업파일내에 개체를 만드는 것이다. 두 개의 작업파일이 동시에 있을 때는 복사하여 붙이기(Copy-and-Paste)명령을 이용해야 한다.

복사할 원본 작업파일에서 복사할 부분을 선택한 후 메인 메뉴에서 Edit/Copy를 선택한다.

복사 대상 작업파일의 제목 줄을 클릭하고 메인 메뉴에서 Edit/Paste를 선택한다. 그러면 원본 작업파일에서 선택하여 복사한 것과 이름이 똑같은 개체가 만들어진다.

원본 작업파일과 복사 대상 작업파일의 주기가 다르면 주기 변환은(가능하면) 복사 대상 작업파일에 붙이기 전에 계열 개체에 주기변환이 일어난다.

⑤ 개체 고정(Freezing Objects)

하나의 개체로부터 똑같은 정보를 만드는 또 다른 방법은 개체를 고정(Freeze)하는 것이다. Objects/Freeze Output을 클릭하거나 개체 툴바에서 Freeze를 누르면 현재 보이는 표나 그래프 개체가 복사되어 만들어진다.

화면을 고정하는 것은 화면을 복사하여 그것이 독립적인 개체로 되는데 원래의 개체를 삭제하여도 남아 있게 된다. 고정된 화면은 원래의 개체에 있던 현재의 상태를 보여주는 것이 아니라 Freeze를 누를 당시의 화면을 보여주는 것이다. 예를 들어, 계열의 스프레드시트를 고정하였다면 새로운 표 개체의 화면을 볼 수 있고 계열의 그래프 화면을 고정하였다면 새로운 그래프 개체를 볼 수 있는 것이다.

개체를 고정함으로써 만들어진 표와 그래프는 발표나 보고서용으로 편집할 수 있다. 작업파일의 데이터 표본이 변하거나 데이터 자체가 변해도 고정된 화면은 변하지 않는다.

⑥ 개체 지우기(Deleting Objects)

작업파일에서 개체를 지울 때는 먼저 작업파일 내용에서 지우고자 하는 개체를 선택한다. 지울 개체를 선택한 후에 Delete나 작업파일의 툴바에서 Objects/Delete Selected를 클릭한다. 이 때 EViews는 정말로 개체를 삭제할 것인지를 묻는 메시지를 나타낸다.

⑦ 개체 출력(Printing Objects)

작업파일 창에서 개체를 선택한 후 메인 메뉴의 View/Print Selected를 선택하면 개체를 출력할 수 있다.

현재 보이는 개체를 출력하기 위해서 개체 창의 툴바에 있는 Print를 누른다. 또한 메인 메뉴에서 File/Print도 선택할 수 있다.

작업파일 툴바에서 View/Print Selected를 선택하여 작업파일 창에서 선택한 개체들을 동시에 출력할 수도 있다

⑧ 개체 저장(Storing Objects)

데이터를 디스크에 저장하는 데는 세 가지의 방법이 있다. 작업파일을 저장하는 것에 대한 것은 앞에서 이미 설명하였고, 이 때의 작업파일은 확장자가 .WF1으로서 하나의 파일로 작업파일 안에 있는 모든 개체가 함께 저장된다. 또 개체를 개별적으로 각 개체의 데이터 뱅크 파일 안에 저장할 수도 있다. 이것은 다른 작업파일에서도 불러서 이용할 수가 있다. 마지막으로 개체를 EViews 데이터베이스 파일로 저장하는 것이다.

데이터 뱅크와 데이터베이스에 개체를 저장하는 방법에 대해서는 데이터베이스부분에서 다루기로 하고 여기서는 하나의 개체로 작업할 때 개체의 툴바나 메뉴에서 Objects/Store to DB...를 클릭하여 그 개체를 데이터 뱅크나 데이터베이스 파일로 보낼 수 있다는 것만 알아두자.

하나 이상의 개체를 저장할 때는 작업파일 창에서 개체들을 선택하고 작업파일 툴바나 메뉴에서 Objects/Store selected to DB...를 누르면 선택한 개체들을 저장할 수 있다.

⑨ 개체 불러오기(Fetching Objects)

이미 저장한 항목들을 데이터뱅크나 데이터베이스에서 불러올 수가 있다. 데이터를 가지고 작업하는 일반적인 방법 중에 하나는 작업파일을 생성하고 이미 저장했던 데이터를 필요할 때 작업파일로 불러오는 것이다.

작업파일로 개체를 불러올 때에는 작업파일 메뉴나 툴바에서 Objects/Fetch from DB...를 선택한다. 이 때 불러올 개체, 디렉토리, 사용할 데이터베이스 위치 등의 정보를 입력하기 위한 대화상자가 나타난다.

⑩ 개체 수정(Updating Objects)

작업을 수정하는 것은 개체를 불러오는 것과 같지만 작업파일에 나타난 개체들이 필요하다. 작업파일에서 개체를 수정할 때는 작업파일 창에서 수정할 개체를 선택하고 작업파일 메뉴나 툴바에서 Objects/Updates from DB...를 클릭하면 불러오기 대화상자가 열린다. 그러나, 불러올 개체는 이미 입력되어 있다. 디렉토리나 데이터베이스 위치를 지정하고 OK를 누른다.

2.3 명령어

- 새로운 작업파일(Workfile)을 생성할 때는 명령어
⇒ `workfile sample1`
새로운 작업파일 test1에 대한 작업범위를 지정하는 대화상자가 나타난다.
- 이름이 없고(untitled) 작업 범위가 1985년 1월에서 1999년 12월까지 새로운 월별 작업파일을 생성할 때의 명령어
⇒ `create m 1985:01 1999:12`
- 작업파일을 저장할 때는 `save` 명령어 다음에 저장할 작업파일의 이름을 쓴다.
⇒ `save sample1`
기본 디렉토리에 `sample1`라는 이름으로 작업파일을 저장한다.
⇒ `save d:\evIEWS\sample.wf1`
`c:\evIEWS` 디렉토리에 `sample`라는 이름으로 작업파일을 저장한다.

3. 기본적인 데이터 처리

데이터 입력, 읽기, 편집, 계산, 생성 등은 대부분 데이터 분석의 기초가 된다. 그러므로 EViews에서 하는 작업은 대부분 데이터를 가지고 하는 것이다. EViews는 이러한 데이터를 가능한 한 손쉽게 간단하게 처리해주는 데이터 처리 도구들을 제공한다.

이 장에서는 EViews에서 데이터 처리를 위하여 가장 중요한 부분인 데이터 입·출력과 편집에 대해 먼저 설명한다.

데이터를 분석하기 위해서는 세 가지의 기본적인 방법이 있는데 두 가지는 일반적인 데이터 개체로서 계열(series)과 그룹(group)이고 나머지는 표본(sample)을 이용하는 것이다.

3.1 데이터 읽기(Importing Data)

EViews는 매우 다양한 형태의 데이터로 작업을 하게 된다. DRI Basic Economics와 같은 외부 데이터베이스나 다른 프로그램에서 다운로드 했거나 직접 작업하여 만든 스프레드시트나 텍스트 파일도 있을 것이다.

EViews에서 외부의 데이터를 읽는 방법은 몇 가지가 있다. 다음에서는 스프레드시트, 텍스트 파일 또는 출력된 형태들의 데이터를 계열이나 그룹 개체로 읽어들이는 데이터 읽기의 기본적인 것을 설명하고자 한다.

1) 데이터 입력(Entering data)

크기가 작은 데이터 셋이라면 키보드로 직접 데이터를 입력할 수 있다.

- 먼저 데이터를 입력할 스프레드시트 창을 연다. 메인 메뉴에서 Quick/Empty Group(Edit Series)를 선택하여 제목이 없는(UNTITLED) 그룹 창을 연다.

<그림 3.1> 데이터 입력

obs					
obs	COIN				
1970:01					
1970:02					
1970:03					
1970:04					

- 그 다음은 계열을 생성하여 이름을 지정한다. 위 방향 화살표를 움직여서 왼쪽의 열에 두 번째 'obs'라벨이 보이도록 한다. 두 번째 obs라벨의 오른쪽 행에 원하는 계열의 이름을 입력하거나 수정하면 된다.
두 번째 계열을 생성하여 이름을 지정한다. 위 방향 화살표를 움직여서 왼쪽의 열에 두 번째 행에 'obs'라벨이 보이도록 한다. 두 번째 obs 라벨의 오른쪽 행에 원하는 계열의 이름을 입력하거나 수정하면 된다. 두 번째 obs 라벨의 옆에 있는 셀을 클릭하고 첫 번째 계열의 이름을 입력한다. 여기서는 명령어 창에 'COIN'이라고 입력하였다.(셀에 있던 이름은 명령어 창에 입력할 때 변하게 된다.) 엔터를 누른다.
- 만일 이미 입력했던 계열의 이름을 수정하고자 할 때는 그 계열이 있는 셀을 선택하고 이름을 다시 입력한 후 엔터를 누른다. 이 때 다시 입력한 이름으로 바꿀것이지를 확인하는 메시지가 나온다.
- 데이터를 입력할 때는 원하는 셀을 선택하고 숫자를 입력한다. 숫자를 입력하고 엔터를 누르면 다음 셀로 이동한다. 스프레드시트에서 커서를 움직여서 사용할 수도 있다.
- 데이터 입력이 끝나면 그룹 창을 종료한다. 먼저 Name 버튼을 눌러 그룹의 이름을 지정한다. 만일 그룹을 유지하고 싶지 않다면 EViews가 삭제할 것인지를 묻는 메시지에 Yes 하면 된다.

2) 복사 및 붙이기(Copying and Pasting)

윈도우 클립보드는 EViews와 EViews 또는 EViews와 다른 응용프로그램 사이에서 데이터를 쉽게 이동시킬 수 있는 편리한 방법이다. 윈도우의 복사하여 붙이기 기능이 있는 엑셀이나 다른 응용프로그램에서 흔히 쓰이는 기능이다.

① 응용프로그램에서 복사하기

다음은 엑셀 스프레드시트를 이용한 예제이다. 그러나 기본원칙은 다른 윈도우 응용프로그램에도 적용된다.

엑셀 스프레드시트(a:\data\sample.xls)에 있는 동행종합지수(COIN)와 산업생산지수(IP) 데이터를 EViews로 읽어오기를 해 보자.

<그림 3.2> 엑셀 스프레드시트의 자료(COIN과 IP)

	A	B	C	D	E	F	G
1	OBS	COIN	IP				
2	1970:01	12.6	3.4				
3	1970:02	12.8	3.2				
4	1970:03	13	3.8				
5	1970:04	13.2	3.9				
6	1970:05	13.3	4.1				
7	1970:06	13.4	4				
8	1970:07	13.6	4				
9	1970:08	13.6	3.9				
10	1970:09	13.8	3.9				
11	1970:10	13.8	4.2				
12	1970:11	13.9	4.1				

먼저 EViews로 가져갈 데이터를 선택한다. 열의 위 부분에 EViews에서 변수로 사용할 COIN과 IP가 있으므로 이 부분도 선택한다. EViews는 날짜로 된 데이터를 인식하고 예제에서는 월별 주기의 작업파일을 생성할 것이므로 날짜가 있는 열은 복사할 필요가 없다. 열 이름 B를 클릭하여 C까지 끌고 가면 스프레드시트의 두 열이 선택된다.

<그림 3.3> 자료의 선택

	A	B	C	D	E	F	G
1	OBS	COIN	IP				
2	1970:01	12.6	3.4				
3	1970:02	12.8	3.2				
4	1970:03	13	3.8				
5	1970:04	13.2	3.9				
6	1970:05	13.3	4.1				
7	1970:06	13.4	4				
8	1970:07	13.6	4				
9	1970:08	13.6	3.9				
10	1970:09	13.8	3.9				
11	1970:10	13.8	4.2				
12	1970:11	13.9	4.1				

선택한 데이터를 클립보드에 복사하기 위해서 Edit/Copy를 선택한다.

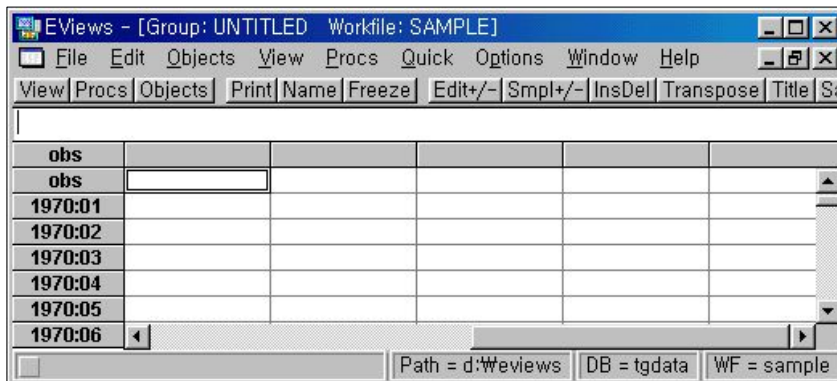
② 새로운 계열에 붙이기

EViews를 시작하고 새로운 작업파일을 만들거나 이미 있던 작업파일(a:\data\sample.wf1)을 불러 엑셀 스프레드시트에 있는 날짜(예제에서는 1970:01부터 1999:07까지)로 된 새로운 월 주기의 작업파일을 만든다. 표본은 엑셀에서 복사한 데이터들과 같도록 지정한다.

새로운 작업파일 또는 기존의 작업파일을 연 다음 Quick/Empty Group(Edit Series)를 선택한다. 스프레드시트는 편집상태로 열리기 때문에 Edit+/- 버튼은 누르지 않아도 된다.

여기서는, 1970:01부터 1999:12까지의 범위로 된 월 주기의 작업파일을 만들었다. EViews 스프레드시트의 첫 번째 행의 라벨은 1970:01이다. 복사한 계열이름을 붙이기 해야 하므로 스크롤 바에 있는 위 방향 화살표를 한 번 클릭하여 위쪽으로 이동한다.

<그림 3.4> 새로운 그룹 설정



위의 그림에서 커서를 왼쪽 상단의 셀 즉, 두 번째 obs의 바로 오른쪽 셀에 놓는다. 그리고 메인 메뉴에서 Edit/Paste+/- (툴바에 있는 Edit+/-가 아님)를 선택한다. 엑셀 스프레드시트에서 복사했던 데이터가 그대로 나타나게 된다.

<그림 3.5> EViews 그룹에 붙이기

The screenshot shows the EViews software interface. The title bar reads 'EViews - [Group: UNTITLED Workfile: SAMPLE]'. The menu bar includes 'File', 'Edit', 'Objects', 'View', 'Procs', 'Quick', 'Options', 'Window', and 'Help'. The toolbar contains buttons for 'View', 'Procs', 'Objects', 'Print', 'Name', 'Freeze', 'Edit+/-', 'Smpl+/-', 'InsDel', 'Transpose', 'Title', and 'Se'. The main window displays a group named 'COIN' with a table of data. The table has columns for 'obs', 'COIN', and 'IP'. The data rows are as follows:

obs	COIN	IP
1970:01	12.60000	3.400000
1970:02	12.80000	3.200000
1970:03	13.00000	3.800000
1970:04	13.20000	3.900000
1970:05	13.30000	4.100000
1970:06		

At the bottom of the window, there are status bars showing 'Path = d:\weviews', 'DB = tgdata', and 'WF = sample'.

여기서, 계열 COIN과 IP는 작업파일에서 만들어 졌고 어떤 EViews 절차에서도 사용할 수가 있다. 이제 그룹 창을 종료하고 두 계열을 유지한 채로 제목 없는 그룹을 삭제한다.

③ 기존 계열에 붙이기

클립보드에 있는 데이터를 Edit/Paste를 이용해서 이미 생성된 EViews 계열이나 그룹 스프레드시트로 불러 올 수도 있다. 이 때 주의할 점이 있다.

- 몇 개의 계열을 붙일 때는 먼저 계열이 들어 있는 그룹 창을 연다. 가장 쉬운 방법은 Show를 클릭하고 클립보드에 나타난 순서대로 계열의 이름을 입력하는 것이다. 또 다른 방법은 첫 번째 계열을 선택하여 제목 없는 그룹을 만든 후에 각각의 계열을 순서대로 선택하여 클릭하고 그것을 열기 위해 더블 클릭한다.
- 다음, 그룹 창은 편집상태여야 한다. 만약 편집상태가 아니면 Edit+/- 버튼을 눌러 편집상태로 만든다. 커서를 목표 셀에 놓고 Edit/Paste를 선택한다.
- 마지막으로, Edit+/-를 클릭하여 보호상태로 만든다.
- 계열 하나만을 붙이기 할 때는 계열 창이 편집 상태로 되어 있어야 하고, 계열은 하나의 열에 모두 보여지는 상태여야 한다. 계열이 넓은 화면으로 보여지는 상태라면 Smpl+/- 버튼을 누른다. 데이터를 Edit/Paste하고 데이터를 보호하기 위해 Edit+/-버튼을 누른다.

3) 파일에서 데이터 읽기

다른 프로그램에서 만들어진 파일에서도 직접 데이터를 읽을 수가 있다. 데이터는 표준 아스키 형태나 로터스(.WKS, .WK1, .WK3) 또는 엑셀

(.XLS) 스프레드시트 양식으로 되어 있는 것이다.

먼저, 데이터 내용을 받아들이기 위한 작업파일이 열려 있어야 한다.

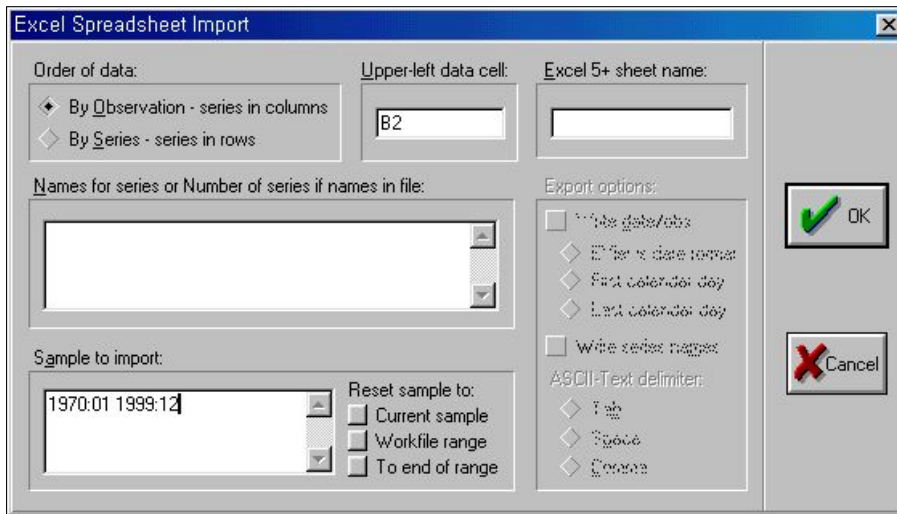
다음은 Procs/Import/Read Text-Lotus-Excel...을 클릭한다. 그러면 파일의 형식과 이름을 지정하는 대화상자가 열린다. 파일 형식을 선택하고 파일이 있는 디렉토리를 찾아서 그 파일을 더블 클릭한다. 그렇지 않으면 읽어 올 파일을 직접 입력한다.(디렉토리까지 모두 입력해야 함) 파일 형식을 지정하지 않으면 EViews가 자동적으로 아스키 파일로 처리한다. Open을 클릭한다.

데이터를 불러오기 위한 추가정보를 위한 대화상자가 열린다. 이 대화상자는 불러 올 파일이 스프레드시트인지 아니면 아스키 파일인지에 따라 매우 다르다.

① 엑셀파일 읽기

대화상자의 제목 줄은 읽고자 하는 파일의 형태를 나타낸다. 다음은 엑셀의 스프레드시트에서 데이터를 읽어올 때의 대화상자이다.

<그림 3.6> 엑셀의 스프레드시트에서 데이터 읽기



이제 대화상자를 채워 보자

- 먼저, 데이터가 어떤 순서로 되어 있는지를 지정해야 한다. By observation은 첫 번째 관측치에 해당하는 모든 데이터들 다음에 두 번째 관측치가 나오는 것이다. 그리고 By series란 첫 번째 변수에 대한 데이터가 있고 다음에 두 번째 변수에 대한 데이터가 정렬되어 있는 것이다. 'by rows'가 한 변수에 대한 모든

데이터가 행 방향으로 정렬되어 있는 것을 뜻하는 반면에 'by observation'은 변수들이 열 방향으로 정렬되어 있다는 것으로 해석하기도 한다.

예) 앞에서의 엑셀 예제(그림 3.2)에서는 각각의 계열이 열 방향으로 정렬되어 있기 때문에 'by observation'을 선택한다. 만일 엑셀 데이터 다음과 같이 COIN과 IP가 한 행에 모두 정렬되어 있다면 그 때는 'by series'로 읽어야 한다.

<그림 3.7> 엑셀 데이터 읽기(행기준)

	A	B	C	D	E	F
1	OBS	1970:01	1970:02	1970:03	1970:04	1970:05
2	COIN	12.6	12.8	13	13.2	13.3
3	IP	3.4	3.2	3.8	3.9	4.1
4						
5						

- 다음은 실제 데이터의 시작 셀의 위치를 지정해야 한다(이때 데이터의 변수명이나 설명에 대한 것은 포함하지 않음). 위의 예제에서는 B2가 된다.
- 편집상자에서 읽고자 하는 데이터의 변수명을 입력한다. EViews는 다음 행이나 열에 있는 셀에서 데이터를 읽기 때문에 비록 선택한 행들에 대한 데이터만을 읽을 때에도 행 또는 열(데이터가 입력되어 있는 상태에 따라 다름)에 대하여 각각의 이름을 입력해야 한다.
- 마지막으로, 읽고자 하는 데이터의 표본기간을 입력한다.
 처음 대화상자가 나타날 때는 EViews 편집상자에 현재의 작업파일 표본이 기본값으로 입력되어 있다. 이것을 사용자가 원하는 표본으로 수정하면 된다. 데이터 표본을 좀 더 쉽게 입력할 수 있도록 EViews는 자주 사용되는 값을 편집상자에서 수정할 수 있는 3개의 입력버튼을 제공하고 있다.
현재의 표본: 현재의 작업파일 표본을 사용
작업파일 범위: 작업파일의 전체 범위를 사용
마지막 범위까지: 현재 표본의 처음부터 작업파일 범위의 마지막까지 모두 사용
- 엑셀의 파일을 읽을 때는 데이터가 들어있는 시트의 이름을 입력할 수 있는 편집상자가 따로 있다. 만일 그 시트의 이름을 입력하지 않으면 엑셀의 첫번째 시트(sheet1)에서 데이터를 읽게 된다.
- 대화상자를 모두 입력하고 OK를 클릭하면 파일로부터 데이터를 읽어 계열을 생성한다.

※ 명령어를 이용한 엑셀파일 읽기

- 엑셀 파일(a:\data\sample.xls) 중 열기준인 데이터(그림 3.2)를 읽는 방법

⇒ read(b2, s=sample1) a:\data\sample.xls coin

read(c2, s=sample1) a:\data\sample.xls ip

여기에서 read명령 괄호안의 내용 중 b2, c2는 각 데이터의 시작 셀위치이며, s=sample1는 's=sheet1이름'을 의미한다.

그 다음으로 경로명을 지정하고 계열명을 입력한다

- 엑셀 파일(a:\data\sample.xls) 중 행기준인 데이터(그림 3.7)를 읽는 방법

⇒ read(b2, s=sample2, t) a:\data\sample.xls coin

read(c2, s=sample2, t) a:\data\sample.xls ip

여기에서 read명령 괄호안의 내용 중 t는 행기준(by series) 일 경우 꼭 사용해야 데이터를 옳게 읽을 수 있다.

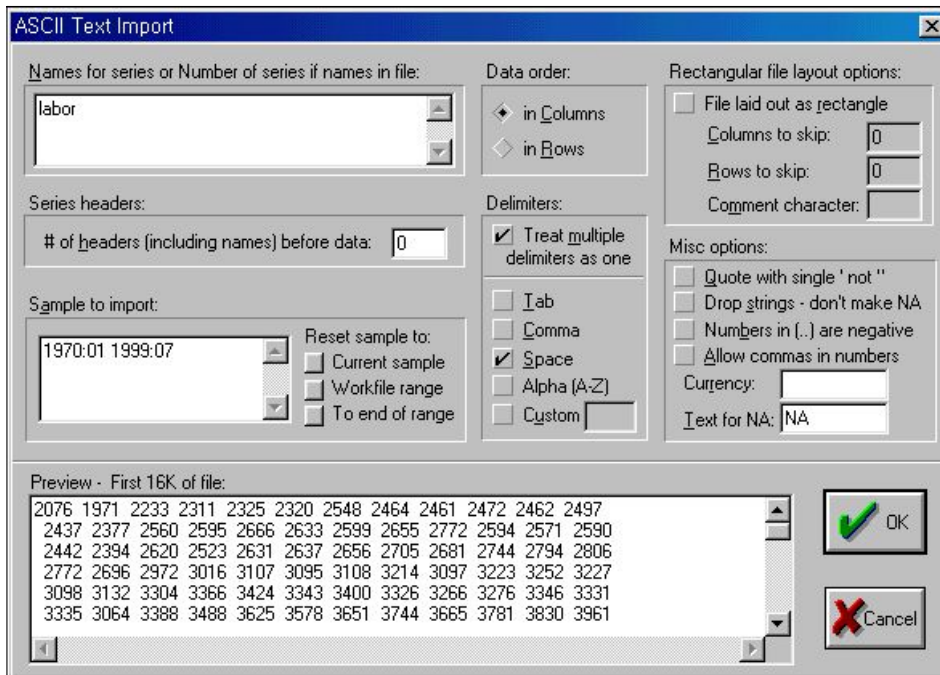
② ASCII파일 읽기

작업파일에서 아스키파일을 읽을 때는 툴바에서 Procs/Import/Read Text-Lotus-Excel...을 클릭하여 특정 아스키파일을 선택하면 “Ascii Text Import” 대화상자가 열린다. 아스키 파일에 대한 대화상자는 표준양식이 없기 때문에 스프레드시트의 대화상자보다 상당히 복잡하다. EViews는 여러 가지 형태의 아스키 파일들을 다룰 수 있는 선택범위들을 제공하고 있다.

(예제1)

아스키 파일(a:\data\labor.dat)을 EViews 작업파일에서 읽기 위해 툴바에서 Procs/Import/Read Text-Lotus-Excel...을 클릭하여 a:\data\labor.dat을 선택하면 다음의 대화상자가 나타난다.

<그림 3.8> 아스키 파일 읽기



앞의 대화상자에서 아래 부분에 표시된 것을 보자. 여기에 읽어들이고 싶은 파일의 처음 16Kbyte의 데이터를 보여주고 있다.

다음의 정보를 입력해야만 한다.

- **변수명 또는 계열의 개수:** 파일 안에 계열의 변수명이 없거나 계열의 변수명을 사용하지 않았다면 파일에 있는 순서대로 계열의 변수명을 공백으로 구분하여 입력해야 한다. 만일 파일 안에 데이터가 시작되기 전에 계열의 변수들이 있다면 이 변수들의 개수를 입력함으로써 그 변수를 사용할 수 있다.

괄호나 *, +, -, /, ^ 등과 같은 수학기호를 변수명으로 사용하는 것은 가급적 피하는 것이 좋다. EViews가 이러한 특수문자가 포함된 변수명을 읽어 들일 때는 자동적으로 밑줄이나 숫자로 대체하여 읽어 들인다. 예를 들어 변수명이 X(-3)이라면 EViews는 이것을 X_3_01로 읽어 들일 것이다. 만일 X_3_01이 이미 파일 안에 있는 변수명이라면 X_3_02와 같은 식으로 읽어 들인다.

EViews가 특수문자를 변환시키지 못하거나 계열이 아닌 데이터로 사용되어 변수명을 대체시키지 못할 경우에는 SER01, SER02 와 같이 변수명을 지정한다.

만일 파일안에 있는 변수명이 작업파일에 있는 것과 똑같은 경우에는 이미 있던 데이터들을 덮어 쓰게 되므로 대화상자에 변수명을 입력할 때는 유의해야 한다.

- **데이터 순서:** 파일 안에 있는 데이터가 어떤 식으로 정렬되어 있는지를 선택해야 한다. 각각의 계열이 세로방향으로 되어 있다면 'in Columns'를 선택해야 한다. 또 데이터가 가로방향으로 정렬되어 있다면 'in Rows'를 선택해야 한다.
- **데이터 표본:** 파일 안에 있는 데이터의 표본을 지정해야 한다. 대화상자에는 현재의 작업파일의 표본이 나타나지만 직접 키보드를 이용하거나 버튼을 이용하여 수정할 수 있다. 여기서 지정한 표본은 불러올 데이터에만 해당되는 것이지 다른 작업파일에는 영향을 주지 않는다.

지정한 표본만큼 데이터는 모두 읽어 오게 되는데 여기에는 몇 가지 규칙이 있다.

1. 지정한 표본만큼 모든 데이터를 대응시킨다. 데이터가 지정한 표본보다 많을 경우에도 표본은 변하지 않는다.
2. 지정한 표본보다 데이터가 너무 적어서 데이터가 없는 표본은 'NA'로 할당시켜 준다.
3. 지정한 표본만큼만 데이터를 읽어 오고 불러올 파일에 남아있는 데이터는 무시한다.

이 외에도 아스키파일을 불러 올 때 다음의 선택사항들을 이용할 수 있다.

데이터를 불러 올 파일의 처음 몇 줄을 보면서 대화상자에서 불러올 형식을 지정해 줄 수 있다. 하지만 이런 방식은 파일의 처음 몇 줄에만 한정적이므로 적절하지 않을 수도 있다.

가) 구분자(Delimiters)

구분자란 파일에서 데이터를 구분할 때 사용하는 문자이다. 적절한 항목을 선택하여 복수의 구분자를 지정할 수 있다. 탭(Tab), 콤마(Comma), 공백(Space) 등은 이미 다른 프로그램에서도 많이 사용하는 것들이다. 그리고 알파(Alpha)는 알파벳의 26개 중 어느 것이라도 구분자로 사용할 수 있다.

선택사항에서 구분자를 지정하지 않고 일반(Custom) 항목을 선택하면 이용자가 구분자로 사용할 기호를 선택할 수 있다. 예를 들어 슬래쉬(/)를 구분자로 사용하고 싶다면 일반항목을 선택한 후 편집상자에 '/'를 입력하면 된다. 여기서 하나 이상의 문자를 입력하게 되면 각각의 문자를 구분자로 인식하게 된다. 예를 들어 편집상자에 '//를 입력했다면 '//가 아닌

‘/’를 구분자로 인식하게 된다. ‘//’는 두 개의 구분자로 인식된다.

EViews에서는 복수의 구분자를 하나의 구분자로 처리할 수 있는 기능이 있다. 예를 들어 ‘,’를 구분자로 하고 ‘복수 구분자를 하나로 취급(Treat multiple delimiters as one)’하는 선택사항을 택하면 ‘,’를 하나의 구분자로 인식하게 된다. 이 선택사항을 해제하면 missing value에 있는 ‘,’를 두 개의 구분자로 인식하게 된다.

나) 사각형 형식 파일(Rectangular file layout options)

사각형 형식으로 된 아스키 파일을 다룰 때는 대화상자의 오른쪽 위에 있는 사각형 형식 파일(‘File laid out as rectangle’)을 선택한다. 만일 파일이 사각형 형식으로 되어 있으면 한 줄 한 줄 넘어갈 때마다 새로운 데이터나 계열로 인식하고 줄 단위로 데이터를 읽게 된다. 사각형 선택사항을 해제하면 전체 데이터를 구분자나 줄 바꿈 표시로 구분되는 하나의 긴 문자열로 인식한다.

파일이 사각형 형식으로 되어 있으면 한 라인에서 읽을 수 있는 데이터를 알 수 있기 때문에 아스키를 다루기가 훨씬 쉽다. 데이터가 사각형 형식으로 되어 있지 않으면 이용자가 데이터 계열 수를 정확하게 입력하여야 한다. 예를 들어 사각형 형식이 아닌 세로방향의 데이터가 4개의 계열을 갖고 있다고 가정해 보자. 이 파일을 읽어 올 때 EViews는 새로운 줄을 무시하고 매번 4개의 데이터를 읽은 후 다시 새로운 데이터를 읽는다.

데이터 파일이 사각형 형식이라면 건너 뛴 행 또는 열을 지정할 수도 있다. 예를 들어, 사각형 형식의 파일을 읽을 때 건너 뛴 행(‘Rows to skip’)을 3으로 입력하면 데이터의 처음 3행을 건너뛰어 네 번째 줄부터 읽어 오게 된다. 이것은 처음의 몇 행이나 몇 열만을 건너 뛰어 데이터를 읽을 수 있고 파일의 중간에서는 할 수 없다.

다) 머릿글 지정(Series header)

파일로부터 데이터를 읽기 전에 계열이름으로 얼마만큼의 셀(행 또는 열)을 이용했는가를 지정하는 것이다. 셀을 지정하는 방식은 데이터 파일이 사각형 형식인지의 여부에 따라 다르다.

데이터 파일이 사각형 형식이면 행(데이터는 열방향으로 정렬) 또는 열(데이터는 행방향으로 정렬)로 지정하게 된다. 예를 들어 다음과 같은 데

이더 파일이 있다면

LABEL	P85	P87	RMT85	CS82	SS82	S82	ME84	REV84	REG	CL
1	33	27	288	13	24	49	2135	2836	1	1
2	19	15	139	14	11	41	957	2035	1	1
1	26	20	196	12	14	41	1530	6030	1	1

여기에서 보면 계열의 이름과 데이터 사이에 한 줄의 간격이 있다. 이 경우 머릿글을 지정할 때 계열의 이름 행과 한 줄의 간격까지 포함해서 2로 지정해 주어야 한다. 만일 이름 행과 데이터 행 사이에 한 줄의 간격이 없다면 머릿글 지정은 1로 한다.

사각형 형식이 아닌 데이터 파일일 때는 구분자로 구분된 공백의 수를 지정해 준다. 다음과 같은 데이터의 예를 보자.

FTP	UEMP	MAN	LIC	GR	CLEA	WM	NMAN	GOV	HE
WE	HOM	ACC	ASR						
260.35	11.0	455.5	178.15	215.98	93.4	558724.	538.1	133.9	2.98
117.18	8.60	39.17	306.18						
269.80	7.0	480.2	156.41	180.48	88.5	538584.	547.6	137.6	3.89
134.02	8.90	40.27	315.16						

위의 데이터는 열 방향으로 정렬되어 있지만 10개의 계열은 첫 번째 줄에 그리고 나머지 4개 계열은 두 번째 줄에 입력되어 각각의 데이터도 두 행에 걸쳐서 입력되어 있다.

이 경우 위의 데이터 파일을 14개의 계열을 가진 사각형 형식의 데이터로서 머릿글을 2로 지정하는 오류를 범하기 쉽다. 만일 이처럼 지정한다면 EViews는 처음 행에서 계열의 이름들을 읽어 들인 후 2번째 행은 건너뛰고 다시 3번째 행부터 차례대로 데이터를 읽어 들인다. 이렇게 되면 처음 행에서 10개 계열의 이름은 제대로 읽지만 두 번째 행의 4개 계열의 이름은 읽지 못하고 계열의 이름이 SER01-SER04로 대체된다. 데이터도 잘못 읽어들이는데 EViews는 각각의 행을 새로운 데이터의 시작으로 인식하기 때문에 위의 데이터에서 다섯 번째 계열인 'GR'의 구성 데이터는 「215.9800, NA, 180.4800, NA」가 된다.

위의 데이터 파일을 제대로 읽어 오기 위해서는 사각형 형식의 선택사항을 해제하고 머릿글 지정을 1로 해야 한다. EViews는 구분자와 줄바꿈 표시에 의해 구분되어 있는 처음 14개의 값들을 왼쪽에서 오른쪽 방향으로 읽어 계열 이름에 대응시킨다. 이것이 머릿글 지정을 1로 한 부분이다. 다음의 14개 데이터는 14개 계열에 대한 첫 번째 데이터가 되는 형식으로 데이터를 읽어 들인다.

라) 기타 선택사항

- **인용부호(')(Quote with single ' not "):** EViews에서는 겹따옴표(" ") 표시 안에 있는 것을 하나의 문자열, 그렇지 않으면 숫자로 취급한다. 이 선택사항은 겹따옴표 대신 홑따옴표(' ')를 사용하여 문자열을 인식하는 것이다. EViews에서는 문자열을 지원하는 기능이 없기 때문에 겹따옴표 안에 있는 것은 missing value로 처리하고 그렇지 않으면 숫자로 인식한다.
- **문자열 뛰어넘기-NA를 만들지 않음(Drop strings-don't make NA):** EViews는 문자열을 지원하는 기능이 없으므로 구분자나 숫자가 아닌 어떤 것을 입력할 때 원칙적으로는 missing value로 처리된다. 예를 들어 '10b'와 '90:4'는 모두 missing value로 처리된다.(알파벳문자 'b'와 특수기호 ':'가 구분자로 사용되지 않았을 경우). 이 선택사항은 이러한 문자열을 'NA'로 읽어 들이지 않고 그냥 무시하고 넘어가는 것이다.
이 선택사항을 지정하게 되면 문자열로 된 계열의 이름도 영향을 받아 주어진 계열의 이름이 아닌 EViews가 지정하는 형식의 SER01, SER02 ...로 계열의 이름이 입력되게 된다. 만일 이 선택사항을 지정하고 파일에 들어있는 계열의 이름을 그대로 사용하고자 한다면 대화상자에 계열의 이름을 모두 입력해 주어야 한다.
만일 어떠한 문자를 missing value로 처리하기 위해 누락문자('Text for NA')의 편집상자에 입력하였다면 데이터를 읽는 중에도 그 문자를 무시하지 않고 missing value로 처리한다.
- **()안의 숫자는 음수(Numbers in (..) are negative):** EViews에서 ()는 문자열로 취급된다. 그러나 이 선택사항을 지정하면 괄호안에 있는 숫자는 음수로서 취급되고 이에 따라 데이터를 읽게 된다.
- **숫자에 콤마(,)를 허용(Allow commas in numbers):** 만일 콤마를 구분자로 지정하지 않으면 일반적으로는 문자열로 취급한다. 예를 들어, 1,000은 NA(만일 문자열 뛰어넘기 선택사항을 지정했다면 무시) 또는 1 과 0(콤마를 구분자로 지정했을 경우)으로 읽어오게 된다. 여기서 숫자에 콤마를 허용하는 선택사항을 지정한다면 1,000은 숫자 1000으로 입력된다.
- **통화(Currency):** 이것은 통화의 기호를 사용할 수 있게 해주는 것이다. 예로 \$10은 \$를 구분자로 지정하지 않으면 원칙적으로는 문자열('NA'나 무시할 수도 있음)로 취급된다. 여기서 만일 통화(Currency)란에 '\$'를 입력해 주면 \$10은 숫자 10으로 입력된다.
통화기호는 숫자의 처음이나 끝에 나타나고 중간에 나타나지는 않는다. 통화기호란에 하나 이상의 기호를 입력하면 입력된 것은 모두 통화코드로

서 인식된다. 통화기호를 사용할 때에는 유의해야 한다. 만일 일본의 통화 단위인 엔(¥)을 대문자 ‘Y’로 표시하였다면 통화란에 입력할 때 소문자 y가 아닌 대문자 Y를 입력해야 한다.

- **누락문자(Text for NA):** 이것은 누락된 데이터를 한 문자로 지정해 주는 것으로 기본값은 ‘NA’이다. 이 선택사항은 데이터를 읽을 때 특별한 문자나 숫자 즉 “.”, “-99” 등과 같은 것을 missing value로 처리하고 싶을 때에도 사용할 수 있다.

missing value로 처리되는 문자는 반드시 하나만 입력해야 한다. 누락문자(Text for NA)란에 입력된 전체 문자를 하나의 missing value 코드로 인식한다.

(예제2)

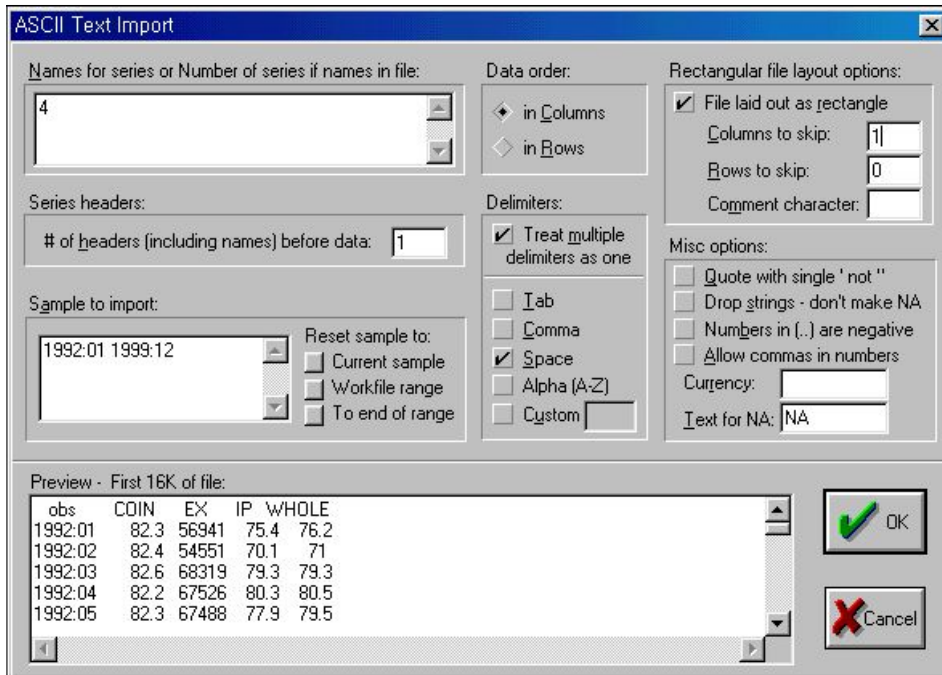
다음의 아스키 파일 예(a:\data\example.dat)를 가지고 설명해 보자. 예제의 데이터는 다음과 같다.

obs	COIN	EX	IP	WHOLE
1992:01	82.3	56941	75.4	76.2
1992:02	82.4	54551	70.1	71.0
1992:03	82.6	68319	79.3	79.3
1992:04	82.2	67526	80.3	80.5
1992:05	82.3	67488	77.9	79.5
1992:06	82.8	74956	77.9	78.0
1992:07	83.2	67501	78.5	76.6
1992:08	83.0	65499	74.3	74.4
1992:09	82.7	70334	73.8	79.9
1992:10	82.8	75644	79.9	81.3
1992:11	82.9	70593	78.3	80.9
1992:12	83.0	74697	79.3	84.6

위의 데이터는 연도와 4개의 계열이 열방향으로 정렬되어 있으며 각각은 공백으로 구분되어 있다.

범위(Range)가 1970년 1월부터 1999년 12월까지인 EViews 작업파일을 생성하여 툴바에서 Procs/Import/Read Text-Lotus-Excel...을 클릭하여 a:\data\example.dat을 선택하면 다음의 대화상자가 나타난다.

<그림 3.9> 아스키 파일 읽기

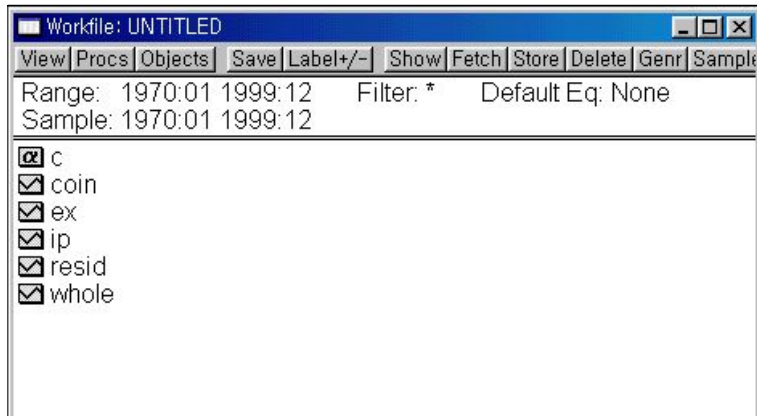


여기서 고려해야 할 사항은 첫 열(column)인 연도 뛰어넘기와 머릿글 지정(Series header), 데이터 읽을 표본기간 등 3가지에 주의하여 데이터를 읽어야 한다.

그러면 데이터를 읽는 방법을 살펴보자

- ㉠ 첫 번째 문자열로 된 것은 제외하고 4개의 계열 이름을 입력한다
(여기서는 계열의 개수를 입력함)
- ㉡ 데이터 파일에 머릿글이 있으므로 머릿글 지정은 1로 입력한다.
- ㉢ 구분자는 공백으로 표시한다.
- ㉣ 연도를 뛰어넘기 위해 사각형 형식 파일(file laid out as rectangle)에서 'Column to skip'에 1을 입력한다.
- ㉤ 읽을 자료가 1992년 1월부터 1999년 12월까지이므로 'Sample to import'란에 "1992:01 1999:12"로 바꿔주면 된다.

<그림 3.10> 아스키파일 읽은 후의 작업파일



※ 명령어를 이용한 아스키파일 읽기

- read(Options) 경로명 및 파일명 작업파일의 계열명
- Options
 - ㉠ 파일 형태 Options
 - t=dat, txt : 아스키 파일
 - t=wk1, wk3 : Lotus 파일
 - t=xls : Excel 파일
 - ㉡ 아스키 텍스트 파일의 Options
 - na=text : na를 text로 지정한다. 기본값은 “NA”
 - t : 계열로 읽음(행렬은 행과 열을 바꿈).
기본값은 열로 계열을 관측치로 읽음
 - d=t : tab을 구분자로 지정
 - d=c : 콤마를 구분자로 지정
 - d=s : 공백을 구분자로 지정

d=a : 알파벳을 구분자로 지정
 custom=symbol : symbol/문자를 구분자로 지정
 mult : 다중 구분자를 하나로 지정
 name : 파일에 계열이름을 지정
 label=integer : 머릿글행과 데이터 사이의 행간격. 'name' 선택사항과 함께 쓰임
 rect : 데이터파일의 모양을 사각형으로 처리
 skipcol=integer : 건너뛴 열 수 지정. 'rect' 선택사항과 함께 쓰임
 skiprow=integer : 건너뛴 행 수 지정. 'rect' 선택사항과 함께 쓰임
 comment=symbol : 주석 부호로 사용할 문자/symbol을 지정. 주석 부호의 오른쪽에 있는 것은 모두 무시된다.
 singlequote : 홑(') 인용부호를 이용
 dropstrings : NA를 문자열로 처리하지 않음.
 negparen : 괄호안의 숫자를 음수로 처리
 allowcomma : 숫자에 포함된逗마를 허용
 currency=symbol : 통화기호로 사용할 부호/문자를 지정

- (예제1)의 아스키파일(a:\data\labor.dat)를 명령어를 사용하여 import 하는 방법
 ⇒ read(t=text, d=s, norect) a:\data\labor.dat labor
- (예제1)의 아스키파일(a:\data\example.dat)를 명령어를 사용하여 import 하는 방법
 ⇒ smpl 1992:01 1999:12
 read(t=text, d=s, rect, label=1, skipcol=1) a:\data\example.dat 4

3.2 데이터 내보내기(Exporting Data)

EViews에서 작업결과로 생긴 데이터들을 다른 곳에서도 응용할 수 있는 여러 가지 방법이 있다.

1) 복사와 붙이기(Copying and Pasting)

작업파일의 스프레드시트나 통계 결과표에서 복사하고자 하는 부분을 마우스로 클릭하여 끈다. Edit/Copy...를 클릭하여 다른 응용프로그램에 데이터를 넣는다. 데이터를 붙여 넣을 때 숫자들을 화면에 보이는 그대로 붙여 넣을 것인지(formatted copy) 아니면 숫자의 전체 사이즈로 붙여 넣을지를(unformatted copy) 묻는 대화상자가 나타난다.

스프레드시트 둘레의 회색부분을 클릭하면 셀이 포함된 전체 행이나 열을 선택할 수 있다. 선택한 하나의 행이나 열을 마우스로 끌면 여러 개의 행과 열을 선택할 수도 있다. 스프레드시트에서 이웃한 몇 개의 계열들을 복사하기 위해서는 맨 위의 행이나 열에서 변수명을 따라 마우스를 끌면 된다. 그러면 데이터 전체가 선택된다. 그리고 Edit/copy를 클릭하여 다른 응용프로그램 데이터를 붙인다.

2) 파일로 내보내기(Exporting to a File)

먼저 Procs/Export/Write Text-Lotus-Excel...을 클릭한 후, 대화상자에서 내보낼 데이터의 이름과 형식을 입력한다. 대화상자를 입력할 때, 다음의 것들을 유의해야 한다.

- 확장자를 함께 입력하면 EViews는 입력된 파일 확장자를 사용한다. 일반적인 스프레드시트 확장자(.xls, .wk3, .wk1, and .wks)를 가진 파일은 거기에 맞는 스프레드시트 형식으로 저장되고 그 외의 것들은 아스키 파일로 저장된다.
- 만일 확장자를 입력하지 않으면 EViews는 콤보 박스에서 선택한 데이터 형식으로 사용한다. 스프레드시트 파일은 적당한 확장자를 가지게 되고 아스키 파일은 대화상자에서 제공한 확장자 없는 이름으로 저장된다. EViews는 사용자가 파일 이름에 확장자를 입력하지 않으면 아스키 파일에 확장자를 붙이지 않는다.

출력 파일을 입력했으면 OK를 클릭하여 데이터출력 대화상자를 연다.

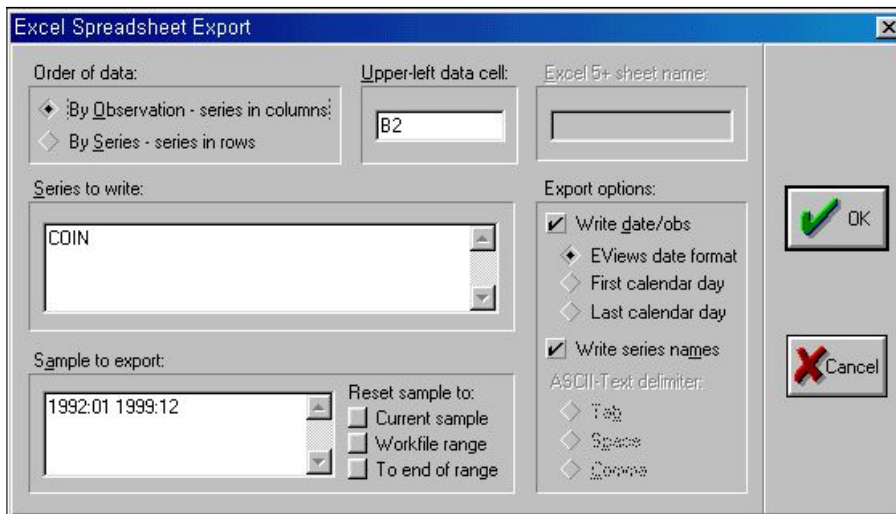
① 스프레드시트로 내보내기

데이터를 스프레드시트로 내보낼 때의 대화상자는 스프레드시트에서 데이터를 읽어 들일 때와 같다. 데이터의 방향, 계열, 표본 등을 입력해야 한다. 추가로 스프레드시트에 계열의 이름과 표본을 포함할 것인지의 여부에 관한 사항이 있다.

만일 계열 이름과 표본 중 하나만 또는 두 개를 모두 다 포함하기를 선택했을 때는 이것에 대한 정보를 입력할 수 있도록 스프레드시트에 여유 공간을 두고 데이터 셀이 시작되어야 한다. 만일 그렇지 않으면 EViews는 선택된 사항들을 무시한다. 예를 들어 A1에서 데이터가 시작되도록 했다면 입력했던 이름과 표본은 스프레드시트에 나타나지 않는다.

예를 들어 작업파일에서 계열을 엑셀파일로 내보내기(Export) 할 경우 작업파일의 특정 계열을 선택한 후 툴바의 Export/Write Text-Lotus-Excel...을 클릭하면 다음과 같은 대화 상자가 나타나는데 여기에서 붙이기 할 셀의 위치, 계열이름, 표본등을 선택한 후 OK버튼을 클릭하면 된다.

<그림 3.11> 엑셀파일로 내보내기



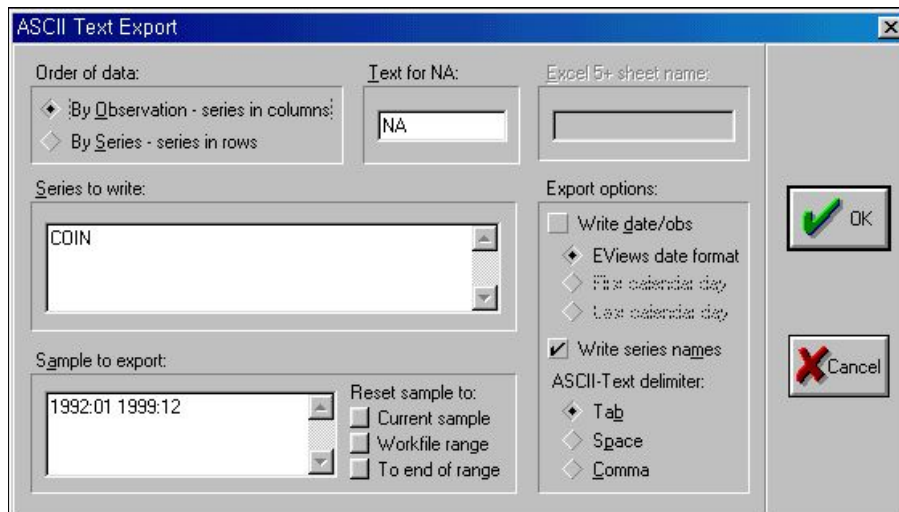
② 아스키 파일로 내보내기

아스키 파일로 내보낼 때의 대화상자는 스프레드시트의 경우와 매우 흡사하다. 그러나 몇 가지 추가적인 선택을 해야 한다.

- missing value로 쓸 문자를 바꿀 수 있다. 간단히 편집상자에서 문자를 입력하면 된다.
- EViews는 탭, 공백 또는 콤마로서 데이터를 구분할 수 있도록 하고 있다. 원하는 형태의 버튼을 클릭하면 된다.

만일 데이터를 계열로 내보내기를 할 때는 경고 메시지가 나타나게 되는데 EViews는 계열의 모든 데이터를 한 줄에 나타내기 때문이다. 데이터가 너무 많으면 다른 프로그램의 line-length가 넘치게 될 수도 있다.

<그림 3.11> 아스키파일로 내보내기



3.3 데이터 개체(Data Objects)

실제로 EViews에서의 데이터 수치들은 하나 이상의 데이터 개체(계열, 그룹, 행렬, 벡터, 스칼라)로 구성되어 있다. 이용자들에게 있어 계열과 그룹은 매우 중요한 개체이므로 이것을 중심으로 설명하고자 한다.

1) 계열(Series)

EViews에서 계열은 변수들의 데이터 집합이다. 계열에서 각각의 데이터는 날짜와 데이터 라벨과 관련되어 있다. 날짜가 포함되어 있는 계열은 보통은 시간에 따른 데이터로 간주된다. 그렇지 않은 것은 어떤 특별한 주기에 따른 것이 아니라고 간주하면 된다.

① 계열 생성하기(Creating a series)

계열 생성하는 방법 중 하나는 메뉴에서 Objects/New Objects...를 선택한 후 Series를 선택한다. 이 때, 계열의 이름을 지정하거나 이름이 없는 새로운 계열을 그대로 선택하면 된다. 그리고 클릭하면 새로운 계열 개체의 스프레드시트 화면이 나타난다.

이 계열에 있는 모든 데이터는 값이 없는 코드로 사용되는 'NA'가 입력되어 있다. 여기서 데이터를 불러오거나 직접 입력할 수 있다.

두 번째 방법은 수학적 식을 이용해서 계열을 만드는 것이다. Quick/Generate Series...를 클릭하고 계열을 정의하는 식을 입력한다. 이것에 관한 설명은 다음 장에서 하기로 하자.

※ 명령어를 사용하여 계열 생성하기

㉠ 단순히 비어있는 계열을 생성하기 위한 명령어

⇒ series 계열명

㉡ 수학적 산식을 이용한 결과를 새로운 계열로 생성하기 위한 명령어

⇒ series 계열명=수학적 산식

⇒ genr 계열명=수학적 산식

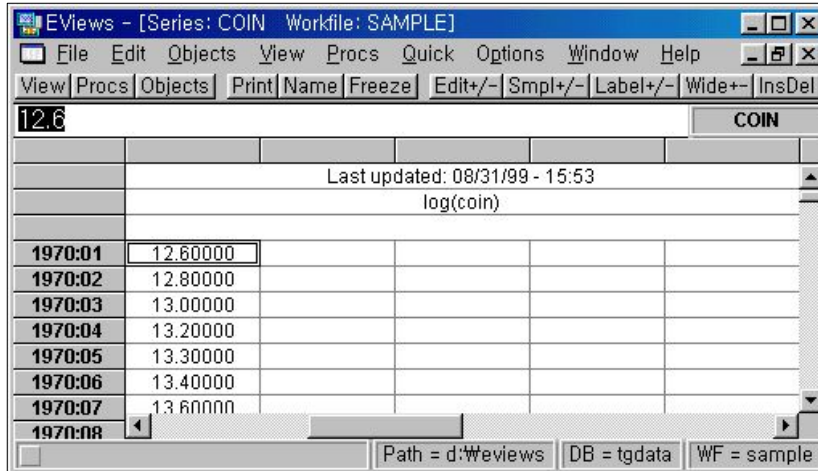
② 계열 편집(Editing a series)

계열 안에 있는 각각의 데이터를 편집할 수 있다.

- 먼저 계열의 스프레드시트 화면을 연다. 만일 계열 창이 스프레드시트 화면이 아니라면 Sheet 버튼을 클릭하거나 View/Spreadsheet를 선택하여 기본 화면으로 바꾼다.

- 다음, 스프레드시트 창은 편집 상태이어야 한다. EViews에서는 스프레드시트 창의 보호상태에서는 편집할 수 없도록 함으로써 계열 안에 있는 데이터를 유지시키는데 툴바에 있는 Edit+/-를 이용하면 편집상태(edit mode)와 보호상태(protected mode)를 왔다 갔다 할 수 있다.

<그림 3.12> 계열 편집



위의 그림은 계열 스프레드시트 창의 편집상태이다. 1970:01에 COIN의 값이 있는 계열 제목줄 바로 아래에 편집 창을 볼 수 있고 스프레드시트에는 선택한 셀의 둘레에 두 겹으로 표시된 상자가 있다.(이 두 겹의 상자는 보호상태에서도 나타난다.)

데이터 값을 바꾸기 위해 셀을 선택하고 값을 입력한 후 엔터를 누른다. 예를 들어 1970:01에 있는 COIN의 값을 바꾸고자 한다면 그 셀을 클릭하고 편집 창에 새로운 값을 입력한 후 엔터를 누르면 된다.

어떤 셀들은 보호되는 것도 있다. 보호된 셀을 선택하면 편집 창에 셀을 편집할 수 없다는 메시지가 나타난다.

- 편집을 끝내고 나면 Edit+/-를 클릭하여 편집상태를 해제함으로써 변환된 값들을 보호해야 한다.

③ 스프레드시트 화면 바꾸기

EViews에서는 스프레드시트 형태에서 데이터를 볼 수 있는 방법이 여러 가지 있다.

좁은 화면(narrow display)은 하나의 열에 계열의 데이터들을 보여주고 한 쪽에는 데이터 라벨을 보여 준다.

넓은 화면(wide display)은 데이터를 왼쪽에서 오른쪽, 위에서 아래 방향으로 정렬하고 행방향의 데이터에 대한 라벨을 한 쪽에 보여준다. 날짜 순으로 된 작업파일은 가능하면 데이터를 데이터의 주기와 대응하도록 정렬시킨다. 이런 식으로 반년 주기 데이터는 한 줄에 두 개, 분기 데이터는 네 개 그리고 월 데이터는 여섯 개씩 데이터가 들어가게 된다.

데이터를 한 열에서 모두 보고자 할 때는 스프레드시트 화면 툴바에서 Wide+/-를 클릭하면 화면보기를 바꿀 수 있다.(이 버튼을 사용할 수 있는 계열 창으로 만들어져 있어야 한다.) Wide+/-는 한 열에서 데이터를 보는 화면과 아래에서 설명하는 화면을 왔다 갔다 할 수 있는 토글버튼이다.

<그림 3.13> 스프레드시트 화면 바꾸기

COIN					
	1	2	3	4	5
Last updated: 08/31/99 - 15:53					
log(coin)					
1970:01	12.60000	12.80000	13.00000	13.20000	13.30000
1970:07	13.60000	13.60000	13.80000	13.80000	13.90000
1971:01	14.10000	14.30000	14.40000	14.60000	14.70000
1971:07	15.00000	15.00000	15.00000	14.90000	15.00000
1972:01	15.00000	15.00000	15.00000	15.30000	15.40000
1972:07	15.70000	15.90000	16.20000	16.50000	16.80000
1973:01	17.50000	17.70000	17.80000	18.10000	18.40000
1973:07					

위의 화면은 열 방향으로 특별한 주기를 가진 데이터를 볼 때 유용하다.

기본화면은 작업파일에 있는 모든 데이터가 현재의 작업구간에 포함되지 않는 데이터까지 모두 나타난다. 작업파일에 있는 모든 데이터 화면과 현재 구간의 데이터 화면은 Smpl+/-를 눌러서 왔다 갔다 할 수 있다.

이러한 토글키를 사용하여 두 개의 화면을 서로 왔다 갔다 하며 사용할 때는 다음의 두 가지를 유의해야 한다.

- 만일 표본기간에 있는 데이터만을 보고자 할 때는 EViews는 한 열에 데이터를 보여주는 화면이 된다.

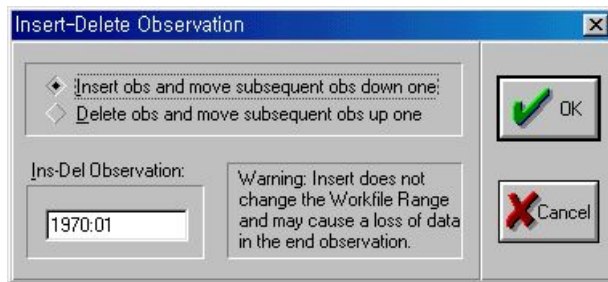
- 넓은 화면으로 보고자 할 때는 EViews는 자동적으로 화면필터를 바꿔 주므로 작업파일에 있는 모든 데이터가 보여진다.

만일 현재 구간에서 데이터를 좁은 화면으로 보고 있다가 넓은 화면으로 보기 위해서는 Wide+/-버튼을 누른다. 그리고 여기서 Wide+/-버튼을 다시 누르면 EViews는 좁은 화면으로 보여주지만 작업파일에 있는 모든 데이터를 보여주게 된다. 이 때 원래의 현재 구간의 좁은 화면으로 되돌아가고자 할 때는 Smpl+/- 버튼을 다시 눌러야 한다.

④ 계열 안에 데이터 삽입과 삭제

계열 안에 데이터를 삽입하거나 삭제할 수 있다. 데이터를 삽입할 때는 먼저 새로운 데이터를 삽입하고자 하는 위치의 셀을 클릭한다. 그 다음 InsDel 버튼을 누른다. 그러면 현재의 위치에 데이터를 삽입할 것인지 삭제할 것인지를 묻는 다음과 같은 대화상자가 나타난다.

<그림 3.14> 계열 안에 데이터 삽입과 삭제



데이터 상자에서 편집하여 삽입 또는 삭제위치를 바꿀 수도 있다. 만일 데이터 삽입을 선택하면 원한 위치에 missing value가 들어 있는 하나의 셀이 생기고 모든 데이터가 아래로 움직여서 작업파일에서 맨 마지막 데이터는 잃게 된다. 이 마지막 데이터를 유지하고 싶으면 데이터를 삽입하기 전에 작업파일을 확장시켜 주어야 한다. 또 데이터 삭제를 선택하게 되면 삭제된 데이터 위치로 모든 데이터가 위로 움직이게 되어 작업파일 범위의 맨 마지막 데이터는 missing value가 된다.

2) 그룹(Group)

여러 계열을 가지고 작업을 할 때, 데이터를 손쉽게 다루기 위해 그룹 개체를 만들 수 있다. 그룹은 계열 이름들의 목록인데 이 계열 이름들은 그 목록 안에 있는 모든 요소에 동시에 접속할 수 있다.

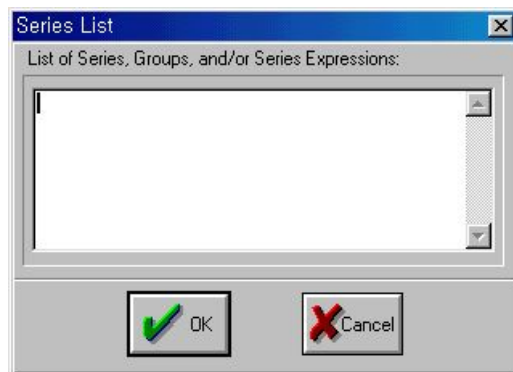
그룹으로 하나의 이름을 이용해서 변수들의 집합을 참조할 수 있다. 그룹개체를 이용하면 각각의 변수들에 대해서 분석하고 그래프를 그려서 출력하기보다는 이 모든 것을 변수들의 집합에 대해서 실행한다. 그러므로, 그룹은 이름들의 목록이 아주 길 때 이것을 입력해야 하는 때에 자주 이용된다. 일단 그룹이 정의되면, 그룹 내에 포함되어 있는 모든 계열을 참조하고자 할 때 그룹 이름을 사용할 수가 있다.

여러 개의 계열을 동시에 검증하거나 분석할 때도 계열의 그룹을 만들 수가 있다. 상관행렬의 계산, 한 계열의 또 다른 계열에 대한 그래프 등도 그룹을 이용할 수가 있다.

① 그룹 생성(Creating Groups)

그룹을 생성하는 데는 몇 가지 방법이 있다. 그 중 가장 빠른 방법은 메뉴나 작업파일 툴바에서 Objects/New Object...를 선택하여 Group을 클릭하고 어떤 경우에는 이름을 지정하면 된다. 다음의 대화상자가 열린다.

<그림 3.15> 그룹 생성



그룹에 포함시킬 계열들의 이름을 입력하고 OK를 누른다. 스프레드시트 화면의 그룹 창이 열린다.

대화상자 안에 그룹 이름이나 계열 식도 입력할 수가 있다. 만일 그룹 이름을 입력하면 그 그룹에 있는 모든 계열은 새로운 그룹에 포함될 것이다. 계열 X, Y, Z가 들어 있는 그룹 GR1이 있고 그룹 GR1과 계열 A, B를 포함하는 새로운 그룹 GR2를 만들려고 한다. 그러면 그룹 GR2에 있는 계열은 X, Y, Z, A, B가 된다. 그룹 GR2는 그룹 GR1에 있는 계열만을 포함하는 것이지 그룹 GR1 자체를 포함하는 것이 아니다. 그러므로 만일 나중에 GR1에 계열을 추가하게 되면 추가된 계열이 GR2에도 추가되는 것은 아니다.

계열식은 뒤에서 자세히 설명하기로 하고 지금은 계열식이 하나 이상의 계열이 포함된 수학적식(즉, $7/2$ or $3*x*y/z$)이라는 것만 알아두자. EViews는 자동적으로 데이터에 대한 식을 계산하여 그 결과를 하나의 계열처럼 나타낸다. 스프레드시트 프로그램을 사용하는 사람들은 이러한 자동적 재계산에 매우 익숙할 것이다.

다음 그림은 계열 COIN, IP의 lag 형태로 된 계열식 $IP(-1)$, 그리고 COIN과 IP가 포함된 계열식으로 이루어진 그룹 스프레드시트 화면이다.

<그림 3.16> 그룹 스프레드시트

obs	COIN	IP(-1)	2*COIN*IP
1970:01	12.60000		7.411765
1970:02	12.80000	3.400000	8.000000
1970:03	13.00000	3.200000	6.842105
1970:04	13.20000	3.800000	6.769231
1970:05	13.30000	3.900000	6.487805
1970:06	13.40000	4.100000	6.700000
1970:07	13.60000	4.000000	6.800000
1970:08	13.60000	4.000000	6.974359
1970:09	13.80000	3.900000	7.076923
1970:10	13.80000	3.900000	6.571429
1970:11	13.90000	4.200000	6.780488
1970:12			

Path = d:\wviews DB = tgdata WF = sample

그룹을 생성하는 또 다른 방법은 Quick/Show...를 선택하거나 작업파일 툴바에서 Show 버튼을 클릭하여 작업파일에 포함할 계열의 목록, 그룹, 계열의 식을 입력한다. 이 방법은 그룹이 생성될 때 개체에 이름을 지정할 수 없다는 점에서 Objects/New object...를 이용하는 것과 약간 다르다.

또 내용이 없는 그룹을 만들 수도 있는데 여기에 새로운 데이터를 입력하거나 다른 프로그램에서 데이터를 복사하여 붙이기를 하여 이용할 수도 있다. 이 방법은 데이터 입력하는 것을 설명할 때 더 자세히 다루기로 하자.

※ 명령어를 사용하여 그룹계열 생성하기 및 보기

- ㉠ 새로운 그룹계열을 만들 때는 'group' 다음에 그룹 이름을 쓰고 그룹에 포함할 계열들을 공백으로 구분해서 나열한다.

group kdh c x1 x2 z

상수 c와 계열 x1, x2, z로 구성된 kdh라는 이름의 그룹을 만든다.

- ㉡ 계열이나 그룹을 보고자 할 때는 'show' 다음에 계열이나 그룹의 이름을 쓰면 된다.

show logy

㉢ 그룹 편집(Editing in a Group)

그룹에서 데이터를 편집하는 것은 계열에서 데이터를 편집하는 것과 비슷하다. 그룹 창을 열고 필요시에는 스프레드시트 화면을 나타내기 위해 Sheet를 클릭한다. 그룹 스프레드시트가 보호 상태로 되어 있다면 Edit+/-를 클릭하여 편집상태로 만든 후에 셀을 선택하고 새로운 값을 입력한 후 엔터를 누른다. 새로운 숫자가 스프레드시트에 나타나게 된다.

그룹은 계열에 대한 참조이기 때문에 그룹 안에서 계열을 편집하면 원래의 계열에 있는 데이터도 변하게 된다.

계열 스프레드시트 화면에서 데이터 전체를 보는 화면과 현재 구간의 데이터만을 보는 화면에 대한 토글키는 Smpl+/-이다. 계열 창과 달리 그룹 창은 항상 하나의 열에만 계열을 보여준다.

3.4 표본(Samples)

EViews의 가장 중요한 개념 중에 하나가 바로 데이터의 표본이다. 표본은 작업파일에서 보기와 통계적 절차를 실행하기 위한 데이터의 집합(때로는 부분집합)을 말한다.

표본은 데이터의 범위를 이용하거나 조건을 만족하는 데이터가 포함하도록 하는 'if conditions'을 이용하여 지정할 수가 있다.

예를 들어, 1970:1부터 1990:12까지, 1995:1부터 1999:12까지의 데이터로 작업을 하겠다는 지정을 할 수 있다. 또 계열 COIN에서 3.6을 초과하는 1970:1부터 1979:12까지의 범위에서 데이터로 작업을 하겠다는 지정도 할 수가 있다.

1) 작업파일 표본(The Workfile Sample)

작업파일을 생성할 때, 전체(global) 또는 작업파일 표본(workfile sample)은 처음에는 작업파일의 전체 범위로 지정된다. 작업파일 표본은 EViews에서 여러 가지 실행을 하는데 이용하고자 하는 데이터의 집합을 말하는 것이다. 다른 데이터 집합으로 작업을 안 해도 된다면 작업파일 표본을 다시 지정할 필요는 없다.

데이터의 작업파일 표본은 작업파일 창의 맨 위 부분을 보면서 언제라도 지정할 수 있다.

<그림 3.17> 작업파일 표본

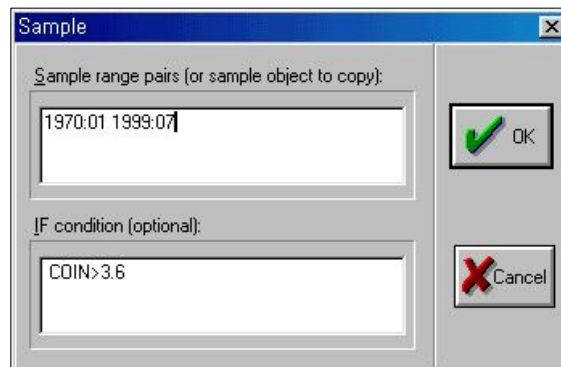


위의 그림에서 표본은 1970:01과 1979:12사이에서 3.6을 초과하는 COIN값이 데이터로 구성되어 있다.

2) 표본 바꾸기

표본을 지정하는데는 세 가지 방법이 있다. 작업파일 툴바에 있는 Sample 버튼을 누르거나 메뉴에서 Quick/Sample...을 선택한다. 또는 작업파일 창에 표시된 표본을 직접 클릭하면 된다. 이 방법 중 하나를 선택하면 추가정보를 위한 대화상자가 나타난다.

<그림 3.18> 표본 바꾸기



위의 대화상자에서 위쪽에 있는 창에는 하나 이상의 날짜를 쌍으로 입력한다. 각 쌍은 표본에 포함된 전체 데이터의 시작과 마지막을 의미한다. 예를 들어 '1970 1980 1990 1999'라고 입력했다면 1970년에서 1980년까지의 데이터와 1990년에서 1999년까지의 데이터를 표본으로 하겠다는 것이다. 1981년부터 1989년까지의 데이터는 제외된다.

EViews에서는 표본의 범위를 좀 더 쉽게 입력할 수 있는 방법이 있다. 첫째, @all을 이용하면 작업파일의 전체 범위를 참조할 수 있다. 위의 작업파일에서, 대화상자에 @all을 입력하는 것은 '1970:01 1999:12'를 입력하는 것과 같다. 또 @first와 @last는 각각 작업파일의 처음과 마지막 데이터를 참조하는 것이다. 다음 세 가지의 표본 범위는 모두 똑같다.

@all

@first 1999:12

1970:01 @last

표본 범위에 수학적식을 포함해서 지정하면 날짜를 생성할 수 있다. 이것은 데이터 표본이 고정된 경우에 크게 유용하다. 예를 들어 위의 작업파

일에서 표본을 '1970:01 1970:01+11' 과 같이 지정했다면 표본은 1970:01에서 시작하여 12개의 데이터가 포함된다.

EViews에서 날짜는 정수를 사용하긴 하지만 정수가 아닌 값을 더하거나 빼 수도 있다. EViews는 자동적으로 숫자를 정수로 변환한다. 그러나 정수로 제대로 변환하지 못하는 경우도 있으므로 유의해야 한다. 이용자가 꼭 정수가 아닌 값을 쓰고자 할 때는 @round, @floor, @ceil 등의 기능을 이용해서 사용하기 바란다.

위의 대화상자의 아래 부분은 추가로 표본 지정에 대한 조건을 입력할 수가 있다. 이 때의 표본은 위 부분에서 지정한 표본의 데이터의 집합과 아래 부분에서 조건으로 지정한 표본에 해당하는 데이터 집합의 교집합이다. 예로서 다음과 같이 입력했다고 하자.

위 부분 : 1980 1999

아래 부분 : ip > 100

여기서 표본에는 1980년과 1999년 사이에서 계열 ip가 100보다 큰 데이터만 포함된다. 또

위 부분 : 1970:1 1999:4

아래 부분 : gdp > gdp(-1)

여기서의 표본에는 1970년 1분기와 1999년 4분기 사이에서 gdp가 전분기의 gdp보다 큰 데이터들만 포함된다.

OR과 AND를 이용해서 좀 더 복잡한 식으로 지정할 수 있다. 예를 들어, 최소한 13년간의 교육을 받은 사람의 일년 수입이 5000달러를 초과하는 사람들만을 분석하고자 한다면 다음과 같이 지정하면 된다.

위 부분 : @all

아래 부분 : income > 5000 and educ >=13

범위를 다중으로 지정해도 조건을 사용할 수 있다.

위 부분 : 50 100 200 250

아래 부분 : income >= 4000 and educ >12

위의 표본은 50번째와 100번째 사이, 200번째와 250번째 사이에서 계열 income이 4000보다 크거나 같고 계열 educ는 12보다 큰 데이터들이 포함된다.

EViews에 저장되어 있는 기능을 이용하면 좀 더 복잡한 조건으로 원하는 데이터만을 표본으로 지정할 수 있다.

위 부분 : 1970:1 1999:4

아래 부분 : (ed>=6 and ed<=13) or earn<@mean(earn)

위의 표본에는 1970년 1분기와 1999년 4분기 사이에서 변수 ed가 6부터 13까지이거나 변수 earn이 그것의 평균보다 작은 모든 데이터를 포함한다.

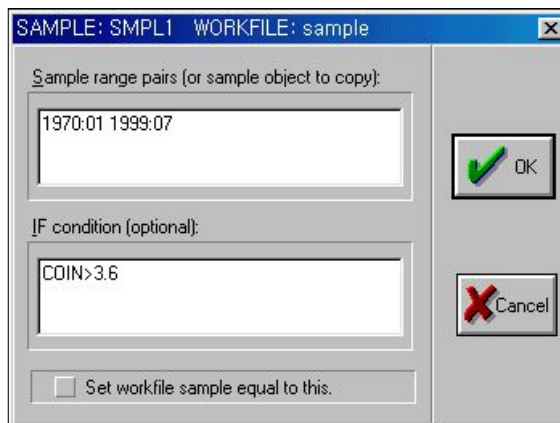
3) 표본 개체(Sample Objects)

이미 앞에서 보았듯이 작업파일의 표본에 매우 복잡한 선택 식을 지정할 수 있다. 그러나 표본을 자주 바꾸어야 하는 경우에는 이런 복잡한 식들을 다시 입력하는 것은 매우 성가시고 시간을 허비하는 일이다. 다행히 EViews에서는 표본 개체에 표본에 대한 정보를 저장하여 그 이름으로 참조하여 이용할 수가 있다. 만일 아주 다양한 데이터로 작업을 할 때는 필수적으로 이 표본 개체를 이용하게 될 것이다.

① 표본 개체 생성(Creating a Sample Object)

표본 개체를 만들 때는 Objects/New...를 선택한다. 새로운 개체 대화상자가 나타나면 Sample을 선택하고 때에 따라 이름을 지정해 준다. 이름을 지정하지 않으면 자동적으로 이름이 주어진다.(이름이 없는 표본 개체는 없다.) OK를 선택하면 다음의 대화상자가 열린다.

<그림 3.19> 표본 개체 생성



SMPL1이라는 대화상자에 표본 개체가 부분적으로 채워져 있다. 이것은 앞에서 설명한 표본을 지정할 때의 대화상자와 비슷해 보이지만 화면에서 볼 수 있는 차이점이 있다. 표본 개체의 이름이 제목 줄에 나타나고 작업파일 표본을 이 표본 개체와 같도록 지정하는 체크박스가 있다.

이 차이에서 표본 개체 대화상자의 뚜렷한 목적 두 가지를 알 수 있다. (1) 표본 개체를 정의하는 것이고 (2) 작업파일 표본을 지정하는 것이다. EViews는 표본개체를 정의하는 것과 작업파일 표본을 설정하는 것이 구분되어 있기 때문에 작업파일 표본을 변경하지 않고도 개체를 정의할 수 있고 또 표본 개체를 정의하면서 작업파일 표본도 변경할 수가 있다.

표본 개체를 정의할 때는 이 대화상자를 모두 채우고 OK를 클릭한다. 그러면 표본 개체가 작업파일 내용에 나타나게 된다.

② 표본 개체의 이용

일단 표본 개체가 생성되면 표본을 지정해야 하는 어느 곳에서도 참조하여 이용할 수 있다.

이미 지정한 표본 개체를 작업파일 표본으로 바로 지정할 수가 있다. 표본 개체의 이름이나 아이콘을 더블 클릭하면 바로 그 표본 개체를 열 수 있다. 이 때 대화상자가 다시 열린다. 만일 표본 개체를 바꾸고 싶으면 표본 지정을 수정하면 된다. 그렇지 않다면 Set workfile sample 이라고 되어 있는 체크 박스를 클릭하고 OK를 누른다.

4) 명령어 창을 이용한 방법

EViews에 어느 정도 익숙해지면 명령어 창에서 작업하는 것이 훨씬 더 쉽다는 것을 알게 될 것이다. Sample을 클릭하거나 Quick/Sample...을 선택하여 표본을 지정하는 대신 smpl 명령어를 이용하면 언제나 표본을 지정할 수 있다. 명령어 창을 활성화시키고 smpl을 입력한 후 표본을 입력한다.

```
smpl 1970:01 1999:07 if COIN>3.6
```

그리고 엔터를 누른다.(위의 예제에서 if는 표본 지정을 두 부분으로 분리시키기 위한 것이다.) 작업파일 창에 표본이 변하는지를 확인한다.

표본 개체를 이용할 때는 먼저 `sample` 정의를 이용해서 표본 개체 이름을 입력하여 표본 개체를 생성하고 표본을 입력한다.

```
sample mysample 1990:01 1999:12 if COIN>3.6
```

표본 개체를 이용해서 작업파일 표본을 지정할 때는 `smpl` 명령어 다음에 표본 개체 이름을 입력하면 된다.

```
smpl mysample
```

표본 개체 `mysample`에서 지정한 것에 따라 작업파일 표본이 지정된다.

또한 다음과 같은 개체를 생성하여 표본으로 사용할 수 있다.

```
%period="1990:01 1995:12"
```

```
smpl %period
```

4. 그래프(Graphs), 표(Tables)

EViews의 계열(series), 그룹(group), 등식(equation) 등은 View를 통하여 그래프, 표와 같은 다른 형태로 나타낼 수 있는데, 이러한 그래프, 표는 원래의 개체가 변하거나, 현재의 표본이 변하면 또한 변하게 된다. 그래프나 표가 변해도 그 변화에 따라 현재의 그래프나 표가 변하지 않도록 유지하고 싶을 때가 있는데 이 때는 View를 고정(freezing)하면 된다. 그래프 View를 고정하는 것은 그래프 개체(Graph Object)를 생성시키는 것이고 표 View를 고정하는 것은 표 개체(Table Object)를 생성시키는 것이다.

이번 장에서는 EViews의 계열 그룹 등식들을 그래프, 표로 표현할 수 있는 방법에 대하여 설명한다.

4.1 그래프(Graphs)

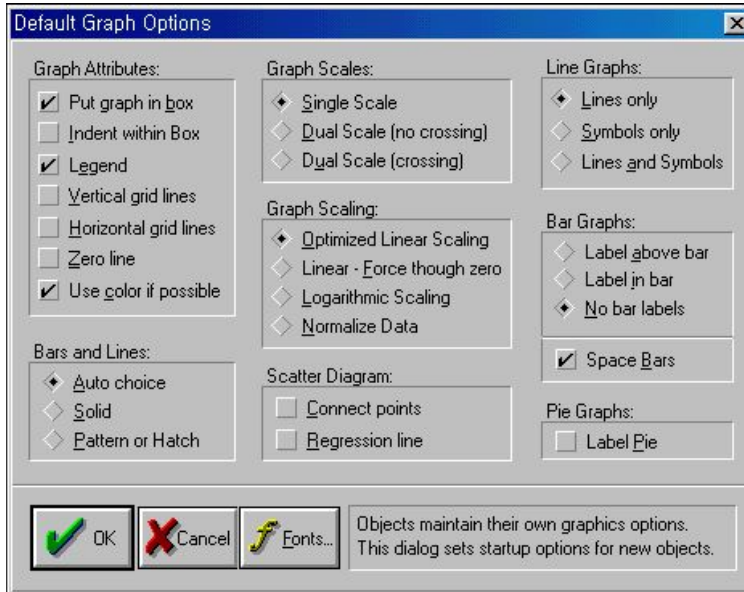
EViews 그래프는 크게 그림부분(graph option), 축(scale labels), 범례(legend), 문자열 상자(addtext), 음영(addshade) 등 5가지의 요소들로 구성되어 있다. 이 요소들 중에 하나를 선택할 때는 그 요소와 연결된 부분을 클릭만 하면 된다. 그러면 선택한 요소 주위에 푸른색 상자가 나타난다. 일단 선택을 하면 그 요소를 끌어서 이동시킬 수 있고 더블 클릭하면 그 요소와 관련된 선택사항들을 지정할 수 있는 대화상자가 나타난다. 또 툴바나 단축 메뉴를 이용하여 그 개체를 완전히 삭제할 수도 있고, 다른 그래프에 형판복사(template)도 할 수 있다.

1) 그래프의 기본설정 값(Default Graph Options)

한 개체의 그래프는 전체 그래프 선택사항에서 지정되어 있는 기본값의 영향을 받는다. 그래프의 기본 설정값을 변경하려면 메인 메뉴에서 Options/Graphics Defaults...를 클릭한다. 이 대화상자에서 선택된 값들은

새로운 개체가 생성되었을 때 적용되고 이미 생성되었던 개체들은 영향을 받지 않는다. 그래프 선택사항 대화상자에서 선택한 값들은 이 기본 값들을 수정하는 것이고 보통 그 개체에 계속 유지된다. 사용자 정의 눈금, 선 모양과 글자에 대한 경우는 고정된 개체에만 적용된다.

<그림 4.1> 그래프의 기본설정 값

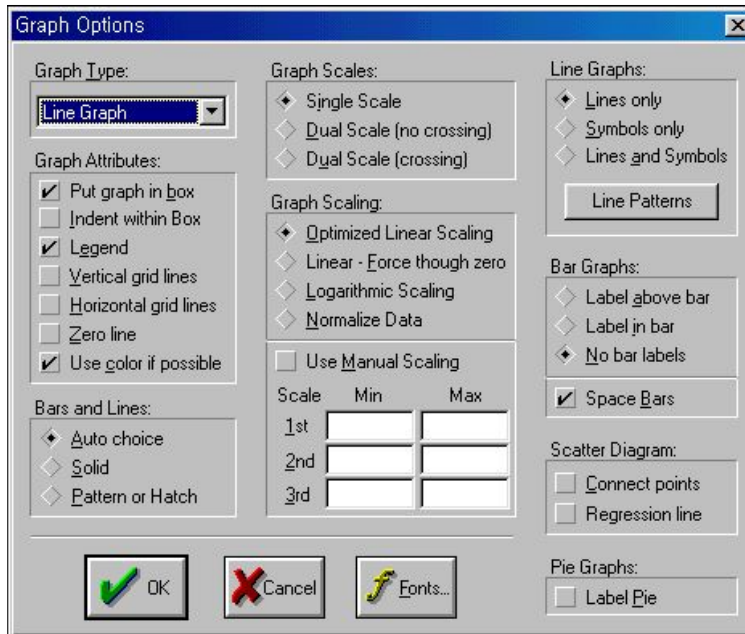


2) 그래프 선택사항(The Graph Options)

그래프에 있어서 중요한 선택사항은 대부분이 그림영역이다. 이 영역 안쪽을 더블 클릭하면 그래프 선택사항 대화상자가 나타난다.

예) a:\data\sample.wf1의 COIN이란 개체를 더블 클릭하면 동행지수의 시계열이 나타난다. 여기서 View/Line Graph를 선택하여 선 그래프를 그린 후 그림부분을 더블 클릭하면 다음과 같은 그래프 선택사항이 나타난다

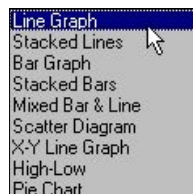
<그림 4.2> 그래프 선택사항 대화상자



① 그래프 형태(Graph Type)

그래프 선택사항 대화상자(그림 4.2)의 왼쪽 상단에 있는 콤보 박스는 그래프 형태를 선택하는 것이다. 그래프 형태를 변경하려면 콤보 박스에서 원하는 항목을 선택하면 된다. 어떤 그래프는 다른 데이터 형태에는 맞지 않는 것들도 있다.(예를 들어 단일 계열에 대해서는 산점도를 볼 수 없다) 게다가 어떤 보기들은 그래프 형태를 변경할 수 없는 것도 있다. 이런 경우에 그래프 형태는 Special로 나타나고 콤보 박스에서는 그 항목을 취할 수 없게 된다.

<그림 4.3> 그래프 형태



그래프 형태에는 다음과 같은 것들이 있다.

- **선 그래프(Line Graph):** 데이터의 척도나 시간에 대한 값을 그린다.
- **누적 선형(Stacked Lines):** 각각의 선이 계열의 누적합계를 나타내는 그림이다. 각 계열의 값은 각각의 선들 사이의 수직간 거리로서 척도나 시간에 대하여 그림이 그려진다.
- **막대 그래프(Bar Graph):** 각 계열의 값을 막대의 높이로 나타내는 그림이다.
- **누적 막대(Stacked Bars):** 각 계열의 값을 하나의 막대에 색깔로 구분 지은 조각들로 그리는 그림이다. 이때 막대의 전체 크기는 각 계열의 값을 합한 값이다.
- **혼합형 막대와 선 그래프(Mixed Bar & Line):** 첫 번째 계열을 막대 그래프로 나머지 계열은 선 그래프로 혼합하여 그리는 그림이다.
- **산점도(Scatter Diagram):** 첫 번째 계열을 가로축에 나머지 계열을 세로축으로 하여 각각 다른 부호로 흩어진 정도의 그림을 그린다.
- **XY 선 그래프(XY Line Graph):** 첫 번째 계열을 가로축에 나머지 계열을 세로축으로 하여 각각을 연결된 선으로 그림을 그린다.
- **하이-로우(High-Low):** high-low-close 그래프를 첫 번째와 두 번째 계열을 세로선으로 연결하고 세 번째 계열(선택적)은 부호로 그리는 것이다.
- **파이 그래프(Pie Chart):** 각 계열을 각기 색깔이 다른 조각들로 하나의 파이 형태를 이루는 그림을 그린다. 이 때 각 조각의 너비는 전체 계열에 대한 비율로 그려진다. 음수값 계열은 차트에서 제외된다.

② 그래프 항목(Graph Attributes)

대화상자의 왼쪽에 있는 그래프 항목의 목록도 변경할 수 있다. 각 항목들은 항목의 왼쪽에 있는 네모를 체크하거나 해제하면 된다. 그래프 외곽선의 유무(Put graph in box), 그래프를 외곽선 안으로 모으기(Indent within box), 계열의 범례(Legend), 가로와 세로축 눈금선(Horizontal and Vertical lines), 가로축과 세로축이 0에서 교차(Zero line), 색깔 이용(Use color if possible)을 할 수 있다.

③ 막대와 선(Bars and Lines)

여기서는 선, 막대, 파이 차트에서의 조각 등을 조정한다. Eviews가 자동적으로 선택하여 그리게 하거나, 점선 표시, 영역의 단일 색 표현, 평행 선 모양을 선택할 수 있다. Auto choice를 선택하면 계열들을 구분하기 위해서 다른 색깔을 이용하여 그린다. 흑백 프린터로 그래프를 출력할 때는 Views는 색깔을 무늬로 대체하여 준다.

④ 그래프 눈금 선택(Graph Scales option)

Graph Scales부분은 모든 계열의 눈금을 왼쪽의 세로축에 단일 눈금으로 적용하거나, 첫 번째 계열을 왼쪽에 나머지 계열은 오른쪽에 눈금을 표시하여 양쪽을 다 사용할 수가 있다. No crossing은 계열이 서로 교차하지 않도록 눈금을 조정하는 것이고 Crossing은 계열의 전체 눈금범위를 적용하여 그래프에서 각 계열이 서로 교차되어 그려진다.

Graph Scaling은 눈금을 조정할 수 있는 사항들이다.

- **최적화(Optimized)**: 계열의 전체 범위를 나타낼 수 있도록 눈금을 나타낸다
- **0을 통과(Force through zero)**: 세로축이 0을 포함하도록 나타낸다.
- **로그 눈금(Logarithmic scaling)**: 음수가 아닌 값에 대해서만 가능하다.
- **표준화(Normalize Data)**: 각 계열을 평균은 0, 표준 편차가 1이 되도록 데이터를 표준화하여 눈금을 나타낸다.

사용자 정의 눈금(Use Manual Scaling)은 각 축의 최대값과 최소값을 직접 지정할 수 있다. 그래프는 임시적으로 세 개의 축을 가진다. 선과 막대 형태에서는 첫 번째 눈금은 왼쪽 축이고 두 번째가 오른쪽 축이다.(양쪽 눈금을 선택한 경우) XY 선그래프나 산점도에서는 첫 번째 눈금은 가로축, 두 번째가 왼쪽, 세 번째가 오른쪽 축이다.(양쪽 눈금을 선택한 경우)

만일 최대값을 표본에 있는 계열의 가장 큰 값보다 작은 값으로 지정하거나 최소값을 계열의 가장 작은 값보다 크게 지정하게 되면 EViews는 이용자가 지정한 값을 무시하고 모든 데이터 값이 나타날 수 있도록 표시한다.

⑤ 선 그래프(Line Graph)

여기서는 각 계열에 대한 데이터 점의 연결을 선, 부호 또는 두 가지를 모두 이용할 것인지를 결정한다.

선 모양(Line Patterns): 각 계열의 선 모양을 조정하는 것이다. 박스를 클릭하면 여러 가지 선 모양이 나타난다. 계열에 색깔을 이용할 때는 선 모양을 선택하면 색깔도 변하게 된다. 선 색깔을 바꾸고 싶으면 선 모양을 다른 것으로 선택하면 된다.

⑥ 막대 그래프(Bar Graph)

여기서는 라벨을 막대의 위나 안에 또는 표시하지 않을 것인지를 결정한다. 또 이웃한 막대들 사이의 여백을 조정한다. 표본이 커서 막대가 아주 많은 경우에는 막대에 라벨이 표시되지 않고 막대 사이에 여백도 없다.

⑦ 산점도(Scatter Diagram)

연속한 점들을 선으로 연결할 수가 있고 가로축과 세로축의 2변량 회귀식에서 얻은 적합선을 그려준다.

⑧ 파이 그래프(Pie Graph)

각 계열의 데이터나 데이터 수의 라벨을 표시해 준다. 파이가 너무 많을 때 라벨은 드문드문 표시된다.

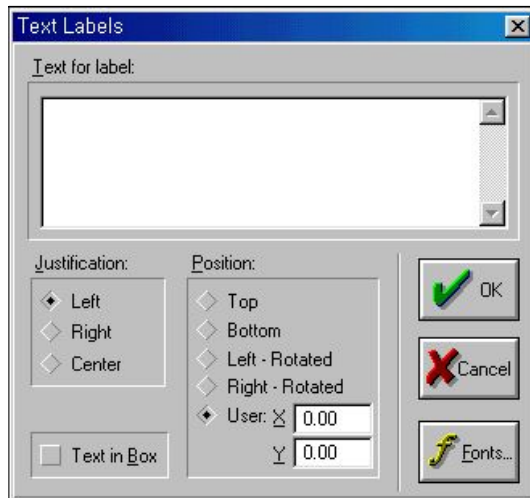
⑨ 글꼴(Fonts)

그래프 라벨(축, 범례, 문자열)의 글자를 변경할 때는 Fonts를 클릭한다. 그래프 선택사항 대화상자에 있는 Font는 모든 그래프 요소에 대하여 기본값으로 설정되어 있다. 이용자가 축, 범례, 문자열에 대해 각각 지정해 줄 수 있다.

3) 문자열 추가(Adding Text)

그래프 위에 문자열을 추가하고 싶을 때가 있을 것이다. 이것은 특정 데이터나 기간을 표시하거나 제목 추가, 그래프에 대한 설명을 덧붙이고자 할 때도 유용하다. 고정된 그래프 개체에서 툴바에 있는 AddText 버튼을 누르거나 Procs/Add text...를 선택하면 된다. 다음은 문자열 상자의 대화상자이다.

<그림 4.4> 문자열 추가 상자



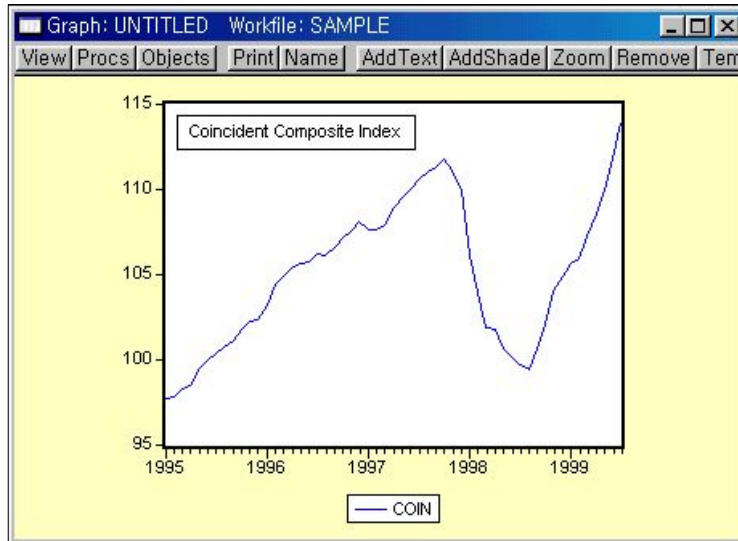
편집상자에 원하는 문자를 입력한다. 공백이나 대소문자도 사용할 수 있다. 만일 한 줄 이상 쓸 경우에는 각 라인마다 엔터를 누른다.

- 정렬(Justification)은 문자열을 왼쪽, 오른쪽, 가운데로 정렬시킨다.
- Text in Box는 라벨을 상자 안에 넣는다.
- 글꼴(Fonts)은 라벨의 글자 속성을 지정한다.

한가지 주의할 점은 문자열 입력시 한글은 “<”, “(”와 같은 특수기호를 먼저 입력한 다음 한글을 입력한 후 “>”, “)”를 사용하여 닫아야 된다.(예: <동행지수>)

예) EViews파일(a:\data\sample.wf1)에서 동행종합지수(COIN)을 선택하여 선그래프(View/Line Graph) 그려서 제목을 “Coincident Composite Index”라고 입력하면 다음의 결과가 나타난다.

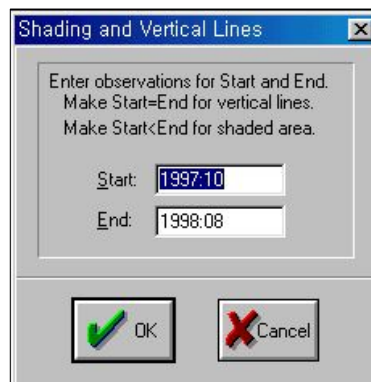
<그림 4.5> 선그래프에서 문자열 추가한 경우



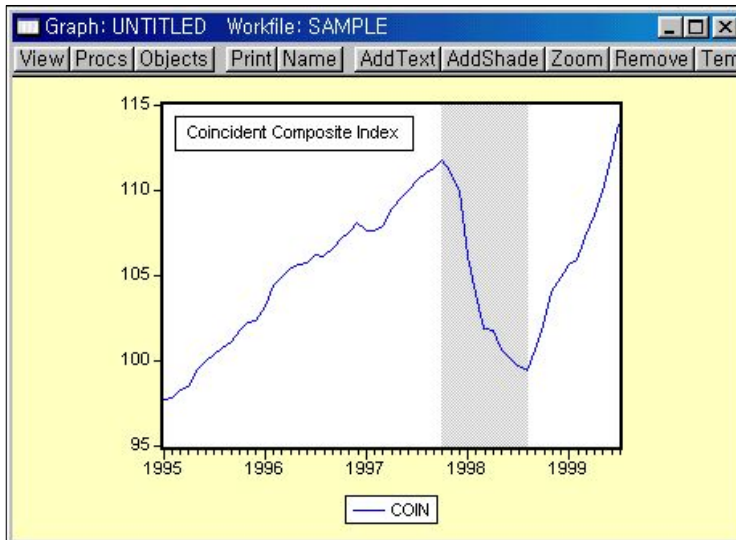
4) 음영 추가(Adding Shading)

그래프에서 데이터 구간이나 세로 방향으로 음영을 줄 수가 있다. 그래프 개체에서 툴바에 있는 AddShade를 클릭하거나 Procs/Add shading...을 선택한다. 음영기간을 입력할 수 있는 대화상자가 나타난다. 음영을 넣고자 하는 시작과 끝 날짜(또는 데이터 수)를 입력한다. 만일 시작과 끝 날짜(또는 데이터)가 같으면 그 날짜에 세로선이 생긴다.

<그림 4.6> 음영 추가(addshade)



<그림 4.7> 음영 추가한 선 그래프



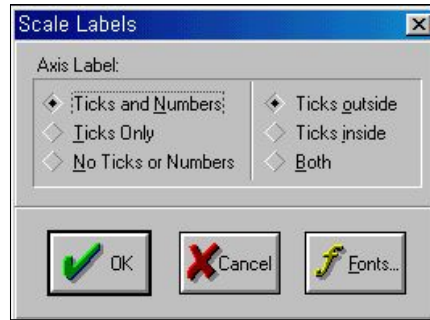
음영을 지우려면 음영 부분을 클릭한 후 툴바에서 Remove를 누른다. AddShade는 현재 그래프의 가로축에 있는 날짜(또는 데이터)에 대해서만 유효하다.

다중 그래프에서는 툴바에서 AddShade를 선택하면 모든 그래프에 지정한 범위의 음영이 생긴다. 다중 그래프에서 하나의 그래프에만 음영을 넣고 싶을 때는 AddShade를 클릭하기 전에 음영을 넣고자 하는 그래프를 선택해야 한다.

5) 축 서식(Customizing Axis Labels)

그림 영역 밖에 있는 가로나 세로축을 더블 클릭하면 눈금 라벨 대화상자가 나타난다. 이 대화상자에서는 축의 눈금표시와 숫자를 표시할 것인지의 여부와 눈금표시를 축의 밖이나, 안 또는 교차하도록 표시할 것인지를 결정한다. 눈금 숫자의 글꼴은 대화상자에서 Font를 클릭하면 된다.

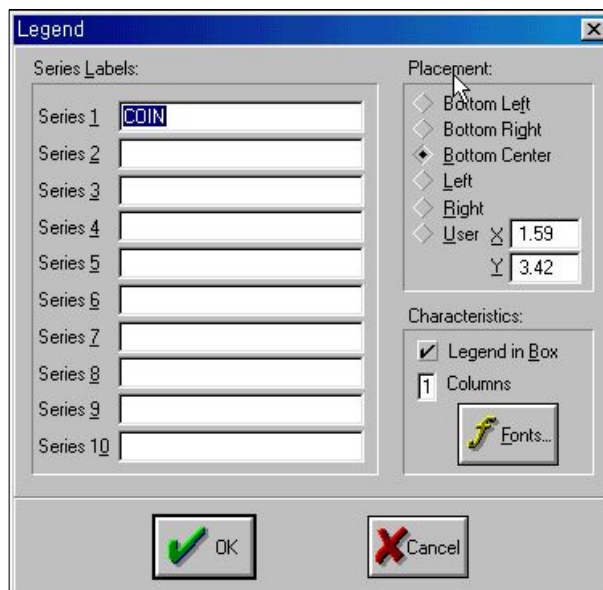
<그림 4.8> 축 서식 대화상자



6) 범례 서식(Customizing Legends)

범례를 두 번 클릭하면 범례 대화상자가 나타난다. 이 대화상자에서는 다음과 같은 것들이 있다.

<그림 4.9> 범례 대화상자



- 계열의 이름을 구분할 수 있도록 계열의 라벨을 수정한다.
- 범례의 위치를 지정한다.

- 범례를 들어 있는 상자의 외곽선을 없애려면 Legend in Box를 체크하지 않으면 된다.
- Columns은 그래프에 계열이 많이 있는 경우 범례를 몇 개의 열로 나타낼 것인지를 지정한다.
- 범례의 글꼴은 Font를 눌러서 조정한다.

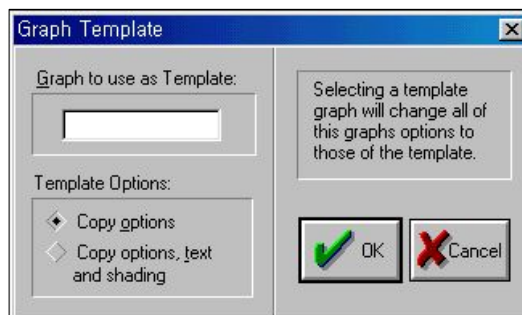
7) 그래프 항목 삭제(Removing Graph Elements)

그래프 툴바에 있는 Remove는 고정된 그래프에서 선택한 요소를 삭제한다. 예를 들어 그래프에 추가한 문자열을 제거하고자 할 때는 그 문자열을 클릭한다. 그러면 그 문자열 주변에 선이 생기게 된다. 이 때 Remove를 누르면 문자열이 없어진다. 같은 방법으로 범례, 눈금, 음영도 모두 삭제할 수 있다.

8) 그래프 형판(Graph Templates)

그래프를 이용자가 원하는 양식으로 만들어 놓으면 그 그래프를 다른 그래프에도 똑같은 양식으로 적용하고 싶을 때가 있다. EViews에서는 어떤 그래프에 원하는 양식을 형판(template)으로 만들어 놓고 이 양식을 새로운 그래프나 이미 존재하는 그래프에 적용하는 기능이 있다. 이 형판을 하나의 그래프 양식으로 생각하고 다른 그래프에 적용할 수 있다.

<그림 4.10> 그래프 형판(Template)



먼저, 형판으로 이용하고자 하는 그래프 개체에 이름을 지정해 준다. 그런 다음 형판을 적용하고자 하는 그래프 개체의 툴바에서 Template를 클릭한다. 형판으로 이용할 그래프 개체의 이름을 입력하면 된다. 여기에는

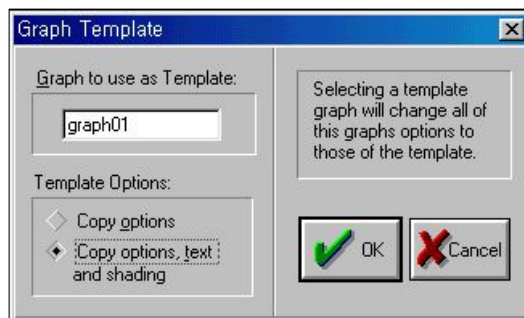
다음의 두 가지 선택사항이 있다.

- **Copy options:** 형판 그래프에서 설정한 그래프 선택사항만 적용된다.
- **Copy options, text and shading:** 형판 그래프에서 설정한 문자 라벨과 음영까지도 똑같이 적용된다.

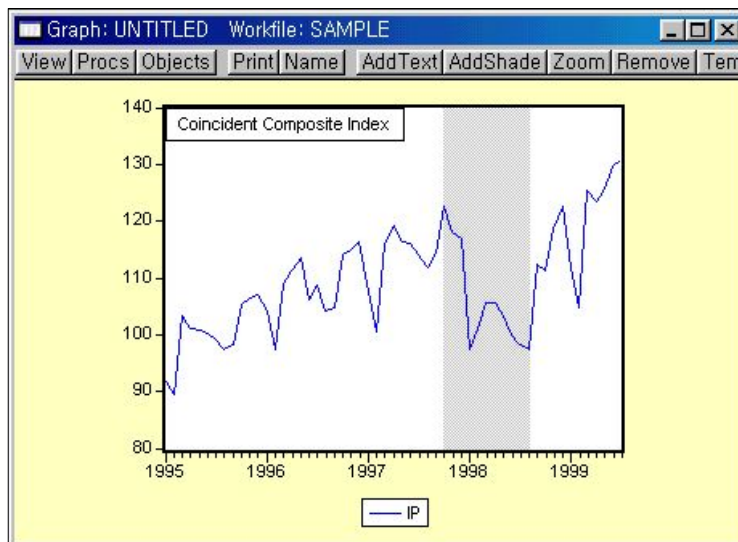
또한 다중 그래프 창에서 Template를 선택하면 선택사항은 다중그래프 개체의 각 그래프에 적용된다.

예) a:\data\sample.wf1에서 <그림 4.7>를 graph01로 저장한 후, 산업생산 지수(IP)의 선그래프를 그려서 그래프 형판을 이용하여 문자 라벨, 음영까지 복사하면 다음과 같다.

<그림 4.11> 그래프 형판(Template)의 예



<그림 4.12> 그래프 형판(Template) 복사 결과



9) 그래프 명령어(Graph Commands)

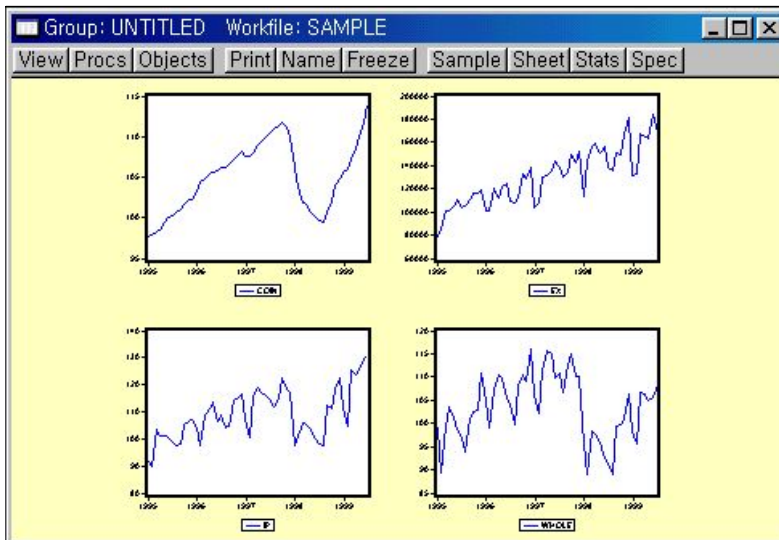
이런 몇 가지 절차들을 자동으로 처리하고자 하는 이용자 즉, 규칙적으로 보고서를 만드는 사람들을 위해서 EViews는 명령어와 프로그램을 이용해서 그래프 서식을 수행할 수 있게 해 준다.

- ㉠ 먼저 EViews파일 a:\data\sample.wf1을 open한다
⇒ open a:\data\sample.wf1
- ㉡ a:\data\sample.wf1에서 COIN자료를 선그래프로 그려서 graph1로 고정하는 명령어
⇒ freeze(graph1) coin.line
- ㉢ graph1에서 범례의 제목을 “동행지수”로 바꾸는 명령어
⇒ graph1.name(1) <동행지수>
• 주의할 점: graph1.name뒤에는 꼭 (1)를 사용해야 함
- ㉣ graph1에서 문자열 추가를 이용하여 TOP에 “전국의 동행종합지수”라고 입력하는 명령어
⇒ graph1.addtext(t) <전국의 동행종합지수>
• 주의할 점: BOTTOM일 경우 graph1.addtext뒤에 (b)를 사용하면 됨
- ㉤ graph1에서 문자열 추가를 이용하여 LEFT TOP에 “1995=100”라고 입력하는 명령어
⇒ graph1.addtext(lt) 1995=100
• 주의할 점: graph1.addtext뒤에 lt, rt, lb, rb 등을 사용할 수 있으나 동일한 위치인 lt에 놓여짐
- ㉥ graph1에서 1997년 10월에서 1998년 8월까지 음영을 추가할 수 있는 명령어
⇒ graph1.shade 1997:10 1998:8
- ㉦ ㉠~㉥단계를 거친 그래프를 바로 볼 수 있는 명령어
⇒ show graph1

4.2 다중 그래프 개체(Multiple Graph Objects)

다음 몇 가지 개체들은 다중 그래프로 이루어져 있다. 단일 그래프와 같이 이것도 고정시켜서 그래프 개체를 바꿀 수 있다. 예를 들면, 동행종합지수(COIN), 수출액(EX), 산업생산지수(IP), 도소매판매액지수(WHOLE)의 다중 그래프를 나타낸다. Freeze를 클릭하면 다중 그래프를 포함하는 그래프 개체를 생성하게 된다.

<그림 4.13> 다중 그래프



다중 그래프를 포함하는 그래프 개체는 이미 존재하는 그래프 개체들의 이름을 연결해서 생성할 수도 있다. 원하는 그래프를 선택하고 선택한 이름들 중의 아무거나 하나를 더블 클릭하면 된다.

그래프를 연결하는 또 다른 방법으로는 Quick/Show...를 선택하고 그래프 이름을 입력한다.

- 그래프를 배치시킬 수 있다. 개별 그래프를 한 번 클릭한다.(위 그림에서는 왼쪽 상단의 그래프가 선택되어졌다) 그리고 그것을 끌어서 새로운 위치로 이동시킨다.
- 선택한 그래프를 삭제할 수 있다. 원하는 그래프를 선택하고 그래프 툴바에서 Remove를 누른다.
- 각각의 그래프에 대해 선택사항을 설정할 수 있다. 단일 그래프에 대한

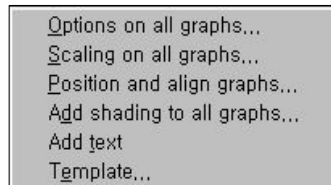
선택사항을 변경하려면 그래프 영역을 더블 클릭하여 그래프 선택사항 대화상자를 연다.

1) 다중 그래프 선택사항(Multiple Graph Options)

다중 그래프에서는 위에서 말한 것처럼 각각의 그래프에 대해 선택사항을 다르게 지정하거나 또는 모든 그래프에 똑같은 선택사항을 지정할 수도 있다.

모든 그래프에 똑같은 선택사항을 적용하려면 그래프의 노란색 바탕 아무 곳이나 더블 클릭하거나 Procs를 선택한다. 다음과 같은 메뉴가 나타난다.

<그림 4.14> 다중 그래프 선택사항



위의 메뉴에서 지정한 항목들은 그래프 개체에 있는 모든 그래프에 적용된다.

- 모든 그래프를 똑같은 눈금으로 표시하려면 그래프 전체 눈금 변경 (Change scaling on all graphs...)을 선택한다.
- 다중 그래프에 있는 개별 그래프는 끌어서 자유롭게 위치를 이동할 수 있지만 전체 그래프를 몇 개의 열로 나타낼 것인지 또 그래프간의 간격을 조정하고 싶을 때는 Procs/Position and align graphs...를 선택하면 된다.
- 만일 그래프가 똑같은 데이터 축(X축)을 가질 때는 개체에 있는 각각의 그래프에 음영을 추가할 수 있다. 이것은 Add shading to graphs...를 선택하고 음영을 넣을 시작과 끝 날짜를 지정해 주면 된다.
- Add text를 선택하면 다중 그래프에 대한 주석을 달 수 있다. 다중 그래프에 한 항목을 추가하는 것은 개별 그래프에 추가하는 것과 다른데 그것은 전체 그래프가 개별 그래프와 같이 움직이지 않기 때문이다.

2) 다중 그래프 명령어

예를 들면, 동행종합지수(COIN), 수출액(EX), 산업생산지수(IP), 도소매 판매액지수(WHOLE)의 다중 그래프를 나타내는 것을 명령어로 표현하면 다음과 같다.

㉠ 먼저 EViews파일 a:\data\sample.wf1을 open한다

⇒ open a:\data\sample.wf1

㉡ a:\data\sample.wf1에서 COIN, EX, IP, WHOLE의 선그래프를 그려서 각각 고정하는 명령어

⇒ freeze(graph2) coin.line

freeze(graph3) ex.line

freeze(graph4) ip.line

freeze(graph5) whole.line

㉢ 4개의 그래프를 다중그래프로 만드는 명령어

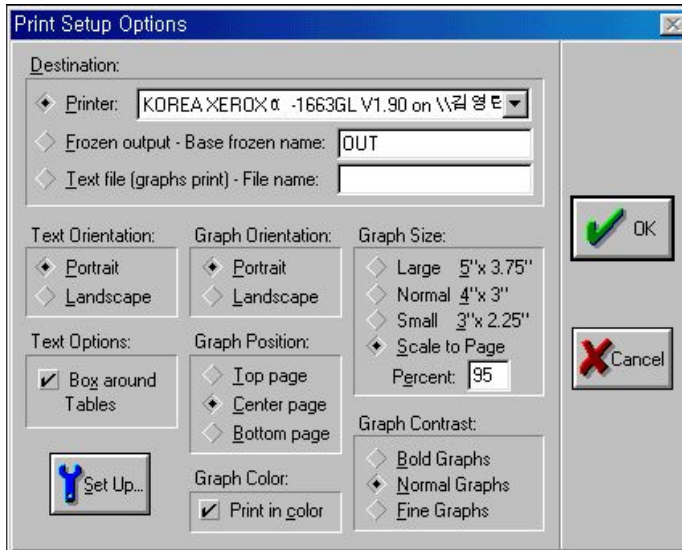
⇒ freeze(graph6) graph2 graph3 graph4 graph5

다중 그래프를 그린 후 문자열 추가(addtext), 음영추가(addshade), 셀 서식, 범례 서식, 그래프형판 복사 등은 단일 그래프와 같은 방법으로 사용하면 된다.

4.3 그래프 출력

그래프 보거나 그래프 개체 창 툴바에서 Print를 클릭하면 그래프를 출력할 수 있다. 메인 메뉴에서 File/Print Setup 또는 Options/Print Setup을 이용하면 출력에 관한 선택사항을 지정할 수 있다.

<그림 4.15> 프린트 선택사항



대부분의 선택사항들은 알 수 있는 것들이다. 만일 칼라 프린터를 이용해서 출력하고자 할 때는 Print in color 상자를 표시해 주면 된다. 반대로 모노 프린터기를 이용할 때는 이 상자를 표시하지 않아야 색깔을 선 모양으로 구별해 준다.

4.4 다른 프로그램에 그래프 복사하기

EViews 그래프를 워드프로세서의 문서에 직접 삽입할 수 있다. 먼저 그래프가 있는 개체 창을 활성화시킨다. 창의 아무곳이나 클릭하면 제목 줄이 밝은 색으로 변한다. 메인 메뉴에서 Edit/Copy를 클릭하면 대화상자가 나타난다.

그래프를 클립보드나 디스크 파일로 복사할 수 있다. 칼라 그래프나 굵은 선의 그래프로 할 수가 있다. 만일 칼라 프린터가 아니면 그래프에 색깔을 이용하지 않은 모노 그래프로 복사하는 것이 좋다. 만일 이 그래프를 다른 문서나 응용프로그램에서 이용할 때는 Make metafile "Placeable"을 선택해야 한다.

그래프를 클립보드로 복사하면 워드프로세서의 문서에 붙이기를 할 수 있다. 한글, MSWord 및 WordPerfect같은 프로그램에서는 그래프의 크기

와 위치를 조정할 수 있다. 또한 그리기 프로그램에도 붙이기를 할 수 있는데 워드 프로세서나 다른 응용 프로그램에 붙이기 전에 좀 더 수정을 할 수도 있다.

4.5 표(Tables)

그래프가 아닌 개체들은 표로 표시된다. 그래프에 두 가지 형태가 있는 것처럼 표에도 두 가지 형태가 있다. 표 개체는 표 보기를 고정해서 생성할 수 있다. 그래프 개체와 같이 표 개체는 원래 개체의 현재 내용을 반영하지 않고 개체가 고정된 시점에서의 개체 내용을 기본으로 한다.

1) 표 선택사항

그래프 개체와 같이 표에 대한 편집 선택사항은 표 보기보다 더 많이 있다. 이 선택사항들은 Procs나 표 창의 툴바를 이용한다. 표를 편집할 때 편집할 셀을 선택하거나 사각형으로 둘러싸인 셀을 선택하면 된다. 셀은 표에서 열 문자와 행 번호로 구분되어져 있다. Edit 버튼을 누르기 전에 셀을 선택하면 대화상자는 선택한 셀에만 적용된다.

- **글꼴(Font):** 표에서 사용할 글자의 속성을 지정한다.
- **삽입-삭제(Insert-Delete):** 원하는 위치에 행이나 열을 삽입 또는 삭제한다.
- **열 너비(Column width):** 표에서 열 너비를 지정한다. 너비는 문자의 수로 측정되고 표에서 사용하는 글자에 따라 다르다.
- **셀 서식(Number Cell Format):** 몇 개의 셀 또는 표 전체에 대하여 사용할 숫자의 양식을 지정한다. Fixed characters는 전체 자리수이고 Fixed decimal은 소숫점 아래의 자리수를 지정하는 것이다. 만일 셀 너비가 충분히 넓지 않을 때는 소수점 아래의 자리수가 안 보일 수도 있다. Justification은 Number Cell Format을 선택했을 때만 나타나는데 선택한 모든 셀의 정렬을 어떻게 할 것인지를 지정해야 한다.
- **정렬(Justification):** 숫자와 문자를 정렬한다.
- **취소선(Horizontal Lines):** 지정한 셀에 두줄의 취소선을 수평으로 그리거나 제거한다. 선은 셀의 내용위에 겹쳐서 그려진다. 선 그 자체만을 그리려면 InsDel로 빈 행을 삽입한다.

- **셀구분(Grid+-)**: 셀 구분선을 그리거나 해제하는 토클이다.
- **제목(Title)**: 표 중앙의 위 부분에 제목을 넣는다. 이것은 작업파일에 있는 표에 대한 개체 이름을 지정하는 Name과는 다른 것이다.
- **편집(Edit+-)**: 표에서 문자나 숫자를 편집할 수 있는 편집 모드로 토클이다.

2) 표 명령어

이런 몇 가지 절차들을 자동으로 처리하고자 하는 사용자 즉, 규칙적으로 보고서를 만드는 사람들을 위해서 EViews는 명령어와 프로그램을 이용해서 그래프 서식을 수행할 수 있게 해 준다.

㉠ a:\data\sample.wf1에서 COIN자료를 표로 만들어 table1로 고정하는 명령어

⇒ freeze(table1) coin

㉡ table1에서 2번째 자료의 열크기를 조정하는 명령어

⇒ setcolwidth(table1,2,12)

• 주의할 점:

2: 두 번째 자료라는 의미, graph1에서 1은 연도이며 2는 coin자료임

12: 두 번째 자료를 열의 너비를 12로 조정하라는 의미

4.6 다른 프로그램에 표 복사하기

EViews에서 작성한 표를 스프레드시트나 워드 프로세서로 잘라서 붙이기를 할 수 있다. 복사하고자 하는 표 영역을 선택한 후 메인 메뉴에서 Edit/Copy를 선택한다. 이 때 나타나는 대화상자에서 표를 현재 보이는 그대로 복사할 것인지 아니면 더 정밀한 숫자의 상태로 복사할 것인지를 선택할 수 있다. 둘 중 하나를 선택한 후 붙이고자 하는 응용프로그램으로 이동한 다음 붙이기를 하면 된다.

EViews는 표를 RTF(Rich Text Format) 파일로 복사하는데 이때 표에서 작성한 양식을 그대로 유지할 수 있다. 이렇게 Microsoft Word나

RTF를 지원하는 다른 응용프로그램으로 표를 복사하여 붙이기를 하면 이용자가 EViews에서 작성한 상태의 표를 그대로 얻을 수 있다. 예를 들어 다음은 EViews에서 복사하여 한글Word로 직접 붙이기를 한 결과이다.

Pairwise Granger Causality Tests

Date: 10/26/99 Time: 13:48

Sample: 1995:01 1999:07

Lags: 2

Null Hypothesis:	Obs	F-Statistic	Probability
IP does not Granger Cause COIN	55	0.09597	0.90866
COIN does not Granger Cause IP		5.23359	0.00863

어떤 워드 프로세서는 자유형식의 파일로 클립보드의 내용을 붙이기 하는 선택사항이 있는 것도 있다. 만일 이용자가 표를 자유형식의 파일로 붙이기 하고자 한다면 Edit/Paste Special을 선택해야 한다.

5. 데이터 작업(Working with data)

이 장에서는 EViews 언어로 계열과 그룹의 데이터 생성, 처리 등의 사용법에 대해 설명하겠다. 먼저 EViews에서 수학적 표현에 대한 기본적인 규칙을 설명한 다음 계열과 그룹 데이터에서 이런 수학적 표현들을 어떻게 사용하는 지를 설명하겠다.

5.1 식 이용하기(Using Expressions)

EViews의 가장 큰 특징 중 하나는 수학적 표현을 이용하여 데이터를 처리할 수 있다는 것이다. EViews는 내부에 외장함수를 갖고 있어 데이터에 대한 복잡한 수학적 처리를 단 몇 개의 키보드 입력으로 간단하게 처리할 수 있다.

EViews 표현은 숫자, 계열 이름, 함수, 수학적이고 관계적인 연산들의 조합이다. 어떤 특별한 형태에서는 EViews 개체를 포함하는 수학적 연산을 설명하기 위한 표현을 사용하기도 한다.

다른 프로그램에서와 마찬가지로, 이런 표현들은 이미 존재하는 계열에서 새로운 계열을 계산, 데이터의 표본을 지정, 추정 및 예측에 관한 식들도 지정할 수가 있다. 이런 간단한 표현뿐만 아니라 계열을 사용하고자 하는 어느 곳에서든지 상호적으로 이런 표현들을 사용할 수가 있다. 이런 중요한 특징에 대해 간단하게 언급하고 먼저 표현에 대한 기본적인 이용에 대해 설명하겠다.

1) 연산자(Operators)

EViews 표현은 일반적인 산술 연산의 연산자가 포함된다. 더하기(+), 빼기(-), 곱하기(*), 나누기(/)와 거듭제곱(^)과 같은 연산자들이 사용되어

$$5+6*7.0/3$$

$$7+3e-2/10.2345+6*10^2+3e3$$

$$3^2-9$$

와 같은 식들이 모두 가능하다. 표기된 값들은 정수, 소수 그리고 정확한 표기여야 한다.

앞의 예제에서, 첫 번째 표현은 6에 7을 곱한 후 그것을 3으로 나누는 것에 5를 더하는 것이다.(5+14=19) 마지막 표현은 3의 제곱에서 9를 빼는 것이다.(9-9=0) 이러한 표현들은 아래에서 설명하는 순서대로 계산된다.

‘-’와 ‘+’ 연산자는 마이너스(음수)와 플러스 부호로서 작용한다.

2-2
 -2+2
 2+++++++++2
 2-----2

위의 결과는 모두 0이다.

연산의 일반적인 순서는 연산자 우선 순위가 높은 것부터 왼쪽에서 오른쪽으로 진행된다.

- -(마이너스), +(플러스)
- ^
- *, /
- +, -(빼기)
- <, >, <=, >=, =
- and, or

마지막 두 개 연산자는 논리적 표현에서 사용된다. 이런 표현들이 아래에 설명되어 있다.

먼저 계산되어야 할 항목이 있을 때는 괄호를 사용한다. 수학적 분석에서 사용했던 것처럼 괄호로 묶여진 것은 가장 안쪽에서부터 바깥쪽으로 먼저 처리된다. 수식을 이용할 때 처리 순서가 애매한 경우에는 괄호를 사용하는 것이 안전하다.

몇 가지 예를 들어보자.

- -1^2 : 마이너스가 거듭제곱보다 연산 우선 순위가 높으므로 $(-1)^2=1$ 이 된다.
- $-1+-2*3+4$: $-1+-6+4=-3$ 이 된다. 마이너스가 먼저 연산되고 그 다음은 곱하기를 계산한다. 마지막으로 덧셈을 한다.
- $(-1+-2)*(3+4)$: $-3*7=-21$ 이 된다. 마이너스와 플러스가 먼저 계산되고 마지막으로 곱셈을 한다.
- $3*((2+3)*(7+4)+3)$: $3*(5*11+3)=3*58=174$

2) 계열 표현(Series Expressions)

EViews가 무엇보다도 유용한 점은 표현식에 현재의 표본에 있는 계열의 모든 데이터, 요소에 대한 계열 연산이 가능하다는 것이다. 예를 들면, 계열 식 '2*y+3'는 y의 모든 표본값에 2를 곱한 후 3을 더하는 것이다. 또 여러 계열로 작업할 때도 연산을 할 수 있다. 즉, 'x/y+z'는 x의 모든 데이터를 그것에 대응하는 y의 데이터로 나눈 값에 또 거기에 대응하는 z 값을 더한다는 것이다.

2) 계열 함수(Series Functions)

EViews는 함수를 내장하고 있어서 현재의 표본에서 계열의 모든 요소들을 연산할 수 있다. “요약함수(summary functions)”가 새로운 계열을 만들거나 행렬 언어로 작업을 할 때 이용되는 스칼라, 벡터 또는 행렬을 되돌려 주는데 반해 “요소함수(element functions)”는 계열의 각 요소의 값을 되돌려 준다.

EViews에서 대부분의 함수는 @부호로 시작한다. 예를 들면 @mean은 현재 표본에서 계열의 평균값을 계산하고 @abs는 각 데이터의 절대값을 말한다.

모든 요소함수들은 입력값이 존재하지 않거나 유효하지 않을 때, 또는 결과가 정의되지 않았을 때는 NA로 나타난다. 요약정보를 되돌려 주는 함수들은 일반적으로 현재의 표본에 없는 데이터에 대한 연산은 제외된다. 예를 들면 @mean 함수는 missing 값이 없는 표본에서 데이터들의 평균을 계산한다.

■ 예제

내장함수의 몇 가지 예를 들어보겠다.

```
@trend(1980:1)
@movav(x,4)
@mean(x)*y+@var(x)*z
1/@sqrt(2*@acos(-1))*exp(-1/2*x^2)
d(x,4)
```

4) 계열 요소(Series Elements)

한 계열에서 어떤 특별한 데이터를 찾고자 할 때가 있다. EViews에는 이런 기능이 있는데 바로 @elem이다. 이것은 계열의 특별한 값을 이용하는 것이다.

@elem은 두 개의 의미가 있는데 첫째는 계열의 이름이고 둘째는 날짜나 데이터를 확인하는 것이다.

예를 들어, 분기주기의 계열 Y의 1980:3의 값 또는 날짜 순으로 되어있지 않은 계열 X에서 323번째 데이터를 이용하고자 한다. 이 때 함수

@elem(y, 1980:3)

@elem(x, 323)

은 각각의 주기에서 각 계열의 값을 돌려준다.

5) 논리적 표현(Logical Expressions)

논리적 표현은 참과 거짓을 판단하는 것이다. EViews는 논리적 표현에서 missing 값은 허용하지만 이 점에 대해서는 아래의 missing값에 대한 설명을 할 때까지 잠시 미루기로 하자.

논리적 표현은 프로그램에서 수학적 연산, 표본 문장, if 조건문 등의 한 부분으로서 이용된다.

이미 앞에서 표본과 표본개체에 대한 설명에서 이 논리적 표현에 대하여 약간 다루었다. 예를 들어 표본 조건을 들었었는데

ip>100

는 특별한 조건을 만족하는 데이터만을 선택하는 것이다. 이것이 논리적 표현의 예제이다. ip가 100을 초과하는 데이터는 참이고 그렇지 않으면 거짓이다.

좀 더 일반적으로 논리적 표현에는 비교 연산자(<, >, <=, >=, =, <>)가 있다.

위에서 설명한 표본의 예제에서 'and' 와 'or'의 연결 연산자를 사용해서 좀 더 복잡한 논리적 표현을 세울 수 있다.

(ip>100 and educ>=13) or (ip>120)

EViews에서 참값은 1, 거짓값은 0을 이용한다. 이런 내부적 표시는 논

리적 표현을 포함한 복잡한 표현을 만들 수 있다는 것을 의미한다. 예를 들어 데이터를 기억시키기 위한 논리식을 이용할 수 있는데

$$0*(inc<100)+(inc>=100 \text{ and } inc<200)+2*(inc>=200)$$

이것은 inc가 100보다 작으면 0을, inc가 100보다 크거나 같고 200보다는 작으면 1을 그리고 inc가 200보다 크거나 같으면 2를 가져오게 된다.

등호(=) 연산자는 값을 할당하거나 값을 비교할 때 모두 쓰이므로 약간의 설명이 필요하다. 이것은 아래에서 계열 만들기과 수정에 대한 설명을 할 때 좀 더 깊게 다루기로 하자.

$$incm=2000$$

이것은 만일 incm이 정확하게 2000이면 참이고 그렇지 않으면 거짓이다.

6) 결항치(Missing Data)

작업을 하다 보면 어떤 기간은 데이터가 유효하지 않거나 또는 데이터가 없는 경우, 또는 결과치가 정의되지 않아서(즉, 0으로 나누거나 음수의 로그형태 등) 수학적 연산을 수행해야 하는 경우가 있을 것이다. 이 때 EViews에서는 이런 missing 값들을 표시할 때는 NA(Not Available)를 이용한다.

대부분의 경우는 NA에 대해 걱정할 필요는 없다. EViews에서는 적절히 NA를 생성시켜 주고 자동적으로 통계적 계산을 수행하여 NA를 가진 데이터를 제외시킨다. 예를 들어, 어떤 등식을 추정할 때는 EViews는 missing 값이 없는 표본의 데이터 셋을 이용한다.

그렇지만 NA로 작업을 해야 하는 경우도 있을 수 있기 때문에 몇 가지 점들은 유의해야 한다.

다중 계열로 작업을 수행할 때 NA는 두 가지로 처리할 수가 있다. EViews에는 casewise exclusion(일반 표본)과 listwise exclusion(각각의 표본)이 있다. 전자는 모든 계열에 missing값이 없는 경우에만 사용된다. 이 규칙은 항상 적용되는데 등식 추정 같은 경우이다. 후자는 각 계열에서 가능한 한 데이터 수를 최대한 이용하는 것인데 계열목록에서 각 계열에 대하여 따로 따로 데이터를 제외시키는 것이다. 예를 들어 그룹 계열에서 이산 통계량을 계산할 때 각 계열에 대해 다른 표본을 사용하는 선택을 할 수 있다.

만일 작업에서 NA를 직접 이용해야 할 때 EViews의 NA에는 IEEE법

척이 있다는 것을 알아야 한다. 이것은 NA로 수학 연산을 수행하면 missing 값이 산출된다는 것이다. 다음에 나오는 표현들은 모두 missing 값이 생성된다.

```
@log(-abs(x))
1/(x-x)
(-abs(x)^(1/3))
3*+na
exp(x*na)
```

등식(=)과 부등식(<>) 연산자는 na가 어떤 값을 가지는 것으로 처리하여 작업을 수행한다. 다음 표현을 보자.

```
x=5
y<>5
x=na
y<>na
```

위의 식들은 계열에 있는 각각의 값에 따라 참 또는 거짓으로 평가한다.

반대로 NA를 사용한 부등식(<, >) 비교는 논리적으로 정의되지 않기 때문에 항상 NA값을 돌려준다.

```
x>=na
y<na
```

위의 두 표현은 x, y에 상관없이 모든 데이터에 대해 NA가 된다. 비슷하게

```
x>5
```

는 NA가 아닌 x값에 대해 참 또는 거짓을 평가하고 x값이 NA일 경우에는 missing 값이 된다.

수학적 표현에서 논리식을 나타낼 때 NA를 이용하면 missing값으로 처리한다. 예를 들어, 계열 x에 NA가 있는 데이터는 다음의 수학적 표현

```
5*(x>3)
```

에서 결과는 NA가 된다. 그러나 논리식이 표본이나 if 조건문의 일부분으로 이용되었을 때 NA는 거짓으로 인식된다.

```
smpl 1 1000 if x>3
smpl 1 1000 if x>3 and x<>NA
```

위의 두 식은 조건 x>3이 NA값을 검사하는 것을 함축하고 있기 때문에 같은 결과이다. 그러나 다음의 식

```
smpl 1 1000 if x<=NA
```

의 표본에는 아무 것도 없는 결과가 나타난다.

EViews의 이전 버전에서도 한 가지 예외를 제외하고는 EViews 3 버전과 똑같이 missing 값에 대한 동일한 법칙을 따른다. 이전 버전에서는 어떤 숫자에 0을 곱한 결과는 0이었다. 그러나 EViews 3에서는 NA값에 0을 곱하면 똑같이 NA가 된다. 계열 x에 있는 NA를 숫자 y로 바꾸고자 할 때 버전 2에서는 다음과 같은 식이 이용되었다.

$$x=(x<>na)*x+(x=na)*y$$

하지만 버전 3에서는 새로운 함수 @nan을 이용하면 된다.

$$x=@nan(x, y)$$

이렇게 하면 x의 NA값에 y의 값이 기록된다.

예) 프로그램 작성시 시작년도가 다른 3개(A자료: 1970년, B자료: 1971년, C자료: 1973년)의 통계자료들의 표준화증감률을 합하여 종합증감률을 산출할 경우, B자료는 1970년 자료, C자료는 1970~1972년자료가 'NA'로 되어 있어 1970년에서 1972년까지 표준화증감률을 합할 수 없으므로 이 경우 @nan을 사용하면 해결이 가능하다.

(프로그램 예제)

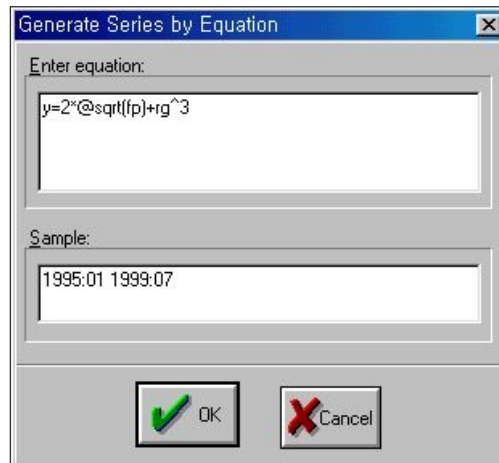
```
series sum_rate=0
for !i=1 to !k step 1
    genr nstrate_co{!i}=@nan(strate_co{!i}, 0)
    genr sum_rate=sum_rate+nstrate_co{!i}
next
```

5.2 계열 작업(Working with series)

제일 중요하게 사용되는 표현중의 하나가 이미 존재하는 데이터로부터 새로운 계열을 생성하거나 계열의 값을 수정하는 것이다. 표본을 조합하여 이용하면 데이터의 복잡한 변환이나 결과를 새로운 또는 이미 있던 계열 개체에 저장할 수가 있다.

계열을 생성하거나 수정하기 위해서는 Quick/Generate Series...를 선택하거나 작업파일 툴바에 있는 Genr을 누른다. 추가 정보를 입력할 창이 나타난다.

<그림 5.1> 계열의 생성



창의 위쪽에 있는 편집상자에 계열 생성을 위한 지시문을 쓰고 아래 편집상자에는 이에 관련된 표본 기간을 입력한다.

지시문은 실제로 반복되는 것으로 표본의 첫 번째 데이터부터 시작하여 계열에 포함할 데이터를 계산한다.

1) 기본적인 지시문(Basic Assignment)

계열 이름을 쓰고 다음에 등호와 식을 쓴다. 표본의 모든 요소에 대해서 등식의 오른쪽에 있는 식을 계산하고 왼쪽에 있는 계열에 그 값들을 할당하게 된다. 필요시에는 계열을 생성하기도 한다.

예를 들어, y 라는 계열이 없었다고 하자.

$$y=2*x+37*z$$

위의 지시문은 먼저 계열 y 를 생성하고 그 안에 모두 NA로 채우게 된다. 그런 다음, 현재 표본에 있는 모든 데이터에 대해 식에서 계산한 값들을 계열 y 에 채운다. 만일 계열 y 가 이미 존재하고 있었다면 현재의 표본에 있는 y 의 데이터를 위의 식에서 계산한 값들로 대체시키게 된다. 표본에 없는 데이터는 바뀌지 않는다.

오른쪽에 있는 표현이 상수일 때는 하나의 특별한 형태로만 일어난다.

$$y=3$$

$$y=37*2+3$$

위의 두 표현은 표본에 있는 모든 데이터를 거치긴 하지만 하나의 상수로 할당된다.

2) 표본 이용하기(Using Samples)

지시문에 사용된 데이터의 표본을 수정하면 Genr 명령어를 여러 번 이용하여 계열들을 서로 연결시킬 수가 있다. 예를 들어 각기 다른 표본의 3개의 지시문을 생성했다고 하자.

위 부분: $y=z$

아래 부분: @all if $z \leq 1$ and $z > -1$

위 부분: $y = -2 + 3 * z$

아래 부분: @all if $z > 1$

위 부분: $y = -0.9 + 0.1 * z$

아래 부분: @all if $z < -1$

여기서 계열 z 에 대한 각각의 선형함수로 y 를 생성할 수가 있다.

if 조건문이나 반복과정을 이용하여 이런 형태의 연산이 가능하므로 되도록이면 Genr과 표본을 지정하는 방법을 이용하기 바란다.

3) 동적 지시문(Dynamic Assignment)

EViews는 표본에 있는 각 데이터에 대해서 지시문을 계산하기 때문에 등호의 오른쪽에 lag값을 이용해서 기능적 지시문을 수행할 수 있다. 예를 들어, 범위가 1979년부터 1998년까지의 연 데이터가 있다고 하자. 이 때

위 부분: $y = y + y(-1)$

아래 부분: 1980 1998

위의 지시문에서 y 계열은 y 의 누적값으로 대체된다. 작업파일에 있는 처음 값은 변환시키지 않기 위해서 표본을 1980년부터 시작했다. 그 때 각 연도에 대해서 y 는 y 의 현재값에 y 의 한 번 lag된 값 즉, y 의 전 값이 더해진다. 이렇게 다음 연도로 이동할 수 있기 때문에 이런 것을 동적 지시문이라고 하며 y 의 값에는 누적값이 들어있게 된다.

이런 절차는 원래의 데이터를 변화시키기 때문에 누적값을 가지는 새로운 계열을 생성하기 위해서는 지시문을 두 단계로 수행해야만 하는데 먼저 원래 계열을 복사한 후에 동적 지시문을 수행해야 한다.

4) 함축적 지시문(Implicit Assignment)

지시문의 왼쪽에 간단한 공식을 사용하여 함축적 지시문을 만들 수가 있다. EViews는 지시문을 검사하여 왼쪽에 있는 첫 번째 계열 이름을 목적계열로서 선택한다. 그런 다음 표본에 있는 모든 데이터에 대해서 기능적 관계를 이용한 값을 할당해 준다. 예를 들어

$$\log(y)=x$$

는 y 를 목적계열로 인식하여 표본에 있는 모든 데이터에 대해서 $y=\exp(x)$ 를 계산하게 된다.

다음은 y 가 목적계열이 되는 예들이다.

$$1/y=z$$

$$\log(y/x)/14.14=z$$

$$\log(@\text{inv}*y)*x=z$$

$$2+y+3*z=4*w$$

$$d(y)=\text{nrnd}$$

일반적으로 EViews는 등호의 왼쪽에 다음과 같은 것들을 이용한 식을 해결할 수 있다.

: +, -, *, /, ^, log(), exp(), sqr(), d(), dlog(), @inv().

Genr은 일반적인 등식을 계산하는 것이 아니기 때문에 등식을 표준화시키지 못하는 경우도 있을 수 있다. 예로서 지시문

$$@\text{tdist}(y, 3)=x$$

는 @tdist()가 EViews가 해결할 수 있는 등식이 아니기 때문에 처리하지 못한다. 비슷하게 EViews는 목적계열이 등호의 왼쪽에 한 번 이상 나타날 때도 처리하지 못한다.

$$x+1/x=5$$

위의 두 경우에는 'Unable to normalize equation'이라는 에러 메시지가 나온다.

목적계열은 등호의 양쪽 모두에 쓸 수가 있다. 예를 들어

$$\log(x)=x$$

은 맞는 지시문이다. EViews는 $x=\exp(x)$ 를 수행하여 x 에는 원래 x 값의 지수값이 할당된다. EViews는 $\log(x)=x$ 를 만족하는 값 x 는 처리하지 못한다.

5) 명령어 창 지시문(Command Windows Assignment)

명령어 창에서도 계열을 생성하여 값을 할당시킬 수 있다. 먼저, 표본 지정을 하고 지시문을 입력한다.

지시문은 여러 가지 형태로 이용할 수 있다. 먼저, 계열이 존재하지 않으면 series나 genr을 입력한 다음에 지시문을 쓴다. 다음의 두 지시문

```
series y=exp(x)
```

```
genr y=exp(x)
```

은 똑같이 계열 y를 생성한다. 일단 계열이 생성되면 계속해서 계열을 생성하는 지시문은 쓰지 않아도 된다.

```
smpl @all
```

```
series y=exp(x)
```

```
smpl 1980 1999 if y>300
```

```
y=y/2
```

■ 예제

다음 예제는 계열을 생성하는 작업을 어떻게 실행하는지에 대한 설명이다.

- 시계열 추세(1997:1은 값이 0이다)

```
series time=@trend(1997:1)
```

- 물가지수(1995:6은 값이 1이다)

```
series index=price/@elem(price, 1995:6)
```

- Dummy 변수

```
series high_inc=income>12000
```

- 논리식을 이용한 재기록

```
series newser=x*(y>30)+z*(y<=30)
```

5.3 자동계열의 작업(Working with Auto-series)

표현식으로 작업하는 또 다른 중요한 방법은 계열의 위치에 식을 이용하는 것이다. EViews가 처리하는 표현의 유용한 도구는 식을 이용하고자 하는 계열의 위치에 대체시킬 수 있다는 것이다. 등식 지정이나 추정, 모형 등에서 계열 개체, 그룹 요소로서 이용되는 계열 등이다

여기서 계열의 위치에 이용할 식의 형태를 자동계열(auto-series)이라고 하는데 식에 포함된 변환은 함축적인 지시문 없이도 자동적으로 계산해 주기 때문이다.

자동계열은 계열의 함수가 실행되는 과정을 보고자 할 때 유용하게 쓰인다. 하지만 변환된 계열을 유지하지는 않는다. 또한 기본 계열 데이터가 자주 변하는 경우에 유용하다. 자동계열식은 기본 데이터가 변할 때마다 자동적으로 재계산 하기 때문에 원래의 데이터는 남아 있지 않게 된다.

1) 자동계열 생성(Creating Auto-series)

자동계열을 생성하고 이용하는 것은 아주 쉽다. 계열 이름을 이용해야 하는 어느 곳에서나 EViews 표현식을 입력하면 된다. 예를 들어, log(IP)의 시계열 그림을 1990:01부터 1998:12까지 그리려고 한다. 이 때 이 값들을 그리는 것은 새로운 계열 즉, LOGIP를 생성하는 것으로 이것의 그림을 그리는 것이다. 새로운 계열을 생성하기 위해서

series logip=log(ip)

작업 영역에서 계열 logip를 열고 View/Line Graph를 선택한다.

또, Show 버튼을 클릭하거나 Quick/Show...를 선택하고 log(ip)를 입력하면 자동계열을 생성하여 이용할 수 있다. EViews는 스프레드시트 상태의 계열 창이 열린다.

<그림 5.2> 자동계열의 생성

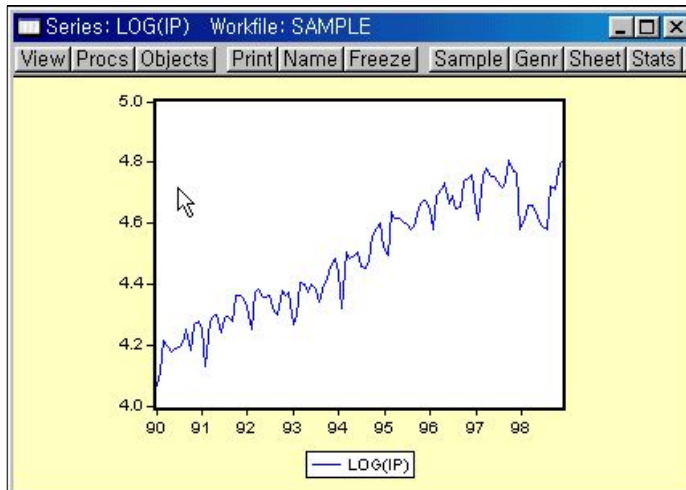
LOG(IP)	
Last updated: 10/18/99 - 13:59	
1970:01	1.223775
1970:02	1.163151
1970:03	1.335001
1970:04	1.360977
1970:05	1.410987
1970:06	1.386294
1970:07	1.386294
1970:08	1.360977
1970:09	1.360977
1970:10	

실제 계열이름의 위치에는 자동계열을 생성하기 위해 입력했던 식이 있게 된다.

또 자동계열도 계열 창으로 인식되어 계열 보기, 절차와 같은 것들이 모두 가능하다.

LOG(IP)의 시계열 그래프를 보려면 View/Line Graph를 선택한다.

<그림 5.3> 자동계열 그래프 작성



다른 모든 보기(view)와 절차(procedure)도 이 메뉴에서 가능하다.

만일 IP 계열의 데이터가 바뀌면 자동계열도 이러한 변화들을 반영해 준다. 예를 들어 IP 계열의 처음 7년간의 데이터에 10을 곱해 보자

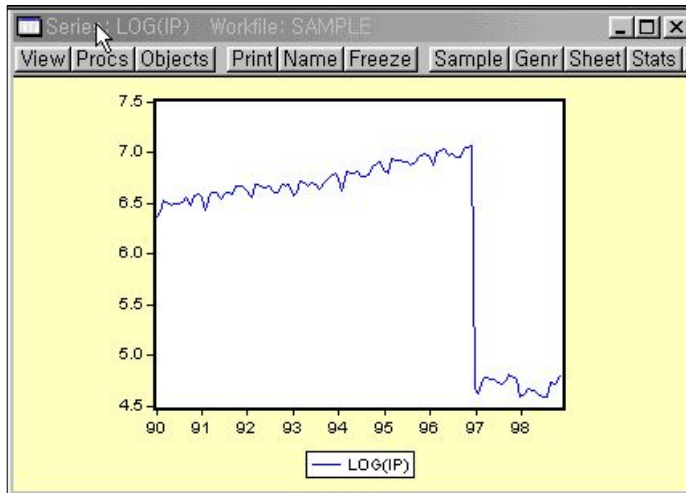
```
smpl 1990:01 1996:12
```

```
ip=ip*10
```

```
smpl 1990:01 1998:12
```

자동계열 그래프는 자동적으로 새로운 데이터를 반영하여 다음과 같이 된다.

<그림 5.4> 새로운 자료를 반영한 자동계열 그래프 작성

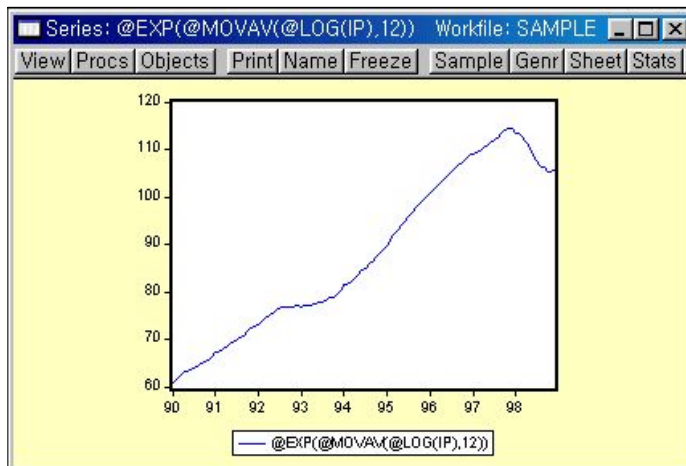


비슷하게, 자동계열로 원래의 IP데이터를 12개월 후행하는 기하학적 이동평균을 계산할 수 있다. 명령어는 다음과 같다.

```
show @exp(@movav(@log(ip), 12))
```

다음 그림은 기하학적 이동평균을 계산한 자동계열을 나타내고 있다.

<그림 5.5> IP데이터를 12개월 후행시켜 이동평균한 그래프



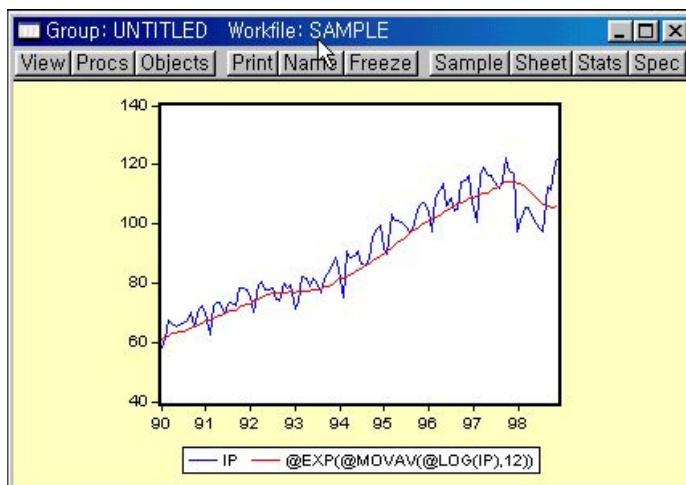
2) 그룹에서 자동계열 이용하기(Using Auto-series in Groups)

자동계열로 작업하는 것 중 더 유용한 것은 그룹에서 작업하는 것이다. 그룹을 생성하고 계열 이름이 놓일 곳에 표현식을 쓰면 된다. Objects/New Object/Group을 선택하고 엔터를 누른다.

```
ip @exp(@movav(@log(ip), 12))
```

위의 식에서 계열 두 개 즉, IP와 기하학적 이동평균을 나타내는 자동계열을 포함하는 그룹을 생성할 수 있다. 이 때 원래의 데이터와 평활화된 데이터를 비교하여 그래프를 그려주는 그룹 개체를 이용할 수 있다.

<그림 5.6> 원래의 데이터와 평활화된 데이터를 비교한 그래프



5.4 그룹 계열로 작업하기((Working with Groups of series))

EViews에는 그룹 개체의 형태로 된 그룹 계열로 작업을 할 수 있는 특별한 기능이 있다. 제4장의 스프레드시트에서 데이터를 불러오기 위해 그룹을 이용했었다. 다시 말해 그룹이란 하나 이상의 계열이나 식의 집합이라고 할 수 있는데 그룹은 각각의 계열에 있는 데이터를 포함하는 것이 아니라 단지 계열에 있는 데이터를 참조하는 것뿐이다.

그룹을 만들 때는 제4장에서 설명한 절차에 따르면 된다. Objects/New Object/Group를 선택하고 계열이름과 자동계열을 입력하여 대화상자를

채운다. 또는 작업파일의 툴바에서 Show를 선택하고 대화상자를 채우면 된다. 또 다른 방법은 명령어 창에 group을 입력한 후 그룹 이름을 지정하고 계열과 자동계열의 이름을 입력하면 된다.

```
group macrolist gdp invest cons
```

이렇게 하면 계열 gdp와 invest, cons를 포함하는 그룹 macrolist가 생성된다. 또,

```
group altlist log(gdp) d(invest) cons/price
```

위의 것은 계열 로그gdp, invest의 1차 차분, cons를 price로 나눈 계열을 포함하는 altlist 그룹이 만들어진다.

그룹(Group)에서 알아 두어야 할 몇 가지 성질이 있다.

- 그룹은 단순히 계열의 목록이지 계열에 있는 데이터를 복사하는 것은 아니다. 그러므로 만약 그룹에 있는 계열의 하나의 데이터가 변하면 그룹에서도 그것이 반영되어 변하는 것을 알 수 있다.
- 만일 작업파일에서 한 계열을 삭제하면 그룹 안에 포함되었던 계열도 사라지게 된다. 그룹에 삭제된 계열만이 존재했었다면 그룹 자체도 없어지게 된다.
- 계열 이름을 바꾸면 그 계열이 포함되어 있는 모든 그룹에도 영향을 미친다.

5.5 스칼라 작업(Working with Scalars)

스칼라 개체는 창 보기(view)가 없다는 점에서 계열이나 그룹과는 다르다. 스칼라는 명령어 형식으로만 만들어진다.

```
scalar scalar_name=number
```

스칼라 이름에 숫자를 할당한다. 숫자는 표현식이나 스칼라 값을 돌려주는 특정한 함수를 이용한다.

스칼라의 내용을 검사하기 위해서는 명령어 show 다음에 스칼라 이름을 입력한다. 그러면 EViews 창의 밑 부분에 있는 상태 표시줄에 스칼라 값이 표시된다. 예를 들면,

```
scalar sf1=@mean(abs(ip))
```

```
show sf1
```

앞의 것은 표준화인자(sf1)를 구하는 방법으로 ip라는 자료에 절대치를 취한 후 평균한 값으로 저장되고 상태 표시줄에 이 값이 표시된다. 또 다른 방법으로 작업 창에서 스칼라 이름을 더블 클릭하면 상태 표시줄에 값이 표시된다.

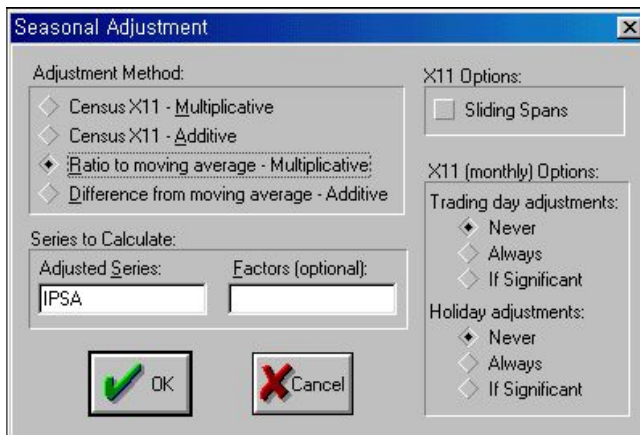
5.6 계절 조정(Seasonal Adjustment)

시계열은 매월 또는 매 분기마다 발생하는 주기적인 움직임을 보인다. 예를 들어 아이스크림 판매는 매년 여름기간에 많이 팔리고 장난감은 매년 겨울 크리스마스 기간에 정점(Peak)을 이룬다. 계절 조정이란 계열에서 이러한 주기적인 움직임을 제거하고 계열을 이루고 있는 추세 요인들만을 추출해내는 과정을 말한다.

계절조정 과정은 분기 또는 월 주기의 계열에서만 가능하다.

예를 들어 a:\data\sample.wf1에서 산업생산지수(IP)를 계절조정하기 위해서는 계열창의 툴바에서 Procs/Seasonal Adjustment...를 클릭하면 다음과 같은 계절조정에 관한 대화상자가 나타난다.

<그림 5.7> 계절조정 대화상자



이 상자에는 다음과 같은 정보를 입력해야만 한다.

- **조정 방법(Adjustment Method)**

: X-11은 미국 Census Bureau에서 사용하는 표준 조정 방법이다. 계절조정하기에 다른 두 가지 방법이 있다.

계절조정방법에는 승법과 가법 두 가지가 있다. 승법은 계열을 추세 요인과 계절요인의 곱으로 적절히 분해하는 것이고 가법은 계열을 이 두 요인의 합으로 분해하는 것이다.

승법(X-11이나 ratio-moving average를 선택)은 계열이 양수일 때만 이용할 수가 있다.

- 계산된 계열(Series to Caculate)

: 이것은 계절조정된 계열의 이름이다. EViews에서는 원래의 이름에 SA를 붙여서 이름을 만들지만 이용자가 원하는 이름으로 변경할 수 있다. 계절 요인 계열도 이름을 지정할 수 있는데 이 경우에는 이것도 작업파일에 저장된다.

1) Census X-11 방법

이 방법(승법 및 가법)은 미국 상무성에서 공식적으로 발표되는 데이터를 계절조정할 때 이용하는 표준 방법이다. X-11은 Census에서 제공한 프로그램이 있고 Eviews 디렉토리에 있는 X11Q2.EXE와 X11SS.EXE를 실행시켜서 설치한다. 이 프로그램은 EViews 디렉토리에서 X11DOC1.TXT부터 X11DOC3.TXT까지의 파일에 설명되어 있다.

X-11 프로그램은 도스나 EViews에서 직접 실행할 수 있다. EViews에서 X-11을 실행할 때는 계절 조정 대화상자에서 X-11을 선택하면 조정된 계열과 요인 계열은 자동적으로 EViews 작업파일 안에 만들어진다. 이 절차의 맨 마지막에는 X-11 요약 출력과 에러 메시지가 나타난다.

X-11 방법에는 선택사항이 많이 있는데 가장 중요한 것은 계절 조정 대화상자이다. 그러나 EViews 대화상자에는 별로 중요하지 않은 선택사항들도 있다. 이러한 선택사항을 이용하고자 할 때는 X-11 프로그램을 도스 명령에서 실행하여야 한다. X-11방법에 대하여 선택이 가능한 것에 대해서는 EViews 디렉토리 안에 X11DOC 텍스트 파일에 설명되어 있다.

계절조정을 할 수 있는 데이터 수는 한계가 있다. X-11으로는 분기와 월 주기 데이터로만 작업할 수 있으며 최소한 4년 이상의 데이터가 있어야 한다. 그리고 월 자료는 20년까지 분기자료는 30년까지 계절조정이 가능하다.

2) X-11방법을 이용한 계절조정

① 명령어 구문

⇒ 계열명.x11(options) 계절조정계열명 계절조정인자명

② 선택사항(Options)

- m: 승법 계절모형
- a: 가법 계절모형
- s: 이동기간(sliding spans)을 사용할 경우(계절조정의 안정성 평가)
- h: 휴일(holidays)조정(승법모형이면서 월데이터인 경우 적용)
- I: 휴일조정이 유효할 경우 조정(승법모형이면서 월데이터인 경우 적용)
- t: 요일(trading days)조정(월데이터일 경우 적용)
- q: 요일 조정이 유효할 경우 조정(월데이터일 경우 적용)
- p: X-11 수행결과를 프린트

③ 예제

- ip.x11(m,h) ipx11 ipfac

설명: 산업생산지수(ip) 시계열을 승법모형(m)을 사용하고, 휴일조정(h)을 하여 계절조정계열은 ipx11, 계절조정인자는 ipfac라는 이름으로 저장

- whole.x11(m,I,q,p) wholx11 wholefac

설명: 도소매판매액지수(whole) 시계열을 승법모형(m)을 사용하고, 휴일조정이 필요한 경우 조정(I)하고, 요일조정이 필요한 경우 실시(q)하여 계절조정계열은 wholx11, 계절조정인자는 wholefac라는 이름으로 저장하며, 계절조정결과를 프린트하라는 의미

3) seas 명령을 이용한 계절조정

: seas 변수는 이동평균의 비율 또는 차를 이용하여 계절조정시 사용하는 방법

① 명령어 구문

⇒ seas(options) 계열명 계절조정계열명 계절조정인자명

⇒ 계열명.seas(options) 계절조정계열명 계절조정인자명

② 선택사항(Options)

- m: 승법모형(이동평균의 비율)
- a: 가법모형(이동평균의 차)

③ 예제

- seas(a) ip ip_adj ip_fac

설명: 산업생산지수(ip)라는 변수를 가법모형(a)을 사용하여 계절조정계열은 ip_adj, 계절조정인자는 ip_fac라는 이름으로 저장

- whole.seas(m) wholesa wholesf

설명: 도소매판매액지수(whole)라는 변수를 승법모형(m)을 사용하여 계절조정계열은 wholesa, 계절조정인자는 wholesf라는 이름으로 저장

5.7 추세치산출(Hodrick-Prescott 필터)

1) Hodrick-Prescott 필터 개념 및 사용법

Hodrick-Prescott 필터는 경제학자들 사이에서 시계열의 장기 추세 요인의 평활화 추정량을 얻기 위해 폭넓게 사용되는 방법이다. 이 방법은 1980년 Hodrick 과 Prescott가 전쟁 이후의 미국의 경기 순환을 분석하기 위해 처음으로 이용하였다.

이론적으로 보면 Hodrick-Prescott(HP) 필터는 우선 시계열 y_t 를 추세 성분 g_t 와 순환성분 c_t 로 분해할 수 있다(물론 $y_t = g_t + c_t$ 이다)라는 가정 하에 평활화 된 계열 g 를 g 주위에 y 의 분산을 최소화하는 양방향 선형 필터이다. g 의 2차 차분을 포함하는 것에 영향을 받는다. 이것은 다음 식을 최소화하는 추세성분 g_t 를 HP 필터로 선택한다.

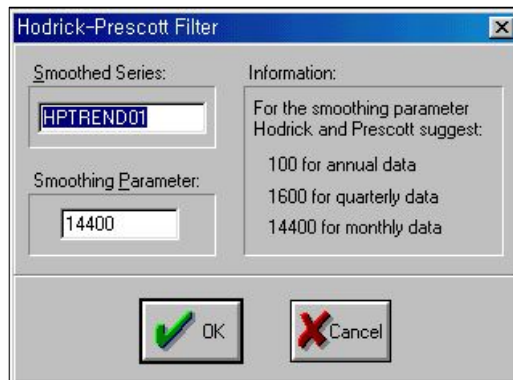
$$\sum_{t=1}^T (y_t - g_t)^2 + \lambda \sum_{t=2}^{T-1} \{(g_{t+1} - g_t) - (g_t - g_{t-1})\}^2$$

모수 λ 는 평활화 모수(smoothing parameter)로서 추세성분 g_t 의 평활성을 조정한다. λ 가 커질수록 g_t 는 평활화 정도가 커진다. λ 가 ∞ 로 가면 선형 추세에 접근한다.

기본적으로 HP 필터는 원계열에서 계절조정 및 불규칙조정 등 비경기적요인을 제거한 후 경기적요인(추세성분, 순환성분)만 가진 상태에서 사용하여야 한다.

HP 필터를 이용해서 계열을 평활화 할 때는 Procs/Hodrick-Prescott Filter...를 선택한다. 다음은 HP 필터의 대화상자이다.

<그림 5.8> Hodrick-Prescott 필터



먼저, 평활화 된 계열에 대한 이름을 지정한다. EViews에서는 이름을 지정하도록 되어 있긴 하지만 이용자가 선택한 이름을 입력해도 된다. 다음으로 평활화 할 모수 λ 로 사용할 정수를 지정한다. EViews에서의 기본 값은 다음과 같다.

- 연간 데이터: $\lambda=100$
- 분기 데이터: $\lambda=1600$
- 월 데이터: $\lambda=14400$

정수 값이 아닌 수치에 대해서는 반올림한다.

OK를 클릭하면 HP 필터로 계산된 값이 원래의 계열과 함께 그려진 그래프가 나타난다. 계산은 현재의 작업파일의 표본에 대해서만 실행된다. 현재의 표본 이외의 평활화 계열은 NA로 채워진다.

2) Hodrick-Prescott 필터를 명령어로 사용하는 방법

예) a:\data\sample.wf1에서 산업생산지수(ip)를 X-11으로 계절조정된 계열인 ipsa를 이용하여 추세변동치 및 순환변동치를 산출하기 위하여 HP 필터를 명령어로 사용하는 과정을 살펴보자

- ㉠ 먼저 EViews화일 a:\data\sample.wf1을 open한다
⇒ open a:\data\sample.wf1
- ㉡ a:\data\sample.wf1에서 ip자료의 x11방법을 이용하여 계절조정계열을 산출하는 명령어
⇒ ip.x11(m,I,q) ipsa
- ㉢ ipsa계열의 HP 필터를 이용하여 추세변동치를 산출하는 명령어
⇒ do ipsa.HPF(14400) hptrend_ip
do 명령어는 사용하지 않아도 무방
- ㉣ 순환변동치를 산출하는 명령어
⇒ genr hpcycle_ip=ipsa/hptrend_ip*100
- ㉤ 추세변동치와 순환변동치를 그룹(group)으로 묶어서 y축을 달리 (crossing)하는 그래프(graph)를 그리는 명령어
⇒ group gp1 hptrend_ip hpcycle_ip
⇒ gp1.line(x)

(참고) 연산함수(Mathematical Operators Functions)

1. 연산자(Operators)

아래에 있는 연산자는 계열이나 스칼라 값을 포함하고 있는 식에서 사용될 수 있다. 계열 표현식에 이용될 때 연산자는 현재의 표본에서 각각의 관측치에 대해서 수행된다. 값을 구하는 절차는 제 5장에서 설명되어 있다. 때에 따라서는 적절하게 괄호를 이용하여 연산순서를 정해주어야 하는 것을 유의하도록 하자.

표시	연 산	설 명	
+	더하기	$x+y$	x와 y의 내용을 더한다
-	빼기	$x-y$	x에서 y의 내용을 뺀다
*	곱하기	$x*y$	x에 y를 곱한다
/	나누기	x/y	x를 y로 나눈다
^	몇승	x^y	x를 y번 곱한다
>	크다	$x>y$	x가 y보다 크면 1, 그렇지 않으면 0값을 취한다
<	작다	$x<y$	y가 x보다 크면 1, 그렇지 않으면 0값을 취한다
=	같다	$x=y$	x와 y가 같으면 1, 그렇지 않으면 0값을 취한다
<>	같지 않다	$x<>y$	x와 y가 같지 않으면 1, x와 y가 같으면 0값을 취한다
<=	작거나 같다	$x<=y$	x가 y보다 크지 않으면 1, 그렇지 않으면 0값을 취한다
>=	크거나 같다	$x>=y$	y가 x보다 크지 않으면 1, 그렇지 않으면 0값을 취한다
and	그리고	$x \text{ and } y$	x와 y가 모두 0이 아니면 1, 그렇지 않으면 0값을 취한다
or	또는	$x \text{ or } y$	x나 y 둘 중에 하나만 0이 아니어도 1, 그렇지 않으면 0값을 취한다

주) 1은 참, 0은 거짓을 의미

2. 날짜와 관측치 함수(Date and Observation Functions)

이 함수는 주어진 관측치에 대한 주기나 주어진 날짜/관측치에 대한 값을 정의하는데 이용된다.

함 수	이 름	설 명
@day	관측일	일 또는 주의 작업파일에 대해 각각의 관측치에 대한 월에서의 관측일을 되돌려 준다
@elem(x,"d"), @elem(x,s)	요소	계열 x에서 관측치나 날짜의 'd'나 문자 's'를 되돌려 준다
@month	관측월	각각의 관측치에 대해서 관측치의 월을 되돌려 준다(월, 일, 주 데이터에 대해서)
@quarter	관측 분기	각각의 관측치에 대해서 관측치의 분기를 되돌려 준다(연간, 반연간, 날짜순이 아닌 데이터는 제외)
@year	관측 연도	각각의 관측치에 대해서 관측치의 연도를 되돌려 준다(날짜순이 아닌 데이터는 제외)

예를 들어 다음과 같은 월주기의 작업파일 계열을 생성했다고 하자.

```
series y = @month
```

이때 y는 각 월에 대한(1부터 12) 숫자 지표를 가지게 된다. 같은 작업 파일에 다음과 같이 계열을 생성하면

```
series z = @quarter
```

계열 z에는 각 관측치와 연결된 분기 지표(1부터 4)로 채워지게 된다.

이 함수들은 적어도 높은 주기를 가진 작업파일에서만 사용이 가능하다. 즉, @month 함수는 주기가 일일 또는 연간 작업파일에서는 이용할 수 있는 반면, @day 함수는 월이나 연간 주기의 작업파일에서는 사용할 수 없고 또 @month 함수도 연간작업파일에서는 사용할 수가 없다.

3. 기본 수학 함수(Basic Mathematical Functions)

이 함수는 기본적인 수학 연산을 수행한다. 이 함수가 계열에 이용되면 현재 표본에 대한 모든 관측치에 대하여 값을 되돌려 주게 된다. 또 행렬 개체에 이용되면 이용된 행렬 개체의 모든 요소들의 값을 되돌려 주게 된다.

다. 입력값이 NA이거나 유효하지 않은 값일 때 이 함수를 이용하면 NA 값을 되돌려 주게 된다. 예를 들어 거듭제곱근 함수 @sqrt 에서는 0보다 작은 모든 관측치에 대해 NA값을 되돌려 준다.

로그형태의 함수는 밑이 'e'(자연 대수)임을 유의하자. 자연 대수를 \log_{10} 으로 변환하려면 다음 관계를 이용하면 된다.

$$\log_{10}x = \log_e x / \log_e 10$$

함 수	이 름	예 제 / 설 명
@abs(x), abs(x)	절대값	@abs(-3)=3
@ceiling(x)	x보다 작지 않은 가장 작은 정수	@ceiling(2.34)=3; @ceiling(4)=4
@exp(x), exp(x)	지수, e^x	@exp(1)≈2.71813
@fact(x)	계승, $x!$	@fact(3)=6; @fact(0)=1
@factlog(x)	계승의 자연 대수, $\log_e(x!)$	@factlog(3)≈1.7918; @factlog(0)=0
@floor(x)	x를 넘지 않는 가장 큰 정수	@floor(1.23)=1; @floor(3)=3
@inv(x)	역수, $1/x$	@inv(2)=0.5
@mod(x,y)	나머지	x/y의 나머지를 x와 같은 부호값으로 돌려준다. 만일 y=0이면 결과도 0
@log(x), log(x)	자연 대수, $\log_e(x)$	@log(2)≈0.693; log(2.71813)≈1
@nan(x,y)	x가 NA일 때 y로 대체	x가 NA가 아니면 x, NA이면 y
@recode(s,x,y)	조건에 따라 대체	조건 s가 참이면 x, 그렇지 않으면 y; @recode(y>0, x, 2)
@round(x)	가장 가까운 정수로 반올림	@round(-97.5)=-98 @round(3.5)=4
@sqrt(x), sqr(x)	제곱근	@sqrt(9)=3

4. 시계열 함수(Time Series Functions)

다음 함수는 시계열 데이터 작업을 할 때 유용하다. 후행된 값이 유효하지 않은 값일 때는 관측치에 대해 NA값이 되돌려 준다. 예를 들어 $d(x)$ 는 후행된 값이 유효하지 않기 때문에 작업파일의 첫 번째 관측치에 대해서는 결측값(missing value)을 되돌려 준다.

함 수	이 름	설 명
$d(x)$	1차 차분	$(1-L)X = X - X(-1)$, L 은 후행 연산자
$d(x,n)$	n 차 차분	$(1-L)^n X$
$d(x,n,s)$	계절차분 s 와 n 번 차분	$(1-L)^n (1-L^s)X$
$dlog(x)$	로그의 1차 차분	$(1-L) \log(X) = \log(X) - \log(X(-1))$
$dlog(x,n)$	로그의 n 차 차분	$(1-L)^n \log(X)$
$dlog(x,n,s)$	로그의 계절차분 s 와 n 번 차분	$(1-L)^n (1-L^s) \log(X)$
@movav(x,n)	n 만큼 후행하여 이동평균	@movav(x,3) = $(X + X(-1) + X(-2))/3$
@movsum(x,n)	n 만큼 후행하여 이동합계	@movsum(x,3) = $(X + X(-1) + X(-2))$
@pch(x)	한 기간 변화율 (소숫점)	$(X + X(-1))/X(-1)$
@pcha(x)	한 기간 변화율 -연간화(소숫점)	@pcha(x) = $(1 + @pch(x))^n - 1$, n 은 1년에 연관된 기간(분기데이터일 경우 $n=4$)
@pchy(x)	일 년 변화율 (소숫점)	$(X + X(-n))/X(-n)$, n 은 1년에 연관된 기간(연간 데이터일 경우 $n=12$)
@seas(n)	계절 더미	분기 또는 월이 n 과 같으면 1, 그렇지 않으면 0값을 돌려준다
@trend, @trend(n)	시간 추세	날짜나 관측수 n 기간을 0으로 정규화한 추세 계열을 되돌려 준다 만일 n 이 생략되면 계열은 작업파일에서 첫 번째 관측치에 정규화된다

6. EViews 데이터베이스

EViews 데이터베이스는 작업파일이 EViews 개체의 집합을 포함하는 것과 비슷하다. 그러나 두 가지 점에서는 작업파일과 다르다. 하나는 작업파일과 달리 그 안에 있는 개체에 접근할 때 메모리에 데이터베이스 전체가 로드되지 않아도 된다. 하나의 개체는 디스크에 있는 데이터베이스에 곧바로 저장하거나 불러 올 수가 있다. 둘째, EViews 데이터베이스에 있는 개체는 하나의 주기나 범위에 제한 받지 않는다. 데이터베이스는 연간, 월, 일일 계열, 다른 숫자의 데이터 모두를 포함할 수 있다.

데이터베이스는 어떤 특정한 계열이나 같은 성질의 계열들을 데이터베이스에서 찾고자 할 때 검색 기능을 이용할 수 있다는 점에서도 작업파일과 다르다. 이 점 때문에 데이터베이스에서 많은 양의 데이터를 처리할 수가 있다.

EViews에는 또한 데이터베이스 링크가 있다. 이것은 데이터베이스와 비슷한 기능을 하는데 데이터 자체를 포함하지 않고 서버에 있는 데이터에 연결할 수 있도록 해 준다. EViews는 Standard & Poor's/DRI에 데이터베이스 링크를 만들어서 EViews의 엄청나게 많은 데이터에 직접 접근할 수 있도록 해 준다.

6.1 개 요

데이터베이스는 EViews 개체의 집합을 포함하는 파일 집합이다. 이 장에서는 다음을 처리하는 방법을 설명한다.

- 새로운 데이터베이스 만들기
- 데이터베이스에 있는 개체를 작업파일로 불러오기, 저장, 복사하기, 새 이름 지정하기, 삭제 등 데이터베이스에서의 개체 작업
- 데이터베이스에서 작업파일에 있는 데이터를 복사하지 않고 직접 데이터로 작업할 수 있는 자동계열 이용하기
- 데이터베이스 이름이 긴 경우에는 단축하여 만들 수 있는 데이터베이스 등록 방법과 작업파일에 없는 계열 이름을 찾는 경로 지정하기
- 특정한 성질을 가진 개체들을 찾기 위해 데이터베이스에서 검색 기능 실행하기
- 포장, 복사, 보수 등 데이터베이스 유지하기

6.2 데이터베이스 기초(Database Basics)

1) EViews 데이터베이스란 무엇인가?

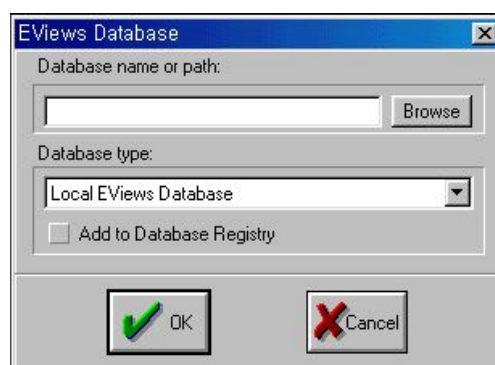
EViews 데이터베이스는 디스크에 있는 파일의 집합으로 구성되어 있다. 여기에는 확장자가 .EDB인 주요 파일이 있는데 이 파일에 실제의 개체 데이터가 들어 있고 데이터베이스에서 개체를 찾을 때 좀 더 빨리 찾을 수 있도록 하는 .E0, .E1A, .E1B와 같은 확장자를 가진 인덱스 파일들이 있다. 보통은 EViews가 자체적으로 이용자를 위해서 처리를 해 주기 때문에 이 구조에 대해 그렇게 자세히 알 필요는 없다. 그렇지만 EViews의 외부에서(윈도우 탐색기 등을 이용해서) 데이터베이스를 복사, 이동, 새이름 지정, 삭제를 할 때는 주요 데이터베이스 파일과 데이터베이스와 관련된 모든 인덱스 파일들에 대해서 작업을 해야한다.

데이터베이스는 메모리보다는 디스크에 보관하는 것이 좋다. 데이터베이스에 어떤 변화가 생기면 바로 여기에 관련된 디스크에도 반영을 해주어야 한다. 그러므로 작업파일과 다르게 일단 데이터베이스에 변화가 일어나면 다시 처음상태로 돌리거나 복구하기는 어렵다. 그렇기 때문에 데이터베이스를 수정할 때는 신중해야 하고 자주 수정하는 데이터베이스는 규칙적으로 백업을 해 놓는 것이 중요하다.

2) 데이터베이스 만들기(Creating a Database)

데이터베이스를 만들 때는 File/New/Database...를 메인 메뉴에서 선택하면 다음의 대화상자가 나타난다.

<그림 6.1> 데이터베이스 생성



로컬 데이터베이스를 만들 때는 'Database name or path'라고 표시된 박스에 데이터베이스 이름을 입력하고 OK버튼을 누른다. 그러면 현재 EViews 경로에 새로운 데이터베이스가 만들어진다.

만약 다른 디렉토리에 데이터베이스를 만들고 싶으면 데이터베이스 이름 앞에 원하는 위치의 경로를 적어주면 된다. 이것은 현재 경로의 하위 디렉토리로 상대 경로(ex: new/test)를 지정할 수도 있고 또 특정한 절대 경로에 데이터베이스를 만들고자 할 때는 절대경로(ex: d:\db\tjdb)를 입력해 주면 된다.

데이터베이스 형식(Database Type)란에서는 다른 형태의 데이터베이스를 만들 수가 있다.

등록기에 추가(Add to Database Registry)는 데이터베이스 등록기에 새로운 데이터베이스를 추가로 등록할 때 편리하다.

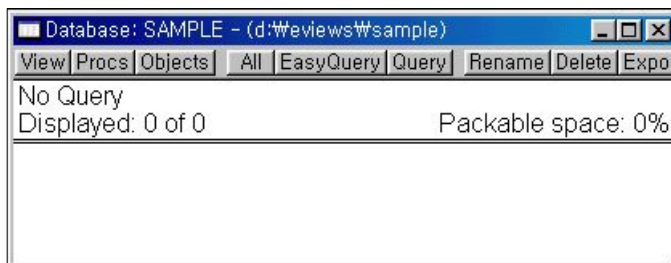
데이터베이스는 명령어 줄이나 명령어를 이용하는 프로그램으로도 만들 수가 있다.

dbcreate db_name

위와 같은 식으로 db_name이라는 데이터베이스를 만든다.

3) 데이터베이스 창(The Database Window)

<그림 6.2> 데이터베이스 창



새로운 데이터베이스를 만들면 데이터베이스 창이 화면에 나타난다. 데이터베이스 창에서는 작업과일에 복사하여 붙이기, 데이터베이스의 기본적인 유지, 데이터베이스 검색기 등을 실행할 수 있다. 많은 데이터베이스 작업은 데이터베이스 창을 먼저 열지 않고서도 실행할 수 있다.

이미 생성한 데이터베이스를 열 때는 메인 메뉴에서 File/Open/Database...를 선택한다. 지정한 디렉토리에 있는 확장자 .EDB를 가진 모

든 데이터베이스 파일이 대화상자에 나타난다. 열고자 하는 파일을 선택하고 Open 버튼을 클릭한다. 열린 데이터베이스 창을 나타내는 새로운 창이 열린다.

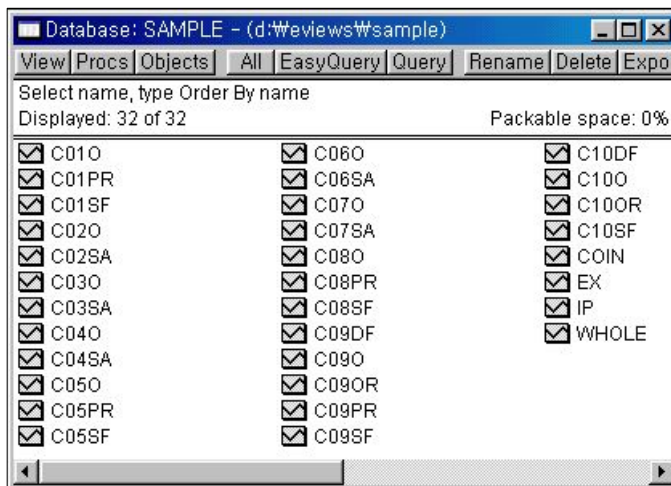
명령어 줄이나 프로그램에서 다음과 같이 하면 데이터베이스 창을 열 수 있다.

```
dbopen db_name
```

작업파일 창과 다르게 데이터베이스 창이 처음 열리면 데이터베이스에 몇 개의 개체가 있다는 것은 표시하지만 그 내용은 표시하지 않는다. 창의 두 번째 줄은 데이터베이스에 저장된 전체 개체 중에서 현재 표시된 개체의 숫자를 말한다.(창이 처음 열렸을 때는 '0'이다.)

All 버튼을 누르면 작업파일에 있는 모든 개체를 알파벳 순서로 나타내 준다.

<그림 6.3> All버튼을 누른 상태의 데이터베이스



작업파일과 마찬가지로 각 개체들은 그 형식을 나타내는 작은 아이콘으로 표시되어 있다. All을 눌렀을 때는 아무 것도 볼 수 없지만 그 개체를 더블 클릭하면 개체가 포함하고 있는 이름, 형태, 수정된 날짜, 주기, 시작과 끝 날짜(계열일 경우), 라벨 등을 볼 수 있다.

데이터베이스가 아주 클 경우에 All 버튼을 누르면 너무 많은 개체를 보여주기 때문에 각 개체에 대한 충분한 정보를 볼 수 없다. 아래에 개체를 간략하게 보여주도록 조절할 수 있는 데이터베이스 검색기에 대한 설

명이 있다. 현재 표시되고 있는 검색기의 텍스트 형식은 항상 데이터베이스 창의 맨 윗줄에서 볼 수 있다.

데이터베이스 창에 있는 Packable space에는 데이터베이스에서 사용되지 않은 공간을 퍼센트로 표시하는데 이것은 데이터베이스 압축(pack) 작업으로 회복할 수 있는 공간이다.

- 참고: EViews에서 데이터베이스 창이 열려 있다고 해서 파일도 열려 있는 것은 아니다. EViews에서는 파일이 작업중일 때만 파일을 연다. 결과적으로 여러 명의 사용자가 동시에 데이터베이스를 열 수 있고 동시에 작업을 수행할 수 있는 것이다. 그러나 다른 사용자가 데이터베이스에 쓰기를 하는 동안에는 데이터베이스에서 읽어 올 수가 없다. 하지만 EViews는 이런 상황을 검사하여 데이터베이스에 쓰기를 할 수 있을 때까지 기다렸다가 다시 시도하게 된다. 만일 데이터베이스가 지정 시간 내에만 이용 가능하다면 이 때 데이터베이스에 공유 위반(sharing violation)이 일어났다는 에러 메시지를 나타낸다.

4) 기본 데이터베이스(The Default Database)

일반적인 실행을 하기 위해서 EViews는 ‘기본 데이터베이스’라는 개념을 이용한다. 기본 데이터베이스란 몇 군데에서 이용되는 것으로 데이터베이스가 명확하게 지정되지 않았을 때 저장하거나 불러오는 실행을 하기 위한 기본 자료나 목표로서 아주 중요하다.

기본 데이터베이스는 새로운 데이터베이스 창을 열거나 화면에 여러 개의 데이터베이스가 열려있는 경우 이미 열려 있는 데이터베이스 창을 클릭함으로써 지정한다. 기본 데이터베이스의 이름은 주요 EViews 창의 맨 아래에 있는 상태표시줄에 나타난다. 현재 열려 있는 데이터베이스가 없을 때도 기본 데이터베이스는 존재한다는 한 가지만 제외하고 개념은 현재 작업파일과 비슷하다. 작업파일에서는 열린 작업파일이 없을 때에는 현재 작업파일이 아무 것도 없음(‘none’)으로 표시된다.

5) EViews.DB? 파일

여기서 설명하는 데이터베이스 성질은 버전 3에서 새로운 것들이다. 이전의 버전이나 MicroTSP에서는 데이터베이스 지정에 관한 많은 제약이 있었다. 개체는 디스크에 하나의 파일에 하나의 개체가 개별적으로 저장

될 수 있었다. 디스크 디렉토리는 하나의 데이터베이스로 사용되었고 그 데이터베이스 항목은 자체의 파일을 가졌다. 이런 파일들은 계열에 대해서 .DB 확장자를 가졌고 그 뒤에 개체의 각 형태에 따른 하나의 문자를 추가할 수가 있었다. EViews는 이런 것을 .DB? 파일로서 참조할 수가 있었다.

이에 새로운 기능이 EViews에 추가되어 데이터를 구축하고 처리하기가 쉬어졌는데 .DB?기능은 계속 유지하면서 또한 데이터를 다른 프로그램에 보급할 수 있는 편리한 기능이 그것이다. 계열 .DB파일은 TSP, RATS, SHAZAM 등의 많은 프로그램에서 지원된다. 또한 NBER(National Bureau of Economic Research)과 같은 기구에서는 데이터를 .DB형식으로 보급하고 있다.

6.3 데이터베이스에서의 개체작업 (Working with Objects in Database)

데이터베이스는 간단히 말해서 다른 EViews 개체들을 담고 있는 용기라고 말할 수 있는데 데이터베이스를 이용해서 하는 작업 대부분이 데이터베이스로 불러오거나 또는 여기에서 내보내는 작업이다. 개체 저장, 불러오기와 내보내기에 관한 것은 다른 방법으로 논하기로 한다.

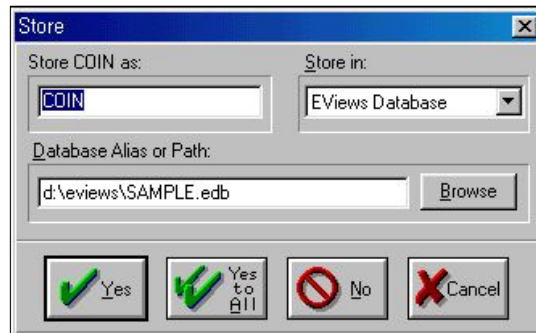
작업을 하다보면 데이터베이스에 있는 개체를 다루어야 할 필요가 있다. 개체를 복사하거나 이름을 새로 지정하고 또 개체를 데이터베이스에서 완전히 지워버리는 일을 할 수 있는데 이 장에서는 복사, 새이름 지정, 삭제에 관한 실행이 어떻게 되는 건지에 대한 설명을 한다.

데이터베이스간에 계열을 이동시키고 작업파일에 있는 주기와 맞추기 위해서 데이터의 주기를 수시로 바꿔주어야 한다. 이 장에서는 이런 것들에 대한 규칙을 설명한다.

1) 데이터베이스에 개체 저장하기

한 개체는 다양한 방법으로 데이터베이스에 저장할 수 있다. 열어 놓은 작업파일이 있고 그 작업파일 안에 있는 개체를 데이터베이스에 저장하고자 할 때는 마우스로 작업파일 창에서 개체를 선택하고 작업파일 툴바에 있는 Store를 클릭한다. 대화상자가 나타나면 선택한 개체에 대하여 이름을 지정하고 저장할 경로를 지정해 준다.

<그림 6.4> 데이터베이스에 개체 저장하기

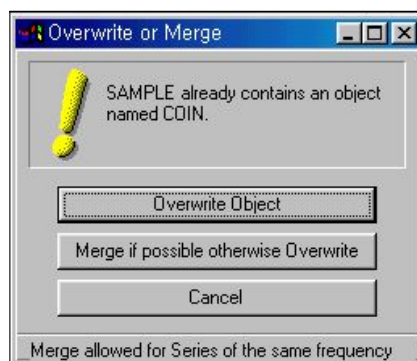


기본적으로, 개체는 작업파일에 있는 이름과 같은 이름으로 현재의 기본 데이터베이스에 저장된다. 이 개체를 저장할 때는 Yes를 클릭하고 대화상자에서 지정한 형태로 선택한 모든 개체를 저장하고자 할 때는 Yes-to-All을 클릭한다.

만일 개체를 다른 이름으로 저장하고 싶을 때는 ‘Store object_name as:’(object_name은 현재 저장할 개체의 이름(COIN)이다.)라고 되어 있는 편집상자에 기존의 개체 이름 위에 원하는 새이름을 입력한다. 또 개체를 다른 데이터베이스에 저장하고자 할 때도 같은 방식으로 ‘Database Alias or Path:’란에서 새로운 데이터베이스 이름을 입력하거나 Browse를 클릭하여 원하는 데이터베이스를 찾는다. 개체를 EViews .DB? 파일로 디스크에 저장하고자 할 때는 오른쪽 방향 화살키를 눌러 ‘Store in:’이라고 되어 있는 부분을 클릭한 후 ‘Individual .DB? files’를 선택한다. 이 때 ‘Path for DB files’라는 곳에 파일이 위치할 경로를 지정해야 한다.

데이터베이스에 이미 같은 이름의 개체가 있을 때는 다음과 같은 화면이 나온다.

<그림 6.5> 데이터베이스에 같은 이름의 개체가 있을 경우



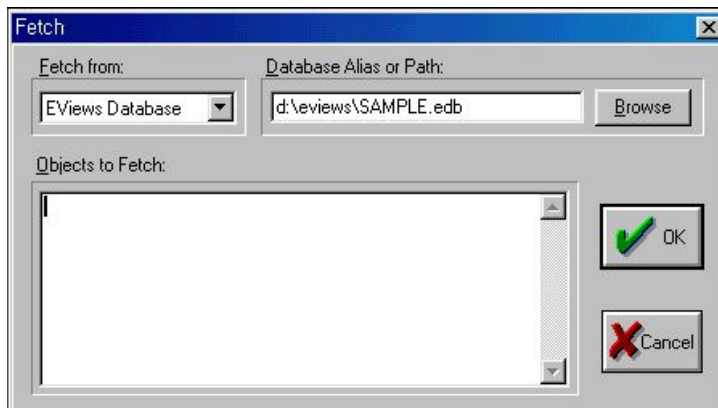
위의 세 가지 선택사항 중 첫 번째와 마지막은 자기 설명적이므로 알 수 있는 것이다. 두 번째는 작업파일로부터 저장하고자 하는 개체와 데이터베이스에 이미 존재하는 데이터베이스 두 개가 모두 같은 주기를 가지고 있을 때만 이용한다. 이 경우, EViews는 원래 데이터베이스에 존재하던 계열과 작업파일로부터 저장하고자 하는 두 계열을 합하여 데이터베이스에는 새로운 계열이 생기게 된다. 예를 들어 데이터베이스에 1970년 1월부터 1990년 12월까지의 월별 데이터가 있고 또 새롭게 이 데이터베이스에 저장하고자 하는 계열은 1990년 1월부터 1999년 8월까지의 월데이터가 있다고 하자. 이 때 새롭게 생기는 계열은 1970년 1월부터 1999년 8월까지의 월별데이터를 갖게 되는데 데이터의 내용은 이미 존재하던 계열이 1970년 1월부터 1990년 12월까지이고 1990년 1월부터 1999년 8월까지는 새롭게 덧붙여진 계열의 데이터이다. 중복되는 부분은 새롭게 덧붙여진 계열이 우선 순위를 갖게 된다.

2) 데이터베이스로부터 개체 불러오기 (Fetching objects from the Database)

데이터베이스에 있는 개체를 불러오는 방법에는 여러 가지가 있는데 대부분이 저장할 때의 방법과 비슷하다.

첫 번째 방법은 개체를 불러들이고자 하는 작업파일의 툴바에서 Fetch를 클릭하는 것이다. 이 때 저장할 때와 비슷한 대화상자가 나타난다.

<그림 6.6> 데이터베이스의 개체 불러오기



대화상자에 불러 올 개체들의 이름과 그것을 찾기 위한 데이터베이스나 디렉토리를 지정한다.

‘Object to Fetch’란에 불러올 개체들의 이름을 입력한다. 또는 마우스로 Fetch를 누르기 전에 작업파일 창에서 개체를 선택하면 개체들의 이름이 자동적으로 나타난다. ‘Database Alias or Path’와 ‘Fetch from’은 저장할 때의 대화상자와 똑 같지만 한 가지가 다르다. 또 Fetch from에는 ‘EViews Database’ 와 ‘Individual .DB? files’외에도 ‘Search Database’라는 선택사항이 있는데 이 것은 EViews가 지정한 이름과 맞는 개체를 여러 개 찾을 때 사용할 수 있다. 이 선택사항을 이용할 때는 먼저 데이터베이스 등록기에 찾는 순서를 지정해야 한다. 이것은 아래에서 데이터베이스 등록기를 참고하자.

OK를 클릭하면 EViews는 모든 개체를 불러온다. 불러오고자 하는 개체가 이미 작업파일에 있을 때는 그 개체를 대체할 것인지를 묻는 메시지가 대화상자에 나타난다. 작업파일에 있는 개체를 대체할 때는 Yes, 작업파일에 있는 개체를 대체하지 않고 그냥 둘 거라면 No를 클릭한다.

작업파일은 주기와 범위가 정해져 있기 때문에, 작업파일로 계열을 불러올 때 작업파일에 있는 주기와 맞추기 위해서 계열의 데이터가 수정되는 경우도 있다. 그러므로 작업파일로 계열을 로드하고 작업한 후 다시 데이터베이스에 저장했을 때 데이터베이스에 저장한 계열이 잘리거나 주기가 변환되는 것은 이런 이유 때문이다. 그러므로 이러한 작업을 할 때는 주의해야 한다.

작업파일의 툴바에 있는 Object/Update selected from DB...는 Fetch와 똑 같은데 불러온 파일로 덮어 쓸 것인지를 묻는 메시지가 없다는 점만 다르다. 데이터베이스에 있는 개체가 작업파일에 있는 것과 같은 형태라면 자동적으로 덮어쓰게 되지만 다른 형태라면 불러오기가 수행되지 않는다. 업데이트는 각각의 개체 창에 있는 Objects에서 할 수 있다.

3) 데이터베이스 내보내기(Database Export)

데이터베이스 창에 있는 데이터를 작업파일로 옮길 수도 있다. 열린 데이터베이스 창에서 마우스로 복사하고자 하는 개체를 선택한 후 데이터베이스 창의 맨 위 부분에 있는 툴바에서 Export를 클릭한다. 데이터베이스 내보내기라는 제목의 대화상자가 화면에 나타난다.

<그림 6.7> 데이터베이스 내보내기(Export)



Workfile for selected objects 필드의 콤보 박스를 클릭하면 현재 열려져 있는 모든 작업파일의 목록이 나타난다. 개체를 복사하고자 하는 대상 작업파일을 선택한다. OK를 누르면 선택한 개체가 지정한 작업파일로 복사된다.

열려진 작업파일의 목록에서 'New Workfile'이라는 선택사항이 있다. 이것은 선택한 개체를 포함하는 새로운 작업파일을 만들 수가 있음을 의미한다. 기본 주기는 선택한 개체들 중에서 제일 낮은 것으로 지정되고 범위는 개체에 있는 데이터를 모두 포함할 수 있는 것으로 지정된다. OK를 누르면 새로운 작업파일 창이 열리고 거기에 선택한 개체를 복사하는데 필요시에는 주기변환도 수행된다.

4) 개체 복사(Copying Objects)

EViews에서는 개체를 이동하는 방법으로 위에서 설명한 것 이외에도 EViews 상자 개체(작업파일이나 데이터베이스)간에 복사와 붙이기를 이용할 수 있다. 이 기능을 이용하면 두 개의 데이터베이스나 작업파일간에 개체 이동을 할 수 있고, 그 안에 있는 개체를 복사하여 똑같은 개체를 만들 수도 있으며 저장과 불러오기의 또 다른 방법으로서 이용할 수가 있다.

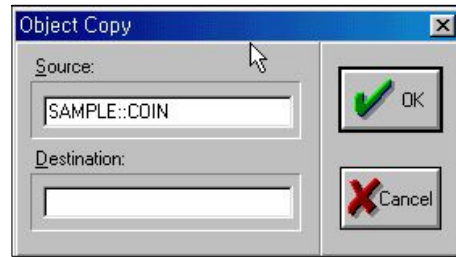
두 개의 상자 개체간에 개체를 복사하는 것은 개체가 들어 있는 상자 개체의 형태가 무엇이든 상관없이 비슷하다. 작업을 하기 전에는 반드시 두 개의 상자 개체가 모두 화면에 열려 있어야 한다. 개체를 복사하고자 하는 원본 상자 개체에서 원하는 개체를 선택하고 메뉴에 있는 Edit/Copy를 클릭한다. 그 다음 복사하여 붙일 대상 상자 개체를 클릭한 후 메뉴에서 Edit/Paste를 선택한다.

두 상자 개체의 형태에 따라 하나나 그 이상의 대화상자가 나타난다.

불러오기와 저장하기를 할 때는 작업파일 창에서 툴바를 이용해서 실행할 때와 같은 대화상자가 나타난다.

상자 개체 안에 개체를 복사할 수 있는 것뿐만 아니라 개체 복사를 이용해도 가능하다. 메인 메뉴에서 Object/Copy(이것은 Object/Copy selected...로 나타남)를 선택하면 다음과 같은 개체 복사의 대화상자가 나타난다.

<그림 6.8> 개체 복사



원본(Source) 필드에는 복사하고자 하는 개체들을 입력하고 대상(Destination) 필드에는 복사한 파일들에 대한 이름을 지정하여 준다.

원본 필드에는 확장자까지 포함하여 입력해야 한다.

source_db::source_pattern

여기서 'source_db::'는 선택사항으로 개체를 복사할 데이터베이스를 말한다.(만약 데이터베이스 이름이 없으면 기본 작업파일에서 개체를 선택하게 된다.) 그리고 'source_pattern'은 개체 이름이나 범용 문자를 포함한 형식으로 ?를 사용하면 하나의 문자와 대응시키고 또 *를 사용하면 모든 문자를 대응시킨다.

대상 필드에도 확장자를 포함해서 입력해야 한다.

dest_db::dest_name

여기서 'dest_db::'도 선택사항이며 개체를 복사시킬 대상 데이터베이스를 말한다.(이때도 데이터베이스 이름이 없으면 대상은 기본 작업파일이다.) 그리고 dest_name도 선택사항인데 복사한 개체에 대해 새로운 이름을 지정해 주는 것이다. 만일 이름을 지정하지 않으면, 개체는 원본파일에 있는 개체 이름과 똑같은 것으로 복사된다.

예를 들어, 데이터베이스 DB1에서 DB2로 개체를 복사하는데 원본 개체의 이름을 그대로 유지하려면 대화상자에 다음과 같이 입력하면 된다.

원본 : db1::object_name

대상 : db2::

여기서 object_name은 EViews에서 표시되는 원래의 이름이다.

데이터베이스 DB1에 있는 개체 중에서 문자 X로 시작되는 모든 개체를 현재의 작업파일로 복사하려고 한다면 다음과 같이 한다.

원본 : db1::x*

대상 :

데이터베이스 DB1안에 ABC라는 개체를 XYZ라는 이름으로 복사하고자 할 때는 다음과 같다.

원본 : db1::abc

대상 : db1::xyz

원본 파일이름에 범용문자를 사용하고 대상파일에는 같은 이름을 사용할 때는 범용문자가 포함된 파일이 대상파일에 있는 경우에도 모두 복사를 하기 때문에 조심해야 한다. 범용문자를 이용해서 개체의 집합을 복사할 수는 없고 새로운 이름을 동시에 지정할 수도 없다.

5) 데이터베이스에서 개체 삭제

데이터베이스 안에 있는 개체 삭제는 열려진 데이터베이스 창에서 개체를 선택하고 데이터베이스 창의 툴바에서 Delete를 클릭한다. delete 명령어를 이용할 수도 있다. 이것에 대해서는 'Command and Programming Reference'를 참고하기 바란다.

6) 데이터베이스에서 개체 새이름 지정하기

데이터베이스안에 있는 개체에 새이름을 지정하고자 할 때는 열려진 데이터베이스 창에서 개체를 선택한 후 데이터베이스 툴바에 있는 Rename을 클릭하면 된다. 이때 새 이름이나 개체의 형태를 입력할 수 있는 대화상자가 나타난다. 한 번에 하나의 개체만을 새이름으로 지정할 수 있다. rename 명령어를 이용해도 된다.

7) 주기 변환(Frequency Conversion)

이미 앞에서 언급했듯이, 데이터베이스 안에는 서로 다른 주기를 갖는 많은 개체들이 있다. 하지만 데이터로 작업을 할 때는 모든 데이터가 공통된 주기를 갖는 작업파일을 구성할 필요가 있다. 주기가 서로 맞지 않는 작업파일을 불러오거나 복사할 때 EViews는 자동적으로 작업파일의 주기가 맞도록 데이터를 변환시켜준다.

주기 변환에는 중요한 두 가지 형태가 있다. 만일 계열이 작업파일(작업파일이 계열보다 더 빈번하게 발생한다.)보다 더 낮은 주기를 가지고 있다면 이 계열에 있는 데이터 값을 작업파일 계열에 대응되는 데이터 값으로 다시 만든다. 예를 들어 분기의 작업파일에서 연간 계열을 불러오면 각 연도의 4분기가 모두 원래 계열의 연간값으로 채워진다.

계열이 작업파일보다 높은 주기를 갖고 있으면 여러 변환방법 중에서 선택해야 하는데 다음은 주요 6가지 방법들이다.

- 평균
- 합계
- 첫번째 데이터
- 마지막 데이터
- 최대값
- 최소값

또 하나의 선택사항이 있는데 계산을 할 때 missing 값을 어떻게 처리할 것인지를 결정하는 'Conversion propagates NAs'이다. Propagate NAs를 검사하면 missing 값이 나타나는 곳에서 계산이 이루어지면 그 기간의 결과는 NA가 된다. 하지만 검사를 하지 않으면 missing 값을 무시하고 계산을 한다.(기간의 값이 모두 missing이라도 그 기간에 대한 결과는 여전히 NA가 된다.)

예를 들어, 첫 번째 데이터로 모두 채우는 것을 선택하고 분기의 작업파일에서 월 주기의 계열을 불러온다면 첫 번째 분기 값은 해당하는 연도의 1월 1일 데이터가 될 것이다. 1월 1일의 데이터가 없고 NAs propagate를 선택했다면 분기값에 전부 NA가 채워진다. NAs propagate를 선택하지 않으면 1월 1일과 3월 31일 사이에 이용 가능한 값이 할당된다.

날짜로 되지 않은 계열을 날짜로 된 작업파일로 불러오거나 복사할 때는 아무런 변환 없이 첫 데이터부터 시작하여 순서대로 실행된다.

다음의 세 가지 방법으로 변환을 적용할 수 있다. 1.명령어 선택, 2.계열에 지정, 3.전체적으로 선택사항 지정하는 것이다. 지정은 다음에 나오는 순서에 따라 적용된다.

- 명령을 이용해서 복사와 불러오기를 할 때는 변환방법 지정을 선택적으로 할 수 있고 EViews는 명령에 나열된 모든 계열에 대해 이 방법을 이용한다.
- 그렇지 않고, 계열이 특별한 주기 변환에 대한 선택으로 지정되었다면(계열 개체 메뉴에서 보거나 선택사항을 바꾸기 위해 View/Conversion Options...를 선택) 그 변환방법은 특정한 계열에 대해서 실행된다.
- 전체적으로 선택사항을 지정하여 이용한다.(메인 메뉴에서 보거나 선택사항을 바꾸기 위해서 Options/Frequency Conversion-Dates...를 선택한다.) 전체적으로 지정하는 것은 처음부터 평균값으로 변환되도록 지정하고 또 NAs로 채우지도 않는다.

주기 변환에 대한 예제로서 주식값이 크고, 작고 또 비슷한 데이터로 구성된 일일 데이터 계열을 월별 주기의 작업파일로 구축하고 싶다고 하자. 여기서 기본적인 주기변환을 이용하면 월별 주기의 작업파일에는 일일 데이터 계열의 평균값이 포함되게 된다. 하지만 이것은 원하는 변환이 아니다. 주식값이 큰 것에 대한 계열에는 최대값을 이용하는 주기변환방법을 지정하고, 작은 값에 대한 계열에는 최소값을 이용하는 주기변환, 그리고 값이 비슷한 계열에 대해서는 마지막 값을 이용하는 주기변환으로 지정해 주면 계열의 정의에 부합하는 월별 주기의 작업파일을 정확하게 구축할 수가 있다.

6.4 데이터베이스 자동계열(Database Auto-Series)

계열을 작업파일로 어떻게 불러오는지에 대해서는 이미 앞에서 설명하였다. 그런데 계열을 먼저 복사하지 않고 데이터베이스 안에 있는 계열을 직접 이용해서 데이터베이스 작업을 할 수 있는 방법이 있다. 이 방법의 장점은 데이터베이스의 내용이 변할 때마다 매번 데이터를 불러오는 실행을 하지 않아도 된다는 것이다. 이 방법은 제5장에서 설명한 자동계열의 방법을 이용하는 것이다.

EViews에서는 간단한 계열 이름대신에 $\log(X)$ 와 같은 계열식을 많이 이용하는데 자동적으로 실행과정에서 일시적인 자동계열을 만들어서 사용한다. 이 기능을 확장시켜서 다음과 같은 식을 이용하면 데이터베이스 안

에 있는 계열을 직접 참조하여 이용할 수가 있다.

```
db_name::object_name
```

데이터베이스 이름을 생략하고 다음과 같이 꺾콜론과 개체의 이름을 써도 된다.

```
::object_name
```

EViews는 기본 데이터베이스에서 개체를 찾는다.

새로운 계열을 생성시키는 간단한 예를 보자.

```
series lgdp=log(macro_db::dgdp)
```

EViews는 데이터베이스 macro_db에서 계열 gdp를 불러와서 gdp의 로그형태로 작업파일에 새로운 계열이름 lgdp를 만든다. 이 때 다른 개체가 계열 gdp를 사용하지 않으면 이 계열은 메모리에서 지워진다. 계열 lgdp는 현재의 작업파일의 표본만 포함하게 된다.

회귀식에서 자동계열을 이용할 수도 있다. 예로서

```
ls log(db1::y) c log(db2::x)
```

이것은 데이터베이스 db1에서 계열 y를, db2에서는 x를 불러와 필요시에는 주기변환과 마지막 소숫점을 잘라버리는 작업을 실행하여 현재의 작업파일에서 사용할 수 있는 데이터를 만든다. 그리고 회귀를 실행한다. 이 때 y와 x는 다른 데서 이용되지 않으면 메모리에서 지워진다.

자동계열은 데이터베이스 등록기에서 지정해 준 규칙에 따라 데이터베이스를 자동으로 찾아주는 기능도 한다.('데이터베이스 등록기(Database Registry)'참조) 데이터베이스 등록기에 데이터베이스 목록을 지정해 놓으면 작업파일에서 찾을 수 없는 계열을 언제든지 찾을 수가 있다.

```
series lgdp=log(gdp)
```

위의 명령은 작업파일에서 gdp라는 계열을 찾게 된다. 만일 찾지 못하면, EViews는 gdp라는 계열을 찾을 때까지 데이터베이스를 차례대로 하나씩 검색하게 된다. 찾으면 그 계열을 불러오기 하여 위의 식을 수행하게 된다. 비슷하게 회귀식에서도 적용된다.

```
ls log(y) c log(x)
```

위의 식에서도 x와 y를 작업파일에서 발견하지 못해도 불러올 수 있다. 회귀 결과에는 변수가 참조되어진 데이터베이스 이름이 함께 출력된다.

일반적으로 자동계열을 데이터베이스로부터 직접 이용하면 가장 최근의 데이터를 이용한다는 장점이 있다. 만일 데이터베이스 안에 있는 계열이

수정되더라도 그 데이터를 작업파일로 불러들이는 과정을 반복하지 않아도 되고 등식이나 모형을 재추정 하기만 하면 EViews가 자동적으로 수정된 데이터로 재실행을 해준다.

일반적으로 어떤 규칙들이 자동계열을 수정하고 삭제해 준다는 것은 복잡한 것인데 만일 자동계열의 복사본이 있어서 이미 EViews에서 사용되었는데 이때 같은 식을 두 번째 이용할 때는 그 표현식을 다시 실행해 주지는 않는다.(이 경우 데이터베이스에서 다시 로드 된다.): 그냥 원래 있던 복사본을 이용하게 된다. 만일 데이터베이스에 있는 데이터가 자동계열이 마지막으로 로드 된 이후에 변경 되었다면 새로운 표현식은 그 이전의 데이터를 이용하게 된다.

이것은 데이터베이스가 그룹의 한 요소로서 저장되면 일정시간 동안 데이터베이스가 계열의 복사본을 유지한다는 것을 의미한다. 예를 들어

```
show db1::y db2::x
```

라고 입력하면 이것은 작업파일에 표현식 db1::y와 db2::x를 포함하는 제목이 없는 그룹을 생성한다. 왼쪽에 그룹을 열어놓고 열린 데이터베이스 안에 있는 데이터를 수정해도(store나 copy명령을 이용해서) 그룹 창에서는 자동적으로 데이터가 수정되지 않는다. 더욱이 회귀식에서

```
ls log(db1::y) c log(db2::x)
```

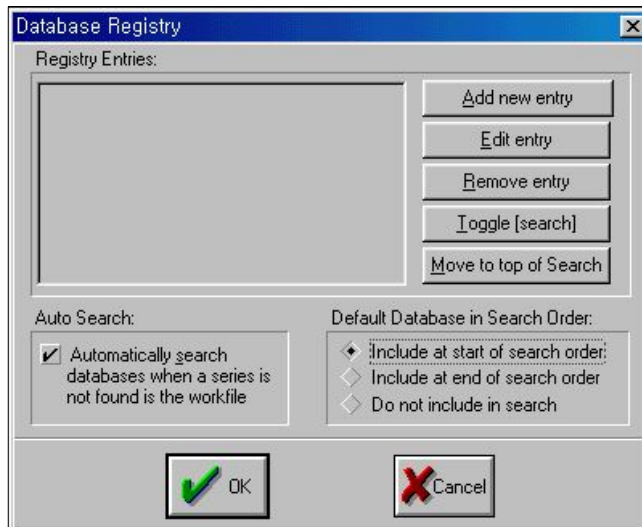
를 다시 실행시키면 이것은 제목 없는 그룹에 있는 계열의 복사본을 이용하게 된다. 데이터베이스에서 계열을 새로운 데이터로 수정하지 않는다.

6.5 데이터베이스 등록기(Database Registry)

데이터베이스 등록기란 디스크에 있는 파일로 이것은 데이터베이스에 대한 실행을 조정할 수 있는 다양한 선택사항들이 있다. 이것은 짧은 외부이름을 할당할 수 있게 해 주어 완전한 데이터베이스 경로의 위치를 이용할 수 있다. 또 EViews의 자동 찾기 성질을 형성할 수 있도록 해 주기도 한다.

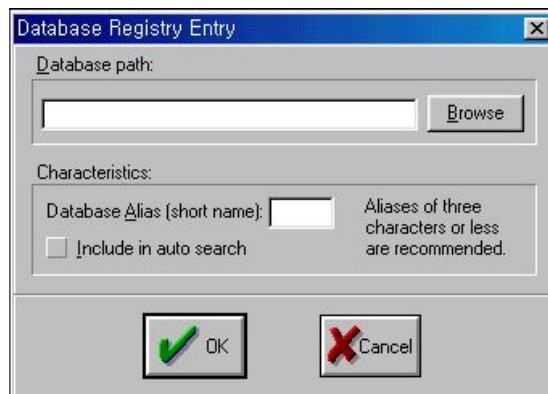
메인 메뉴에서 Options/Database Registry...를 선택하면 데이터베이스 등록기에 관한 대화상자가 나타난다.

<그림 6.9> 데이터베이스 등록기에 관한 대화상자



등록기 항목이라고 된 곳은 EViews에 등록된 데이터베이스들의 목록이다. 대화상자가 처음 나타났을 때는 상자가 비어있다. 새로운 항목 추가 (Add new entry)버튼을 누르면 두 번째 대화상자가 나타난다.

<그림 6.10> 데이터베이스 등록기 대화상자(Add)



대화상자에 세 가지를 지정해 주어야 한다. 데이터베이스 이름(경로포함), 데이터베이스 별명, 그리고 자동찾기에 데이터베이스 포함 여부에 대한 것이다.

데이터베이스의 이름과 경로는 제일 위의 편집상자에 입력하거나 Browse를 클릭하여 원하는 데이터베이스 이름을 선택해도 된다.

다음은 데이터베이스 별명이다. 이것은 짧게 입력하여 EViews 명령어에서 데이터베이스 경로를 써야하는 곳에 이용할 수가 있다. 또 데이터베이스 자동계열의 라벨에도 이 데이터베이스 별명이 쓰인다. 예를 들어, C 드라이브의 DATA라는 디렉토리에 DRIBASIC이라는 데이터베이스가 있다고 하자. 다음의 회귀 명령은 틀린 것은 아니지만 주의해야 한다.

```
ls c:\data\dribasic::gdp c c:\data\dribasic::gdp(-1)
```

여기서 보면 데이터베이스 이름이 너무 길어서 결과치의 라벨이 잘리는 경우가 발생하여 위의 회귀식에서 어떤 계열이 쓰였는지 알기 어렵다. 이때 DRIBASIC을 DRI라는 별명으로 지정하면 명령을 좀 더 간결하게 할 수가 있다.

```
ls dri::gdp c dir::gdp(-1)
```

그리고 회귀식의 결과에도 DRI라는 라벨을 쓰게 된다. 이렇게 이름이 너무 길어 잘리는 것을 방지하기 위하여 데이터베이스 자동계열을 이용할 때는 되도록 짧은 데이터베이스 별명을 이용하는 것이 좋다.

마지막으로 데이터베이스 자동찾기에 포함시키고자 할 때는 'Include in auto search'라는 체크박스를 표시하면 된다. OK를 클릭하면 새로운 항목이 목록에 포함된 메인 데이터베이스 등록기 대화상자가 나타난다.

데이터베이스 목록상자의 오른쪽에 있는 버튼을 이용하여 등록기를 수정, 삭제, 찾기 실행이나 해제, 목록에 있는 항목을 선택하여 찾기 순서의 맨 처음으로 이동하기 등의 작업을 할 수 있다.

데이터베이스 등록기 대화상자에서 그 이외의 것은 자동적인 데이터베이스 찾기에 관해서 지정하는 것이다. 자동 찾기(Auto-search) 체크박스는 현재의 작업파일에서 찾을 수 있는 계열이름을 포함하는 명령을 입력할 때 EViews가 조정할 수 있도록 하기 위해서 사용한다. 체크박스를 선택하면 예러 메시지를 나타내기 전에 찾기에 등록된 모든 데이터베이스에서 EViews가 자동적으로 찾아준다. 만일 이름이 잘못된 계열을 발견하면 EViews는 데이터베이스 자동계열을 생성하고 다시 찾는 과정을 계속한다.

대화상자의 마지막 부분은 기본데이터베이스의 찾기 순서(Default Database in Search Order)인데 이것은 자동 데이터베이스 찾기에서 기본데이터베이스를 어떻게 지정할 것인가에 관한 것이다. 자동 찾기를 수행할 때 EViews는 목록에 나타난 순서(그 항목이 체크되도록 자동 찾기

포함 상자에서 지정)대로 등록기 항목 창에 포함된 데이터베이스를 모두 찾게된다. 이러한 선택사항은 찾기를 실행할 때 기본데이터베이스에 특별한 역할을 할당하게 된다.

- 처음 찾기(Include at start of search order): 목록에 있는 데이터베이스를 찾기 전에 현재의 기본데이터베이스를 먼저 찾는다.
- 마지막 찾기(Include at end of search order): 목록에 있는 데이터베이스를 모두 찾은 후에 현재의 기본데이터베이스를 마지막으로 찾는다.
- 찾기에 포함하지 않음(Do not include in search): 현재의 기본데이터베이스가 목록에 없으면 찾지 않는다.

6.6 데이터베이스 검색기(Querying the Database)

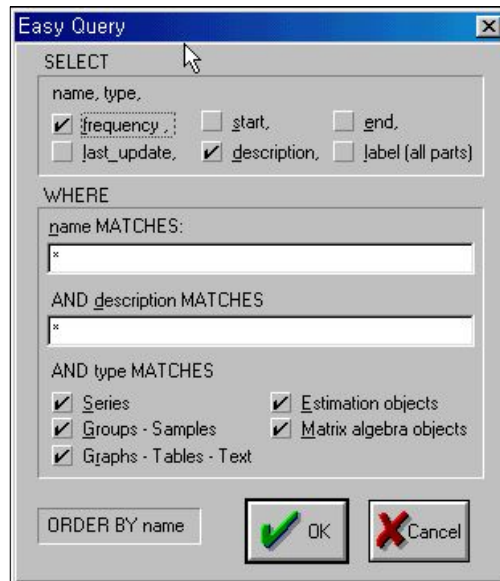
데이터베이스가 좋은 점은 폭 넓은 검색기 기능이 있다는 것이다. 이 기능은 개체를 쉽게 찾을 수 있고 비슷한 특성을 가진 개체들의 집합에 대한 실행도 쉽게 할 수 있다.

데이터베이스의 검색기 기능은 데이터베이스 창 사이에서만 이용할 수 있다. 데이터베이스에서 검색기를 이용하는 방법은 간단한 방법과 고급 방법의 두 가지가 있다. 두 가지 방법은 데이터베이스에 텍스트 검색기를 구축하는 방법이 다르다. 간단한 방법은 가장 일반적인 검색기를 실행할 수 있는 좀 더 간단한 영역을 제공한다. 고급 방법은 비용증가에 대해 좀 더 유연성을 제공하고 있다.

1) 간단한 검색기(Easy Queries)

간단한 검색기를 수행할 때는 먼저 데이터베이스를 열고 데이터베이스 창의 맨 위에 있는 툴바에서 간단한 검색(EasyQuery)을 클릭한다. 대화상자가 나타나는데 여기에는 두 개의 문자 필드와 몇 개의 체크 상자들이 있다.

<그림 6.11> 간단한 검색기(Easy Query)



이 대화상자에는 중요한 두 부분이 있는데 선택(Select)과 위치(Where)이다. 선택(Select)은 각 개체에 대한 검색 조건을 정하는 필드를 결정하는 곳이다. 그리고 위치(Where)는 검색하고자 하는 개체와 맞는 조건을 지정할 수 있는 곳이다. 간단한 검색기에서는 개체 이름, 개체의 설명, 그리고 개체의 형태를 지정할 수 있다.

위치(Where)부분에 있는 두 개의 편집 필드(이름과 설명)와 개체 형태에 대한 체크 상자들은 검색기에서 찾고자 하는 개체들의 조건을 지정하는 곳으로 이름, 설명, 개체 필터들은 연속적으로 작용하므로(논리적 ‘and’ 실행을 이용) 데이터베이스에 있는 개체들 중에서 선택한 모든 영역에 있는 개체들이 검색기 창의 결과에 나타난다.

이름과 설명 필드는 검색기의 조건에 맞는 개체들의 표현형식(*Pattern expression*)을 지정할 때 각각 이용된다. 가장 간단한 표현형식은 단일 형식으로 이루어진다. 형식은 문자와 숫자를 조합한 간단한 단어나 문자와 숫자 그리고 범용카드 기호인 ? 나 *로 만들어진 것들이 가능한데 여기서 ?은 한 문자와 대응되고 *는 '0'이나 하나 이상의 문자와 대응되는 것을 말한다. 예로서

pr?d*ction

은 production, prediction, predilection등의 단어와 대응된다. s로 시작되

는 단어는 s*, s로 끝나는 단어는 *s, 단어 중에 s를 포함하는 것은 *s*등은 자주 쓰이는 형식이다. 대문자나 소문자는 상관없다.

검색 할 때는 텍스트에 있는 단어 중 최소한 한 단어를 기준으로 한 단어 한 단어씩 실행하여 전체와 대응되는 단어를 찾게 된다. 데이터베이스에 있는 개체이름은 한 단어로 되어 있기 때문에 이름과 대응되는 형식도 이런 단어와 대응되는 것들로 이루어져 있다.

설명부분에서는 단어가 다음과 같이 구성된다. 각 단어는 숫자나 문자, 밑줄(_), 달러표시, 따옴표 등이 연속적으로 구성된다. 하지만 다음의 단어들은 무시된다. : a, an, and, any, are, as, be, between, by, for, from, if, in, is, it, not, must, of, on, or, should, that, the, then, this, to, with, when, where, while.(이것은 대상이 아닌 단어를 잘못 대응시키는 것을 최소화하고자 할 때 쓰인다.) 세 가지 단어 즉, and, or, not은 논리적 표현에서 이용한다. 다음을 보자.

예를 들어

bal. of p'ment: seas.adj. by x11

이것은 bal, p'ment, seas, adj, x11로 나뉘어 진다. 여기서 'of'와 'by'는 무시된다.

표현형식은 하나 이상의 형식을 논리 연산자인 and나 or, not으로 연결한 형식으로도 구성할 수 있는데 이것은 EViews의 논리식 표현에서 사용했던 것과 비슷하다. 'and'는 and 양쪽의 조건을 모두 만족하는 것이어야 하고 'or'는 or 양쪽의 조건 중 하나만 만족해도 되는 것이다. 또 'not'은 연산자의 오른쪽 조건이 맞지 않는 것이어야 한다. 예를 들어

s* and not *s

이것은 s로 시작하는 단어 중에 s로 끝나는 단어가 아닌 모든 개체를 대응시킨다.

하나 이상의 연산자가 표현식에서 쓰일 수 있는데 이런 경우 괄호를 사용하면 연산이 수행되는 우선 순위를 결정할 수 있다. 괄호 안에 있는 연산은 언제나 논리적으로 괄호밖에 있는 연산보다 우선 순위를 갖는다. 괄호를 중복해서 쓸 수도 있다. 만일 표현식에 괄호가 없으면 연산의 순서는 not, and, or 순으로 실행된다. 예로서

p* or s* and not *s

위의 것은 p로 시작하는 모든 개체와 b로 시작하면서 s로 끝나는 단어가 아닌 모든 개체와 대응된다.

간단한 검색기의 대화상자에서 세 번째 필터는 개체를 형식에 따라 검색해 주는 것이다. 찾고자 하는 개체 형식을 대화상자의 아래에 있는 체크 박스들을 이용해서 선택해 주면 된다.

2) 고급 검색기(Advanced Queries)

고급 검색기는 개체의 추출과 검색기에 표시되는 결과들의 조건을 다양하게 조정할 수 있다. 이러한 기능 때문에 고급 검색기는 EViews 데이터베이스를 효과적으로 이용할 수 있는 몇 가지의 이해가 필요하다.

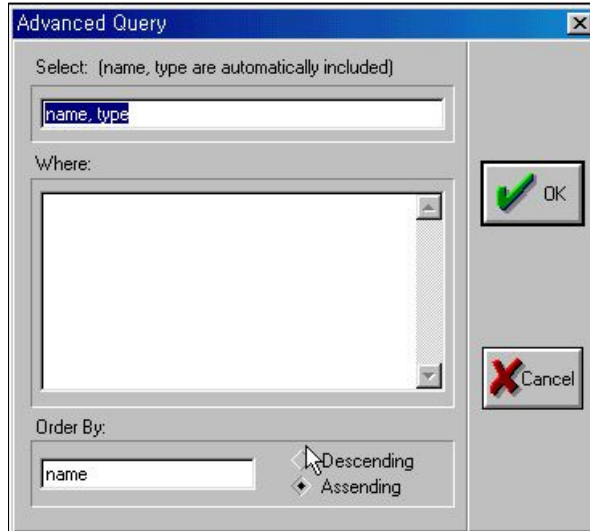
데이터베이스에 있는 각각의 개체는 몇 개의 필드에 그 내용이 설명되어 있다. 또 이런 각각의 필드는 이름으로 구분된다. 필드에는 다음과 같은 것들이 있다.

- name 개체의 이름
- type 개체의 형식
- last_write 개체를 데이터베이스에 마지막으로 쓴 시간
- last_update 개체를 마지막으로 수정한 시간
- freq 개체에 들어있는 데이터의 주기
- start 개체에 들어있는 첫 번째 데이터
- end 개체에 들어있는 마지막 데이터
- obs 계열에 저장된 데이터 개수(missing값 포함)
- description 개체에 대한 설명
- source 개체의 원본 자료
- units 개체의 단위
- remarks 개체와 관련된 추가적 설명
- history 가장 최근의 개체의 변경 사항
- display_name EViews 화면 이름

고급 검색기는 이런 필드들에 대한 내용을 검색할 수 있고 데이터베이스에서 이런 필드에 대한 조건을 대응시켜서 개체들을 검색해 준다. 고급

검색기는 데이터베이스를 열어서 창의 맨 위에 있는 툴바에서 Query를 클릭한다. 다음 그림은 고급 검색기 대화상자이다.

<그림 6.12> 고급 검색기(Advanced Query)



‘Select:’라고 되어 있는 첫 번째 편집상자는 검색 결과에 표시하고 싶은 모든 필터들의 목록을 지정할 때 이용한다. 이 상자에 입력하는 필드는 각각을 콤마로 구분한다. 이름과 형식 필드는 언제나 자동적으로 불러온다.

검색 결과의 표시 순서는 ‘Order by’ 편집상자에서 결정한다. 이 상자에는 이름 필드를 입력하는 것이 가장 유용하다. 만약 설명(description) 필드를 입력했다면 개체들의 순서를 효과적으로 나타낼 수가 없다. ‘Order by’ 필드는 같은 값의 특정한 필드와 함께 개체를 그룹화 하여 이용할 수 있다. 예를 들면 형식(type)에 따라 순서를 정하면 그 결과를 효과적으로 그룹화 하여 같은 형태의 개체들이 데이터베이스 창에 나타나도록 한다. 올림차순(Ascending)과 내림차순(Descending)은 개체들의 순서를 정렬하여 나타내고자 할 때 이용한다. 예를 들어 데이터베이스에 가장 최근에 쓰여진 개체들부터 가장 먼저 쓰여진 개체들까지를 보고자 한다면 last_write를 내림차순으로 정렬하면 된다.

위치(Where) 필드는 검색기 중 가장 복잡한 부분이다. 이 필드 안에 입력은 데이터베이스 필드에 있는 조건들을 이용하여 논리식으로 구성한다. 가장 간단한 표현식은 데이터베이스의 단일 필드에 적용하는 연산자이다.

예를 들면 월 주기나 그 이상의 주기(그 이상의 주기란 시차가 더 많이 나는 것을 말한다.)로 된 모든 계열들을 찾고자 할 때 적절한 표현식은 다음과 같다.

`freq >= monthly`

필드 표현식은 또한 논리 연산자인 `and`, `or`, `not`으로 연결할 수도 있는데 이것은 앞에서 설명한 간단한 검색기와 같다. 예를 들면, 월 주기 또는 그 이상의 주기를 갖는 모든 계열에서 1950년 이전에 시작하는 것들을 찾고자 한다면 그 표현식은 다음과 같다.

`freq >= monthly and start < 1950`

각 필드는 그 안에서 연산자나 상수를 이용할 때 각자의 규칙을 갖고 있다.

3) 이름(Name)

이름 필드에서는 알파벳 순서에 따라 이름을 비교할 수 있는 비교 연산자인 `<`, `<=`, `>`, `>=`, `=`, `<>`등을 사용할 수 있다. 예로서

`name >= c and name < m`

이것은 알파벳 `c`부터 `l`까지 이 문자로 시작하는 이름들을 검색해 준다. 이름 필드에는 또한 'matches'라는 연산자도 사용할 수 있는데 이것은 간단한 검색기에서 이름 필드를 추출해 낼 때 이용하는 것으로 앞 절에서 충분히 설명하였다. 만일 이 연산자를 하나 이상의 단어가 포함된 표현식에서 이용했다면 그 표현식에는 따옴표를 사용해야 한다. 예를 들어

`name matches "x* or y*" and freq = quarterly`

이것은 유효한 검색식이지만

`name matches x* or y* and freq = quarterly`

이것은 `matches` 연산자와 관련된 표현식이 모호하기 때문에 문법 에러가 발생한다.

4) 형식(Type)

이 필드는 '=' 연산자를 이용해서 다음의 개체 형식들과 비교하게 된다. : `sample`, `equation`, `graph`, `table`, `text`, `program`, `model`, `system`, `var`,

pool, sspace, matrix, group, sym, vector, coef, series. 순서에 특별한 논리는 없지만 관련 있는 연산자를 형식 필드에 지정한다. 그러나 'Order by' 필드에서 형식이 비슷한 개체들을 그룹화 하여 순서로 이용한다.

5) 주기(Freq)

주기 필드는 다음의 값 중 하나를 갖는다.

- u: 날짜순으로 되어 있지 않음
- a: 연간
- s: 반년간
- q: 분기
- m: 월
- w: 주
- 5: 5일
- 7: 7일

위의 문자들로 시작하는 단어는 특정한 주기를 나타내므로 월 주기는 m이나 monthly로 사용할 수 있다. 주기에 대한 순서는 계열의 시차간에 더 많은 데이터를 가지는 주기를 '이상(greater)'으로 간주한다. 연산자 <, >, <=, >=, =, <>은 모두 이런 규칙에 의해 정의되는 것이다. 예를 들어

```
freq <= quarterly
```

이것은 주기가 분기, 반년, 연간 또는 날짜 순으로 되지 않은 개체들을 검색하게 된다.

① 시작과 끝

시작과 끝 날짜는 다음의 설명과 같이 이용한다. 연간 계열의 연도 표시는 1980과 같이 쓴다. 반년 계열 표시는 연도 뒤에 6개월 단위로 S를 붙이고 그 뒤에 해당 숫자를 붙여준다. 예를 들면 1980S2와 같이 사용한다. 이 같은 형식으로 분기와 월 데이터도 연도 뒤에 각각 Q와 M을 붙이고 각 기간의 해당 숫자를 쓴다. 주, 5일과 7일 주기의 데이터는

```
mm/dd/yy
```

형식에 따라 쓴다. 여기서 m은 월, d는 일, y는 연도를 나타낸다.

날짜에 작용하는 연산자는 월력상의 순서에 따른다. 월이나 분기 데이터와 같이 한 기간에 날짜 수가 포함된 것의 데이터 순서는 그 기간의 첫 번째일부터 시작한다. 예를 들어

```
start <= 1950
```

은 1950년 1월의 첫날부터 시작하게 된다. 하지만 1950년의 두 번째, 세 번째, 또는 네 번째 분기와 같이 1950년의 다른 날과 관련된 데이터는 포함하지 않는다. 그러나 표현식

```
start < 1951
```

은 1950년의 중간 분기를 포함하게 된다.

② 마지막 쓰기와 마지막 수정(last_write, last_update)

마지막 쓰기는 디스크에 개체를 마지막으로 쓴 시간을 말하는 것이고 마지막 수정은 EViews 안에서 마지막으로 개체를 수정한 시간을 말한다. 예를 들어, 만일 새로운 계열을 작업파일에서 만들고 잠시 후에 데이터베이스에 저장했다면 마지막 쓰기는 저장 명령이 실행된 시간이 되고 마지막 수정은 새로운 계열을 만든 시간이 된다. 이 필드는 모두 날짜와 시간 정보를 다음과 같은 형식으로 나타낸다.

```
mm/dd/yyyy hh:mm
```

여기서 m은 월, d는 일, y는 연도, h는 시, 마지막 m은 분을 나타낸다.

연산자 <, >, <=, >=, =, <>은 시간 필드에서도 정의가 되므로 앞의 날짜와 시간이 뒤의 날짜나 시간보다 적은 것으로 인식된다. 다음은 전형적인 비교 형식이다.

```
last_write >= mm/dd/yy
```

일일 상수는 항상 하루의 시작은 오전 12:00를 참조한다. 하루 중 특정한 시간을 지정하는 방법은 없다.

6.7 데이터베이스 관리

많은 경우에 데이터베이스는 관리 없이도 적절하게 기능을 수행한다. 그러나 EViews가 이용자의 일반 업무를 돕기 위해서 관리가 필요한 곳에는 여러 가지 절차들이 있다.

1) 데이터베이스 파일 운용

데이터베이스에는 많은 파일들이 있기 때문에 이름은 같지만 확장자가 다른 모든 파일들은 복사, 새 이름 지정, 삭제 등과 같은 간단한 파일 운용을 할 때도 EViews의 외부에서 수행할 때는 아주 많은 절차가 필요하다. 데이터베이스 창의 툴바에 있는 Proc에는 데이터베이스 복사, 새 이름 지정, 삭제를 할 수 있는 절차가 있어서 데이터베이스에서 선택한 모든 파일에 대하여 수행할 수 있다.

파일에 대한 운용을 했을 때 자동적으로 데이터베이스 등록기를 수정해 주지는 않는다. 만일 등록된 데이터베이스를 삭제하거나 새 이름으로 지정하였다면 이름과 위치가 같은 새로운 데이터베이스를 만들거나 등록기를 편집해 주어야 한다.

2) 데이터베이스 압축

데이터베이스에서 개체들을 삽입하지 않고 많은 개체들을 삭제하면 데이터베이스에는 사용하지 않은 많은 공간이 남게 된다. 또 데이터베이스에 개체를 자주 덮어쓰는 것도 데이터베이스의 크기가 점점 커지는 경향이 있다. 크기가 커지는 것은 환경에 따라 다르지만 전형적인 데이터베이스는 그것을 하나의 경로에만 쓸 때는 크기가 약 60% 이상이 안정적이다.

데이터베이스는 압축 기능을 이용해서 그것의 크기를 최소화하도록 축소시킬 수가 있다. 데이터베이스 창 맨 위에 있는 툴바에서 Proc을 클릭한 다음 메뉴에서 'Pack the Database'를 선택한다. 데이터베이스 크기나, 이용자가 이용하는 컴퓨터 속도에 따라 이 기능은 상당히 많은 시간이 걸릴 수도 있다.

이 작업이 진행되는 동안 데이터베이스 창의 오른쪽 상단에 표시되는 압축할 수 있는 공간비율을 보면 얼마나 많은 공간이 있는지를 알 수 있다. 예를 들어 30%라는 숫자가 표시된다면 데이터베이스의 약 3분의 1이 사용하지 않은 공간이라는 것을 의미한다. 더 정확한 숫자는 데이터베이스 보기의 데이터베이스 통계량(Database Statistics)을 보면 알 수 있다. 사용하지 않은 공간(used space) 다음에 나오는 숫자는 메인 데이터베이스 파일에 있는 사용하지 않은 공간의 바이트 수를 의미한다.

3) 오류 처리

데이터베이스에 손상이 가는 것을 방지하는 가장 좋은 방법은 규칙적으로 데이터베이스를 백업해 놓는 일이다. 이것은 앞 절에서 설명한 데이터베이스 복사를 이용하면 쉽게 할 수 있다. 이 외에도 EViews에서는 손상된 데이터베이스를 다루는 여러 가지 성질들에 대해서 제공하고 있다.

데이터베이스는 손상된 정도에 따라 기본적으로 두 개의 범주로 나눌 수가 있다. 하나는 데이터베이스 창에서 열리긴 하지만 어떤 작업을 수행할 때 에러가 발생하는 것은 심하게 손상된 것은 아니어서 복구할 수 있다. 두 번째는 데이터베이스 창에서도 더 이상 열리지 않는 것으로 이것은 심하게 손상된 것이므로 새로운 데이터베이스로 다시 만들어야 한다.

위의 첫 번째 경우처럼 EViews의 데이터베이스 창에서 열리는 데이터베이스를 복구하는 방법에는 두 가지가 있다. 데이터베이스 무결 검사(Test Database Integrity)와 데이터베이스 보수(Repair Database)이다. 두 가지 모두 데이터베이스 창에 있는 툴바에서 Proc을 클릭하여 적당한 메뉴를 선택하면 된다.

데이터베이스 무결 검사(Test Database Integrity)는 메인 데이터베이스와 인덱스 파일에서 계열의 유효성을 체크한다. 만일 에러가 발견되면 메시지 상자에 발견된 에러의 형태에 따른 몇 가지 정보가 나타나고 그것을 어떻게 처리해야 하는지에 대한 방법도 제시된다. 이 검사는 데이터베이스 파일에 있는 파일의 양이 많기 때문에 모든 파일에 대하여 일치성을 체크하려면 완료될 때까지 상당히 많은 시간이 걸린다. EViews 창의 맨 아래에 있는 상태 표시줄에 나타나는 메시지를 보면 이 과정을 볼 수 있다. 데이터베이스 검사는 데이터베이스를 수정해 주는 것은 아니고 또 데이터베이스에 추가적인 손상도 입히지 않는다.

데이터베이스 보수(Repair Database)는 데이터베이스를 자동적으로 검사하여 간단한 문제는 치료를 해 준다. 이 명령은 발견한 문제를 가능한 안전하게 치료해 주긴 하지만 그래도 이 절차를 실행하기 전에는 손상된 데이터베이스를 백업해 놓는 것이 좋다.

만일 데이터베이스가 아주 심하게 손상되었을 때는 보수가 불가능하다. 이런 경우에는 'dbrebuild' 명령으로 손상된 데이터베이스를 새로운 것으로 다시 구축해야 한다. 이것은 명령어 줄에서만 수행이 가능하다.(왜냐하

면 데이터베이스를 열 수 없기 때문이다.) 명령어 사용은 다음과 같다.

```
dbrebuild old_dbname new_dbname
```

dbrebuild 명령어는 old_dbname 데이터베이스에서 복구 가능한 개체들을 찾아서 중요한 파일을 점검한다. 발견한 개체들은 새로운 데이터베이스 new_dbname으로 복사된다. 이것은 시간 소비적인 일이긴 하지만 매우 심하게 손상된 파일에서 데이터를 복구해 준다.

6.8 명령어

- sample이라는 새로운 데이터베이스를 생성할 경우
⇒ dbcreate d:\evIEWS\sample
- sample이라는 데이터베이스를 열고 그것을 기본 데이터베이스로 만들 때
⇒ dbopen sample
- sample1이라는 데이터베이스 파일이 존재하는데 열려 있지 않았으면 열고 또는 데이터베이스 파일이 존재하지 않으면 새로운 데이터베이스 sample1을 생성할 때
⇒ db sample1
또한 이 명령은 sample1을 기본 데이터베이스로도 만든다.
- 데이터베이스 sample 전체를 백업할 때
⇒ dbcopy sample sample_bak
- 데이터베이스 sample을 완전히 지울 때
⇒ dbdelete sample
- 개체(GDP와 M1)를 기본 데이터베이스에 저장할 때
⇒ store gdp m1
- 개체 GDP를 두 개의 다른 데이터베이스(sample과 sample1)로 저장할 때
⇒ store sample::gdp sample1:: gdp

- 기본 데이터베이스에 GDP와 데이터베이스 sample1에 있는 X로 시작하는 모든 개체를 저장할 때
 ⇒ `store(1) gdp sample1::x*`
 괄호 안에 있는 1은 공간을 절약하기 위해서 모든 계열을 기본인 정도2(double precision)에 저장하지 않고 정도1(single precision)에 저장하라는 의미이다.
- 두 개의 개체(GDP와 M1)를 각각의 .DB파일에 저장할 때
 ⇒ `store(i) gdp d:\evIEWS\m1`
- 두 개의 개체 GDP와 M1을 찾는 순서대로 모든 데이터베이스를 찾아서 불러올 때
 ⇒ `fetch(d) gdp m1`
- 두 개의 개체를 각각 다른 .DB 파일에서 필요시에는 합을 이용한 주기변환을 하여 불러오기 할 때
 ⇒ `fetch(i, c=s) gdp m1`
- 두 개의 데이터베이스에서 필요시에는 평균을 이용한 주기변환을 하여 이름에 GDP가 들어있는 모든 개체를 불러올 때
 ⇒ `fetch(c=a) sample1::*gdp* dir::*gdp*`
- 데이터베이스 sample1에 있는 GDP를 백업할 때
 ⇒ `copy sample1::gdp sample1::gdp_bak`
- 데이터베이스 sample에 있는 모든 개체를 다른 데이터베이스 sample1으로 복사할 때
 ⇒ `copy sample::* sample1::`

지역 경기종합지수 작성과정

목 차

1. 지역별 경제규모 및 산업구조	213
2. 개별 경제지표의 분석 및 후보지표 선정	215
2.1 개별 경제지표의 수집 및 자료 입력	215
2.2 개별 경제지표의 시계열 분석	219
1) 개별 경제지표의 경기대응력 분석 프로그램	220
2) 개별 경제지표에 의한 경기종합지수 작성 프로그램	232
2.3 후보지표 선정	238
3. 동행종합지수 시산 및 최종 구성지표 선정	243
3.1 동행종합지수 시산 프로그램	244
4. 동행종합지수의 산출	251
4.1 지역 경기종합지수 작성기법	254
4.2 지역 동행종합지수 작성과정 실제 예	258
4.3 동행종합지수 산출 프로그램	259

1. 지역별 경제규모 및 산업구조

지역별 경제규모 및 산업구조를 파악하기 위해 통계청에서 발표하는 지역내총생산(GRDP: Gross Regional Domestic Product) 및 광공업통계조사 자료를 기초로 하여 살펴보았다.

<지역별 지역내총생산 구성비(경상가격 기준)>

(단위: %)

지역	1985	1986	1987	1988	1989	1990	1991	1992	1993	1994	1995	1996	1997
서울	24.5	24.7	24.9	24.7	24.8	25.3	25.0	24.7	25.3	24.6	23.6	23.4	22.7
부산	8.1	8.0	8.1	7.9	7.7	7.8	7.5	7.0	7.0	6.9	6.7	6.7	6.4
대구	3.8	3.8	4.0	4.0	3.9	4.0	3.9	3.8	3.8	3.8	3.9	3.8	3.7
인천	4.4	4.4	4.5	4.6	4.7	5.0	5.2	5.0	5.0	4.9	5.0	4.9	4.9
광주	-	-	2.0	2.3	2.2	2.4	2.3	2.3	2.3	2.4	2.4	2.3	2.3
대전	-	-	-	-	2.3	2.3	2.3	2.4	2.4	2.3	2.2	2.2	2.3
경기	13.1	14.0	14.5	15.1	15.6	15.9	16.2	16.4	16.6	16.6	17.0	17.2	18.2
강원	3.8	3.7	3.7	3.3	3.4	3.0	2.9	2.8	2.8	2.7	2.7	2.9	2.9
충북	3.4	3.1	2.9	2.9	3.0	3.0	3.0	3.1	3.2	3.3	3.3	3.4	3.5
충남	6.4	6.4	6.0	6.1	3.7	3.6	3.6	3.8	3.9	4.1	4.1	4.5	4.5
전북	4.1	4.1	3.9	3.8	3.7	3.5	3.5	3.6	3.6	3.6	3.6	3.7	3.6
전남	8.2	7.9	5.6	5.7	5.4	5.2	5.2	5.3	5.3	5.2	5.3	5.3	5.4
경북	7.5	7.4	7.5	7.4	7.3	7.0	6.9	6.8	6.6	6.8	6.7	6.6	6.7
경남	11.6	11.5	11.4	11.2	11.1	11.1	11.6	12.0	11.3	11.9	12.2	12.0	12.0
제주	1.0	0.9	1.0	1.0	1.1	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0

자료: 통계청, 『지역내총생산』, 각년호

<산업별 지역내총생산 비중(예: 대전)>

(단위: %)

	1989	1990	1991	1992	1993	1994	1995	1996	1997	1997 (전국평균)
농림어업	2.1	1.8	1.5	1.2	1.0	1.0	0.8	1.0	1.0	5.6
제조업	32.8	34.7	32.5	27.1	25.9	25.6	25.4	25.1	24.6	32.5
건설업	11.7	15.8	19.0	22.0	20.9	16.2	15.5	14.4	17.1	13.2
기타	53.4	47.7	47.0	49.7	52.2	57.2	58.3	59.5	57.3	48.7

자료: 통계청, 『지역내총생산』, 각년호

<제조업의 산업 중분류별 부가가치액 비중(예: 대전)>

(단위: %)

산업분류	1991	1992	1993	1994	1995	1996	1997	1997 (전국)
제조업	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0
음식료품	4.8	5.8	7.0	7.4	6.7	9.3	8.2	7.0
담배	29.3	21.1	20.5	**	**	**	**	1.7
섬유제품	7.7	6.9	5.6	5.3	6.1	4.6	5.0	4.9
의복·모피제품	2.2	3.0	2.6	2.3	2.0	1.7	1.5	2.8
가죽·가방·마구류·신발	2.8	3.3	3.4	4.2	3.0	2.6	1.5	1.1
목재·나무제품	0.5	0.6	0.7	0.6	0.5	0.5	0.3	0.7
펄프·종이·종이제품	3.0	3.3	3.5	3.2	4.1	4.8	5.2	2.3
출판·인쇄·기록매체복제업	1.7	1.5	1.8	1.8	1.7	1.6	1.3	2.5
코크스·석유정제업·핵연료	1.8	1.4	1.7	**	**	**	**	4.4
화합물·화학제품	13.3	13.7	13.0	12.2	11.8	8.2	9.3	9.8
고무·플라스틱	10.6	13.5	15.6	16.6	15.8	18.4	15.8	3.7
비금속 광물제품	6.1	8.1	5.7	5.2	4.6	2.8	2.8	4.4
제1차금속산업	3.4	4.0	3.8	4.2	2.9	3.5	3.8	6.2
조립금속제품	2.0	2.4	2.5	2.7	2.6	3.0	2.6	5.0
기타 기계·장비	5.0	4.7	5.4	5.8	5.7	10.5	10.6	8.2
사무·계산·회계용기계	0.2	0.4	0.3	0.1	0.1	0.3	**	2.4
기타 전기기계·전기변환장치	1.9	2.0	1.9	2.2	2.2	2.4	2.3	3.4
영상·음향·통신장비	0.8	0.7	0.8	1.0	0.6	1.0	1.0	12.7
의료·정밀·광학기계·시계	0.5	0.7	0.6	0.6	0.7	0.9	1.0	1.1
자동차·트레일러	0.8	1.0	1.5	1.8	2.0	1.8	1.5	9.8
기타 운송장비	0.4	0.4	0.6	1.0	1.9	-	0.9	4.0
가구·기타제조업	1.2	1.5	1.6	1.5	1.7	1.6	1.2	1.8
재생재료가공처리업	0.0	0.0	0.0	0.0	0.1	0.1	0.1	0.1

자료: 통계청, 『광공업통계조사보고서』, 각년호

주: '**'는 2개이하 사업체로서 보안상 자료공개를 하지 않음

2. 개별 경제지표의 분석 및 후보지표 선정

2.1 개별 경제지표의 수집 및 자료 입력

각 지역 및 국내에서 생산되는 경제지표 중에서 이론적으로 경기변동을 잘 설명하는 월별 통계자료들을 중심으로 경제부문별로 지표를 수집 및 정리해야 한다(예: 대전지역 48개지표)

※ 개별 경제지표 선정

- ① 경제부문간의 균형을 유지하고 포괄범위를 확대하는 지표
경제순환 과정에서 중요성을 갖는 지표를 생산, 소비, 고용, 투자, 금융 등 각 부문에서 골고루 선정하여 경제부문간 균형을 유지하고 아울러 가능한 한 포괄범위가 넓은 지표를 선정하여 구성지표가 특정부문에 편중되어 전체 경기상황을 오도할 가능성을 배제해야 한다.
- ② 경기대응력을 높이고 안정적인 시차를 갖는 지표
가능한 한 모든 순환기에 대하여 경기대응성을 가지면서 시차가 안정적인 지표를 선정하여, 거짓신호의 발생가능성을 최소화해야 한다.
- ③ 통계적 유의성이 높은 지표
경기종합지수의 경기대응력을 높이기 위해서는 무엇보다도 표본의 크기, 통계작성방법, 시계열기간 등을 고려하여 통계의 질과 유의성이 높은 지표를 선정토록 하며, 가능한 한 불규칙변동이 적은 지표를 선정하여 분석이 용이하도록 한다.

<지역의 경제부문별 지표수(예: 대전)>

경제부문	지 표 명	지표수
고 용	비농가취업자수 등	8
생 산	생산지수, 출하지수, 전력소비량 등	11
소 비	소비재출하지수, 백화점판매액지수 등	7
투 자	중간재출하지수, 건축허가면적 등	9
재 고	생산자제품재고지수	1
물 가	소비자물가지수	1
통 화	어음교환액, 어음부도율 등	9
무 역	수출액, 수입액	2
계		48

▣ 지역 경제지표 목록(예: 대전)

A. 고용(8개)

NO	Data name	지 표 명	단 위	최 초 작 성	공표시기	작성기관
1	A2501	15세이상 인구(대전)	1000명	89.1	익월하순	통계청
2	A2502	경제활동인구(대전)	"	"	"	"
3	A2503	취업자수(대전)	"	"	"	"
4	A2504	취업자수(대전, 비농가)	"	"	"	"
5	A2505	취업자수(대전, 제조업)	"	92.1	"	"
6	A2506	실업자수(대전)	"	89.1	"	"
7	A2507	실업자수(대전, 비농가)	"	"	"	"
8	A2508	실업률(대전)	%	"	"	"

B. 생산·출하(11개)

NO	Data name	지 표 명	단 위	최 초 작 성	공표시기	작성기관
1	B2501	산업생산지수(대전)	95=100	90.1	익월하순	통계청
2	B2503	산업생산지수(대전, 제조업)	"	"	"	"
3	B2505	산업생산지수(대전, 중화학공업)	"	"	"	"
4	B2506	산업생산지수(대전, 경공업)	"	"	"	"
5	B2507	생산자출하지수(대전)	"	"	"	"
6	B2509	생산자출하지수(대전, 제조업)	"	"	"	"
7	B2511	생산자출하지수(대전, 중화학공업)	"	"	"	"
8	B2512	생산자출하지수(대전, 경공업)	"	"	"	"
9	B2513	전력소비량(대전)	천kwh	89.3	익익월초순	한전
10	B2514	자동차신규등록대수(대전)	대	89.1	익월하순	대전시
11	B2515	자동차등록대수(대전)	대	"	"	대전시

C. 소비(7개)

NO	Data name	지 표 명	단 위	최 초 작 성	공표시기	작성기관
1	C2501	산업생산지수(대전, 소비재)	95=100	90.1	익월하순	통계청
2	C2502	산업생산지수(대전, 내구소비재)	"	"	"	"
3	C2503	산업생산지수(대전, 비내구소비재)	"	"	"	"
4	C2504	생산지출하지수(대전, 소비재)	"	"	"	"
5	C2505	생산지출하지수(대전, 내구소비재)	"	"	"	"
6	C2506	생산지출하지수(대전, 비내구소비재)	"	"	"	"
7	C2507	백화점판매액지수(대전)	"	"	"	"

D. 투자(9개)

NO	Data name	지 표 명	단 위	최 초 작 성	공표시기	작성기관
1	D2501	생산지수(대전, 자본재)	95=100	90.1	익월하순	통계청
2	D2502	생산지수(대전, 중간재)	"	"	"	"
3	D2503	생산지수(대전, 건설용중간재)	"	"	"	"
4	D2503	생산지수(대전, 건설 장비 및 자재)	"	"	"	"
5	D2504	출하지수(대전, 자본재)	"	"	"	"
6	D2505	출하지수(대전, 중간재)	"	"	"	"
7	D2506	출하지수(대전, 건설용중간재)	"	"	"	"
8	D2503	출하지수(대전, 건설장비 및 자재)	"	"	"	"
9	D2507	건축허가면적(대전)	천m ²	89.1	"	대전시

E. 재고(1개)

NO	Data name	지 표 명	단 위	최 초 작 성	공표시기	작성기관
1	E2501	생산자제품 재고지수(대전)	95=100	90.1	익월하순	통계청

F. 물가(1개)

NO	Data name	지 표 명	단 위	최 초 작 성	공표시기	작성기관
1	F2501	소비자물가지수(대전)	95=100	75.1	익월초순	통계청

G. 통화(9개)

NO	Data name	지 표 명	단 위	최 초 작 성	공표시기	작성기관
1	G2501	어음교환액(대전)	10억	89.1	익월하순	한국은행
2	G2502	어음교환장수(대전)	천장	78.1	"	"
3	G2503	어음부도율(대전)	%	89.1	"	"
4	G2504	어음부도장수(대전)	천장	78.1	"	"
5	G2505	어음부도액(대전)	10억원	89.1	"	"
6	G2506	예금은행예금(대전)	10억원	86.1	익월하순	"
7	G2507	예금은행대출금(대전)	10억원	"	"	"
8	G2508	예금은행시설자금대출금(대전)	10억원	89.1	"	"
9	G2509	예금은행운전자금대출금(대전)	10억원	"	"	"

H. 무역(2개)

NO	Data name	지 표 명	단 위	최 초 작 성	공표시기	작성기관
1	H2501	수출액(대전)	천달러	89.1	익월하순	무역협회
2	H2502	수입액(대전)	"	"	"	"

2.2 개별 경제지표의 시계열 분석

경제지표의 시계열의 분석이란 개별 경제지표가 경기대응성을 갖는지의 여부를 검증하는 작업으로, 하나의 시계열 자료는 경제성장에 따라 상승하는 추세요인, 경제활동의 확장과 수축에 기인하는 순환요인, 계절의 변화, 추석명절과 같이 1년(12개월)을 주기로 변동하는 계절요인, 그리고 홍수나 가뭄과 같은 우발적인 불규칙요인 등 4가지 요인으로 구성되어 있다. 이 중에서 계절요인 및 불규칙요인을 제거한 추세·순환계열(TC계열)의 변동추이를 검토하는 작업을 의미하는 것이다.

개별 경제지표에 의한 경기대응력 분석 및 선행·동행·후행으로 구분할 수 있는 경기시차성을 검토하기 위해서는 한 지역의 총체적인 경기변동을 판단할 수 있는 기준순환일(turning point)이 있어야 하나 본 작업에서는 전국의 기준순환일을 기준으로 2가지 방법(growth cycle, business cycle)에 의해 분석하였다.

① 추세순환계열(TC)에 의한 경기변동 방향 분석: Growth Cycle 분석

- 순환변동치
- 단순 전년동월비
- 평활화 6개월전비(Smoothed six-month change rate)
- 평활화 12개월전비(Smoothed twelve-month change rate)

② 계절조정 및 불규칙조정(3 또는 6개월 이동평균)을 실시한 개별 경제지표의 추세순환계열(TC)에 의한 경기변동 속도 분석

: Business Cycle 분석

1) 개별 경제지표의 경기대응력 분석 프로그램

```

/*=====*/
/*== 프로그램 명칭: 대전 개별지표 경기대응력 분석 프로그램 ==*/
/*== 파일명: a:\pgm\tjanal1.prg ==*/
/*== 결과 파일명: a:\pgm\tjanal1.wf1 ==*/
/*== 프로그램 작성 날짜: 1999년 9월 13일(월) ==*/
/*== 프로그램 수정 날짜: 1999년 10월 14일(목) ==*/
/*=====*/

/*****      작업파일(workfile) 생성      *****/
create m 1989:01 2000:12

/*****      시작 및 끝년도, 구간입력      *****/
%fmon="1989:01"
%lmon="1999:07"
%period="1989:01 1999:07"
%period1="1990:01 1999:07"

%periodsf="1989:01 1998:12"

!k=12

/*****      자료 import      *****/
smpl %period
' /==== 비농가취업자수(a2504) 90.1- =====/          MCD=1
  read(b3, s=sheet1) a:\data\tjdata.xls          a2504
  read(c3, s=sheet1) a:\data\tjdata.xls          a2504pf
  read(d3, s=sheet1) a:\data\tjdata.xls          a2504sf
  genr a2504sa=a2504/a2504pf/a2504sf*100000000
  genr tj1=a2504sa

smpl %period1
' /==== 생산(b2501, b2502s) 90.1- =====/          MCD=4
  read(e15, s=sheet1) a:\data\tjdata.xls          b2501
  read(f15, s=sheet1) a:\data\tjdata.xls          b2502s
  genr tj2=b2502s

' /==== 출하(b2507, b2508s) 90.1- =====/          MCD=4
  read(g15, s=sheet1) a:\data\tjdata.xls          b2507
  read(h15, s=sheet1) a:\data\tjdata.xls          b2508s
  genr tj3=b2508s

smpl %period1
' /==== 전력소비량(b2513) 90.1- =====/          MCD=3
  read(i15, s=sheet1) a:\data\tjdata.xls          b2513
  read(j15, s=sheet1) a:\data\tjdata.xls          b2513pf
  read(k15, s=sheet1) a:\data\tjdata.xls          b2513sf
  genr b2513sa=b2513/b2513pf/b2513sf*100000000
  genr tj4=b2513sa

smpl %period1
' /==== 소비재 출하지수(c2504) 90.1- =====/          MCD=6
  read(l15, s=sheet1) a:\data\tjdata.xls          c2504
  read(m15, s=sheet1) a:\data\tjdata.xls          c2504pf

```

```

read(n15, s=sheet1) a:\data\tjdata.xls          c2504sf
genr c2504sa=c2504/c2504pf/c2504sf*100000000
genr tj5=c2504sa

smpl %period1
'/===== 백화점판매액지수(c2507) 90.1- =====/      MCD=6
read(o15, s=sheet1) a:\data\tjdata.xls          c2507
read(p15, s=sheet1) a:\data\tjdata.xls          c2507pf
read(q15, s=sheet1) a:\data\tjdata.xls          c2507sf
genr c2507sa=c2507/c2507pf/c2507sf*100000000
genr tj6=c2507sa

smpl %period
'/===== 어음교환액(g2501, 경상) 89.1- =====/      MCD=3
read(b3, s=sheet2) a:\data\tjdata.xls          g2501
read(c3, s=sheet2) a:\data\tjdata.xls          g2501sf
genr g2501sa=g2501/g2501sf*10000
genr tj7=g2501sa

smpl %period
'/===== 어음교환액(g2501r, 실질) 89.1- =====/      MCD=3
read(b3, s=sheet2) a:\data\tjdata.xls          g2501
read(d3, s=sheet2) a:\data\tjdata.xls          proddf
read(e3, s=sheet2) a:\data\tjdata.xls          g2501rsf
genr g2501r=g2501/proddf*100
genr g2501rsa=g2501r/g2501rsf*10000
genr tj8=g2501rsa

smpl %period
'/===== 수출액(h2501, 경상) 89.1- =====/      MCD=6
read(f3, s=sheet2) a:\data\tjdata.xls          h2501
read(g3, s=sheet2) a:\data\tjdata.xls          h2501pf
read(h3, s=sheet2) a:\data\tjdata.xls          h2501sf
genr h2501sa=h2501/h2501pf/h2501sf*100000000
genr tj9=h2501sa

smpl %period
'/===== 수출액(h2501r, 실질) 89.1- =====/      MCD=6
read(f3, s=sheet2) a:\data\tjdata.xls          h2501
read(i3, s=sheet2) a:\data\tjdata.xls          exdf
read(j3, s=sheet2) a:\data\tjdata.xls          h2501rpf
read(k3, s=sheet2) a:\data\tjdata.xls          h2501rsf
genr h2501r=h2501/exdf*100
genr h2501rsa=h2501r/h2501rpf/h2501rsf*100000000
genr tj10=h2501rsa

smpl %period
'/===== 수입액(h2502, 경상) 89.1- =====/      MCD=6
read(l3, s=sheet2) a:\data\tjdata.xls          h2502
read(m3, s=sheet2) a:\data\tjdata.xls          h2502sf
genr h2502sa=h2502/h2502sf*10000
genr tj11=h2502sa

smpl %period
'/===== 수입액(h2502r, 실질) 89.1- =====/      MCD=6
read(l3, s=sheet2) a:\data\tjdata.xls          h2502
read(n3, s=sheet2) a:\data\tjdata.xls          imdf
read(o3, s=sheet2) a:\data\tjdata.xls          h2502rsf

```

```

genr h2502r=h2502/imdf*100
genr h2502rsa=h2502r/h2502rsf*10000
genr tj12=h2502rsa

```

```

'***** 3 또는 6개월 이동평균(3 or 6 Month Moving Average) <2.0 TC> *****/
smpl %period
for !i=1 to !k step 1
  if !i=1 or (!i=4 (!i=7 (!i=8))) then genr tj{!i}ma=@movav(tj{!i}, 3)
  else genr tj{!i}ma=@movav(tj{!i}, 6)
  endif
next

'*****보외 *****/
'***** subroutine move*****/
subroutine move(series tjma, scalar k, scalar m)
  if k=1 then
    %date="1989:01"
  else
    %date="1990:01"
  endif

'*****미싱기간 보외 *****/
  if !i=1 or (!i=4 (!i=7 (!i=8))) then
    for !j=2 to 1 step -1
      smpl %date+!j-1+m %date+!j-1+m
      genr tjma=tjma(1)*(tjma(1)/tjma(2)+tjma(2)/tjma(3))/2
    next
  else
    for !j=5 to 1 step -1
      smpl %date+!j-1+m %date+!j-1+m
      genr tjma=tjma(1)*(tjma(1)/tjma(2)+tjma(2)/tjma(3))/2
    next
  endif
endsub

for !i=1 to !k step 1
  if !i=2 or (!i=3) then
    call move(tj{!i}ma, 2, 0)
  else if !i=4 or (!i=5) then
    call move(tj{!i}ma, 2, 0)
  else if !i=6 then
    call move(tj{!i}ma, 2, 0)
  else
    call move(tj{!i}ma, 1, 0)
  endif
endif
endif
next

'***** 개별지표의 순환변동치 산출 *****/
'***** 추세치 산출 *****/
smpl %period
for !i=1 to !k step 1
  do tj{!i}ma.HPF(14400) tr{!i}
next

```

```

'***** 순환변동치 산출 *****/
smpl %period
for !i=1 to !k step 1
    genr cy{!i}=(tj{!i}ma)/(tr{!i})*100
next

'***** 개별지표의 전년동월비 산출 *****/
smpl %period
for !i=1 to !k step 1
    genr ms{!i}=(tj{!i}ma/tj{!i}ma(-12))*100-100
next

'***** 개별지표의 평활화 6개월전비 산출 *****/
smpl %period
for !i=1 to !k step 1
    genr s6ms{!i}=(tj{!i}ma(-12)+tj{!i}ma(-11)+tj{!i}ma(-10)+
tj{!i}ma(-9)+tj{!i}ma(-8)+ tj{!i}ma(-7)+tj{!i}ma(-6)+tj{!i}ma(-5)+
tj{!i}ma(-4)+tj{!i}ma(-3)+tj{!i}ma(-2)+tj{!i}ma(-1))/12
    genr s6ms{!i}=(tj{!i}ma/s6ms{!i})^(12/6.5)*100-100
next

'***** 개별지표의 평활화 12개월전비 산출 *****/
smpl %period
for !i=1 to !k step 1
    genr s12ms{!i}=(tj{!i}ma(-17)+tj{!i}ma(-16)+tj{!i}ma(-15)+
tj{!i}ma(-14)+tj{!i}ma(-13)+ tj{!i}ma(-12)+tj{!i}ma(-11)+tj{!i}ma(-10)+
tj{!i}ma(-9)+tj{!i}ma(-8)+tj{!i}ma(-7)+tj{!i}ma(-6))/12
    genr s12ms{!i}=(tj{!i}ma/s12ms{!i})*100-100
next

'***** 추세조정계열(TC계열)의 대칭변화율(Symmetrical Percent Change) *****/
'***** subroutine symmetric *****/
subroutine symm(series sym_tj, series tjma)
    series sym_tj
    genr sym_tj=(tjma-tjma(-1))/(tjma+tjma(-1))*200
endsub

subroutine symm1(series sym_tj, series tjma)
    series sym_tj
    genr sym_tj=(tjma-tjma(-1))
endsub

for !i=1 to !k step 1
    series sym_tj{!i}
next

smpl %period
call symm(sym_tj1, tj1ma)
call symm(sym_tj2, tj2ma)
call symm(sym_tj3, tj3ma)
call symm(sym_tj4, tj4ma)
call symm(sym_tj5, tj5ma)
call symm(sym_tj6, tj6ma)
call symm(sym_tj7, tj7ma)
call symm(sym_tj8, tj8ma)
call symm(sym_tj9, tj9ma)
call symm(sym_tj10, tj10ma)

```



```

    call symm(sym_tj11, tj11ma)
    call symm(sym_tj12, tj12ma)
'***** 표준화인자(Standardization Factor) *****/
  smpl %periodsf
  for !i=1 to !k step 1
    scalar sf_tj{!i}=@mean(abs(sym_tj{!i}))
  next

'***** 표준화증감률(Standardized Rates of change) *****/
'***** subroutine Standardized Rates *****/
  subroutine strate(series strate_tj, series sym_tj, scalar sf_tj)
    series strate_tj
    genr strate_tj=sym_tj/sf_tj
  endsub

  for !i=1 to !k step 1
    series strate_tj{!i}
  next

  smpl %period
  call strate(strate_tj1, sym_tj1, sf_tj1)
  call strate(strate_tj2, sym_tj2, sf_tj2)
  call strate(strate_tj3, sym_tj3, sf_tj3)
  call strate(strate_tj4, sym_tj4, sf_tj4)
  call strate(strate_tj5, sym_tj5, sf_tj5)
  call strate(strate_tj6, sym_tj6, sf_tj6)
  call strate(strate_tj7, sym_tj7, sf_tj7)
  call strate(strate_tj8, sym_tj8, sf_tj8)
  call strate(strate_tj9, sym_tj9, sf_tj9)
  call strate(strate_tj10, sym_tj10, sf_tj10)
  call strate(strate_tj11, sym_tj11, sf_tj11)
  call strate(strate_tj12, sym_tj12, sf_tj12)

'***** 추세순환변동치(3ma) *****/
freeze(grtj1ma) tj1ma.line
  grtj1ma.shade 1988:1 1989:7
  grtj1ma.shade 1992:1 1993:1
  grtj1ma.shade 1996:3 1996:3

freeze(grtj2ma) tj2ma.line
  grtj2ma.shade 1988:1 1989:7
  grtj2ma.shade 1992:1 1993:1
  grtj2ma.shade 1996:3 1996:3

freeze(grtj3ma) tj3ma.line
  grtj3ma.shade 1988:1 1989:7
  grtj3ma.shade 1992:1 1993:1
  grtj3ma.shade 1996:3 1996:3

freeze(grtj4ma) tj4ma.line
  grtj4ma.shade 1988:1 1989:7
  grtj4ma.shade 1992:1 1993:1
  grtj4ma.shade 1996:3 1996:3

freeze(grtj5ma) tj5ma.line
  grtj5ma.shade 1988:1 1989:7

```

grtj5ma.shade 1992:1 1993:1
grtj5ma.shade 1996:3 1996:3

freeze(grtj6ma) tj6ma.line
grtj6ma.shade 1988:1 1989:7
grtj6ma.shade 1992:1 1993:1
grtj6ma.shade 1996:3 1996:3

freeze(grtj7ma) tj7ma.line
grtj7ma.shade 1988:1 1989:7
grtj7ma.shade 1992:1 1993:1
grtj7ma.shade 1996:3 1996:3

freeze(grtj8ma) tj8ma.line
grtj8ma.shade 1988:1 1989:7
grtj8ma.shade 1992:1 1993:1
grtj8ma.shade 1996:3 1996:3

freeze(grtj9ma) tj9ma.line
grtj9ma.shade 1988:1 1989:7
grtj9ma.shade 1992:1 1993:1
grtj9ma.shade 1996:3 1996:3

freeze(grtj10ma) tj10ma.line
grtj10ma.shade 1988:1 1989:7
grtj10ma.shade 1992:1 1993:1
grtj10ma.shade 1996:3 1996:3

freeze(grtj11ma) tj11ma.line
grtj11ma.shade 1988:1 1989:7
grtj11ma.shade 1992:1 1993:1
grtj11ma.shade 1996:3 1996:3

freeze(grtj12ma) tj12ma.line
grtj12ma.shade 1988:1 1989:7
grtj12ma.shade 1992:1 1993:1
grtj12ma.shade 1996:3 1996:3

'////***** 그래프를 그룹화 *****////'

freeze(group1) grtj1ma grtj2ma grtj3ma grtj4ma grtj5ma grtj6ma
freeze(group2) grtj7ma grtj8ma grtj9ma grtj10ma grtj11ma grtj12ma

'/***** 순환변동치 *****/'

freeze(grcy1) cy1.line
grcy1.shade 1988:1 1989:7
grcy1.shade 1992:1 1993:1
grcy1.shade 1996:3 1996:3

freeze(grcy2) cy2.line
grcy2.shade 1988:1 1989:7
grcy2.shade 1992:1 1993:1
grcy2.shade 1996:3 1996:3

freeze(grcy3) cy3.line
grcy3.shade 1988:1 1989:7
grcy3.shade 1992:1 1993:1
grcy3.shade 1996:3 1996:3

```

freeze(grcy4) cy4.line
  grcy4.shade 1988:1 1989:7
  grcy4.shade 1992:1 1993:1
  grcy4.shade 1996:3 1996:3

freeze(grcy5) cy5.line
  grcy5.shade 1988:1 1989:7
  grcy5.shade 1992:1 1993:1
  grcy5.shade 1996:3 1996:3

freeze(grcy6) cy6.line
  grcy6.shade 1988:1 1989:7
  grcy6.shade 1992:1 1993:1
  grcy6.shade 1996:3 1996:3

freeze(grcy7) cy7.line
  grcy7.shade 1988:1 1989:7
  grcy7.shade 1992:1 1993:1
  grcy7.shade 1996:3 1996:3

freeze(grcy8) cy8.line
  grcy8.shade 1988:1 1989:7
  grcy8.shade 1992:1 1993:1
  grcy8.shade 1996:3 1996:3

freeze(grcy9) cy9.line
  grcy9.shade 1988:1 1989:7
  grcy9.shade 1992:1 1993:1
  grcy9.shade 1996:3 1996:3

freeze(grcy10) cy10.line
  grcy10.shade 1988:1 1989:7
  grcy10.shade 1992:1 1993:1
  grcy10.shade 1996:3 1996:3

freeze(grcy11) cy11.line
  grcy11.shade 1988:1 1989:7
  grcy11.shade 1992:1 1993:1
  grcy11.shade 1996:3 1996:3

freeze(grcy12) cy12.line
  grcy12.shade 1988:1 1989:7
  grcy12.shade 1992:1 1993:1
  grcy12.shade 1996:3 1996:3

'////***** 그래프를 그룹화 *****/
freeze(group3) grcy1 grcy2 grcy3 grcy4 grcy5 grcy6
freeze(group4) grcy7 grcy8 grcy9 grcy10 grcy11 grcy12

'***** 전년동월비 *****/
freeze(grms1) ms1.line
  grms1.shade 1988:1 1989:7
  grms1.shade 1992:1 1993:1
  grms1.shade 1996:3 1996:3

freeze(grms2) ms2.line

```

grms2.shade 1988:1 1989:7
grms2.shade 1992:1 1993:1
grms2.shade 1996:3 1996:3

freeze(grms3) ms3.line
grms3.shade 1988:1 1989:7
grms3.shade 1992:1 1993:1
grms3.shade 1996:3 1996:3

freeze(grms4) ms4.line
grms4.shade 1988:1 1989:7
grms4.shade 1992:1 1993:1
grms4.shade 1996:3 1996:3

freeze(grms5) ms5.line
grms5.shade 1988:1 1989:7
grms5.shade 1992:1 1993:1
grms5.shade 1996:3 1996:3

freeze(grms6) ms6.line
grms6.shade 1988:1 1989:7
grms6.shade 1992:1 1993:1
grms6.shade 1996:3 1996:3

freeze(grms7) ms7.line
grms7.shade 1988:1 1989:7
grms7.shade 1992:1 1993:1
grms7.shade 1996:3 1996:3

freeze(grms8) ms8.line
grms8.shade 1988:1 1989:7
grms8.shade 1992:1 1993:1
grms8.shade 1996:3 1996:3

freeze(grms9) ms9.line
grms9.shade 1988:1 1989:7
grms9.shade 1992:1 1993:1
grms9.shade 1996:3 1996:3

freeze(grms10) ms10.line
grms10.shade 1988:1 1989:7
grms10.shade 1992:1 1993:1
grms10.shade 1996:3 1996:3

freeze(grms11) ms11.line
grms11.shade 1988:1 1989:7
grms11.shade 1992:1 1993:1
grms11.shade 1996:3 1996:3

freeze(grms12) ms12.line
grms12.shade 1988:1 1989:7
grms12.shade 1992:1 1993:1
grms12.shade 1996:3 1996:3

'////**** 그래프를 그룹화 ****//'

freeze(group5) grms1 grms2 grms3 grms4 grms5 grms6
freeze(group6) grms7 grms8 grms9 grms10 grms11 grms12

’/***** 평월화 6개월전비 *****/

freeze(grs6ms1) s6ms1.line
grs6ms1.shade 1988:1 1989:7
grs6ms1.shade 1992:1 1993:1
grs6ms1.shade 1996:3 1996:3

freeze(grs6ms2) s6ms2.line
grs6ms2.shade 1988:1 1989:7
grs6ms2.shade 1992:1 1993:1
grs6ms2.shade 1996:3 1996:3

freeze(grs6ms3) s6ms3.line
grs6ms3.shade 1988:1 1989:7
grs6ms3.shade 1992:1 1993:1
grs6ms3.shade 1996:3 1996:3

freeze(grs6ms4) s6ms4.line
grs6ms4.shade 1988:1 1989:7
grs6ms4.shade 1992:1 1993:1
grs6ms4.shade 1996:3 1996:3

freeze(grs6ms5) s6ms5.line
grs6ms5.shade 1988:1 1989:7
grs6ms5.shade 1992:1 1993:1
grs6ms5.shade 1996:3 1996:3

freeze(grs6ms6) s6ms6.line
grs6ms6.shade 1988:1 1989:7
grs6ms6.shade 1992:1 1993:1
grs6ms6.shade 1996:3 1996:3

freeze(grs6ms7) s6ms7.line
grs6ms7.shade 1988:1 1989:7
grs6ms7.shade 1992:1 1993:1
grs6ms7.shade 1996:3 1996:3

freeze(grs6ms8) s6ms8.line
grs6ms8.shade 1988:1 1989:7
grs6ms8.shade 1992:1 1993:1
grs6ms8.shade 1996:3 1996:3

freeze(grs6ms9) s6ms9.line
grs6ms9.shade 1988:1 1989:7
grs6ms9.shade 1992:1 1993:1
grs6ms9.shade 1996:3 1996:3

freeze(grs6ms10) s6ms10.line
grs6ms10.shade 1988:1 1989:7
grs6ms10.shade 1992:1 1993:1
grs6ms10.shade 1996:3 1996:3

freeze(grs6ms11) s6ms11.line
grs6ms11.shade 1988:1 1989:7
grs6ms11.shade 1992:1 1993:1
grs6ms11.shade 1996:3 1996:3

freeze(grs6ms12) s6ms12.line

```

grs6ms12. shade 1988:1 1989:7
grs6ms12. shade 1992:1 1993:1
grs6ms12. shade 1996:3 1996:3

'////***** 그래프를 그룹화 *****/
freeze(group7) grs6ms1 grs6ms2 grs6ms3 grs6ms4 grs6ms5 grs6ms6
freeze(group8) grs6ms7 grs6ms8 grs6ms9 grs6ms10 grs6ms11 grs6ms12

' /***** 평활화 12개월전비 *****/
freeze(grs12ms1) s12ms1. line
grs12ms1. shade 1988:1 1989:7
grs12ms1. shade 1992:1 1993:1
grs12ms1. shade 1996:3 1996:3

freeze(grs12ms2) s12ms2. line
grs12ms2. shade 1988:1 1989:7
grs12ms2. shade 1992:1 1993:1
grs12ms2. shade 1996:3 1996:3

freeze(grs12ms3) s12ms3. line
grs12ms3. shade 1988:1 1989:7
grs12ms3. shade 1992:1 1993:1
grs12ms3. shade 1996:3 1996:3

freeze(grs12ms4) s12ms4. line
grs12ms4. shade 1988:1 1989:7
grs12ms4. shade 1992:1 1993:1
grs12ms4. shade 1996:3 1996:3

freeze(grs12ms5) s12ms5. line
grs12ms5. shade 1988:1 1989:7
grs12ms5. shade 1992:1 1993:1
grs12ms5. shade 1996:3 1996:3

freeze(grs12ms6) s12ms6. line
grs12ms6. shade 1988:1 1989:7
grs12ms6. shade 1992:1 1993:1
grs12ms6. shade 1996:3 1996:3

freeze(grs12ms7) s12ms7. line
grs12ms7. shade 1988:1 1989:7
grs12ms7. shade 1992:1 1993:1
grs12ms7. shade 1996:3 1996:3

freeze(grs12ms8) s12ms8. line
grs12ms8. shade 1988:1 1989:7
grs12ms8. shade 1992:1 1993:1
grs12ms8. shade 1996:3 1996:3

freeze(grs12ms9) s12ms9. line
grs12ms9. shade 1988:1 1989:7
grs12ms9. shade 1992:1 1993:1
grs12ms9. shade 1996:3 1996:3

freeze(grs12ms10) s12ms10. line
grs12ms10. shade 1988:1 1989:7
grs12ms10. shade 1992:1 1993:1

```

```

grs12ms10.shade 1996:3 1996:3

freeze(grs12ms11) s12ms11.line
grs12ms11.shade 1988:1 1989:7
grs12ms11.shade 1992:1 1993:1
grs12ms11.shade 1996:3 1996:3

freeze(grs12ms12) s12ms12.line
grs12ms12.shade 1988:1 1989:7
grs12ms12.shade 1992:1 1993:1
grs12ms12.shade 1996:3 1996:3

'////**** 그래프를 그룹화 ****////
freeze(group9) grs12ms1 grs12ms2 grs12ms3 grs12ms4 grs12ms5 grs12ms6
freeze(group10) grs12ms7 grs12ms8 grs12ms9 grs12ms10 grs12ms11 grs12ms12

'///// 표준화증감률 /////
freeze(grstrate_tj1) strate_tj1.line
grstrate_tj1.shade 1988:1 1989:7
grstrate_tj1.shade 1992:1 1993:1
grstrate_tj1.shade 1996:3 1996:3

freeze(grstrate_tj2) strate_tj2.line
grstrate_tj2.shade 1988:1 1989:7
grstrate_tj2.shade 1992:1 1993:1
grstrate_tj2.shade 1996:3 1996:3

freeze(grstrate_tj3) strate_tj3.line
grstrate_tj3.shade 1988:1 1989:7
grstrate_tj3.shade 1992:1 1993:1
grstrate_tj3.shade 1996:3 1996:3

freeze(grstrate_tj4) strate_tj4.line
grstrate_tj4.shade 1988:1 1989:7
grstrate_tj4.shade 1992:1 1993:1
grstrate_tj4.shade 1996:3 1996:3

freeze(grstrate_tj5) strate_tj5.line
grstrate_tj5.shade 1988:1 1989:7
grstrate_tj5.shade 1992:1 1993:1
grstrate_tj5.shade 1996:3 1996:3

freeze(grstrate_tj6) strate_tj6.line
grstrate_tj6.shade 1988:1 1989:7
grstrate_tj6.shade 1992:1 1993:1
grstrate_tj6.shade 1996:3 1996:3

freeze(grstrate_tj7) strate_tj7.line
grstrate_tj7.shade 1988:1 1989:7
grstrate_tj7.shade 1992:1 1993:1
grstrate_tj7.shade 1996:3 1996:3

freeze(grstrate_tj8) strate_tj8.line
grstrate_tj8.shade 1988:1 1989:7
grstrate_tj8.shade 1992:1 1993:1
grstrate_tj8.shade 1996:3 1996:3

freeze(grstrate_tj9) strate_tj9.line

```

```

grstrate_tj9.shade 1988:1 1989:7
grstrate_tj9.shade 1992:1 1993:1
grstrate_tj9.shade 1996:3 1996:3

freeze(grstrate_tj10) strate_tj10.line
grstrate_tj10.shade 1988:1 1989:7
grstrate_tj10.shade 1992:1 1993:1
grstrate_tj10.shade 1996:3 1996:3

freeze(grstrate_tj11) strate_tj11.line
grstrate_tj11.shade 1988:1 1989:7
grstrate_tj11.shade 1992:1 1993:1
grstrate_tj11.shade 1996:3 1996:3

freeze(grstrate_tj12) strate_tj12.line
grstrate_tj12.shade 1988:1 1989:7
grstrate_tj12.shade 1992:1 1993:1
grstrate_tj12.shade 1996:3 1996:3

'////**** 그래프를 그룹화 ****////
freeze(group11) grstrate_tj1 grstrate_tj2 grstrate_tj3 grstrate_tj4
grstrate_tj5 grstrate_tj6
freeze(group12) grstrate_tj7 grstrate_tj8 grstrate_tj9 grstrate_tj10
grstrate_tj11 grstrate_tj12

save a:\pgm\tjanal1.wf1

```


2) 개별 경제지표에 의한 경기종합지수 작성 프로그램

```

/*=====*/
/*== 프로그램 명칭: 대전 개별지표의 경기지수 작성 프로그램(tc) ==*/
/*== 파일명: a:\pgm\tjjanal2.prg ==*/
/*== 결과 파일명: a:\pgm\tjjanal2.wf1 ==*/
/*== 프로그램 작성 날짜: 1999년 9월 14일(화) ==*/
/*== 프로그램 수정 날짜: 1999년 10월 14일(목) ==*/
/*=====*/

'////////// 개별 경제지표 명 및 분석 //////////'
'////////// 수정할 부분 //////////'
%jj="비농가취업자수"
'////////// 수정할 부분 //////////'

'*****      작업파일(workfile) 생성      *****/
create m 1989:01 2000:12

'*****      시작 및 끝년도, 구간입력      *****/
%fmon="1989:01"
%lmon="1999:07"
%period="1989:01 1999:07"
%period1="1990:01 1999:07"

%periodbs="1995:01 1995:12"
%periodsf="1989:01 1998:12"

!k=1
'////////// 수정할 부분 //////////'
!KK=0*12
'////////// 수정할 부분 //////////'

'*****      자료 import      *****/
smpl %period
'///// 비농가취업자수(a2504) 89.1- =====/
read(b3, s=sheet1) a:\data\tjdata.xls          a2504
read(c3, s=sheet1) a:\data\tjdata.xls          a2504pf
read(d3, s=sheet1) a:\data\tjdata.xls          a2504sf
genr a2504sa=a2504/a2504pf/a2504sf*100000000
genr tj1=a2504sa
genr co1=tj1

' smpl %period1
'///// 생산(b2501, b2502s) 90.1- =====/
read(e15, s=sheet1) a:\data\tjdata.xls          b2501
read(f15, s=sheet1) a:\data\tjdata.xls          b2502s
genr tj2=b2502s
genr co1=tj2

'///// 출하(b2507, b2508s) 90.1- =====/
read(g15, s=sheet1) a:\data\tjdata.xls          b2507
read(h15, s=sheet1) a:\data\tjdata.xls          b2508s
genr tj3=b2508s
genr co1=tj3

```



```

' read(i3, s=sheet2) a:\data\tjdata.xls          exdf
' read(j3, s=sheet2) a:\data\tjdata.xls          h2501rpf
' read(k3, s=sheet2) a:\data\tjdata.xls          h2501rsf
'          genr h2501r=h2501/exdf*100
'          genr h2501rsa=h2501r/h2501rpf/h2501rsf*100000000
'          genr tj10=h2501rsa
'          genr col=tj10
'
' smpl %period
' /===== 수입액(h2502, 경상) 89.1- =====/
' read(l3, s=sheet2) a:\data\tjdata.xls          h2502
' read(m3, s=sheet2) a:\data\tjdata.xls          h2502sf
'          genr h2502sa=h2502/h2502sf*10000
'          genr tj11=h2502sa
'          genr col=tj11
'
' smpl %period
' /===== 수입액(h2502r, 실질) 89.1- =====/
' read(l3, s=sheet2) a:\data\tjdata.xls          h2502
' read(n3, s=sheet2) a:\data\tjdata.xls          imdf
' read(o3, s=sheet2) a:\data\tjdata.xls          h2502rsf
'          genr h2502r=h2502/imdf*100
'          genr h2502rsa=h2502r/h2502rsf*10000
'          genr tj12=h2502rsa
'          genr col=tj12
'
' /***** 3 또는 6개월 이동평균(3 or 6 Month Moving Average) <2.0 TC> *****/
' /***** 수정할 부분 *****/
' smpl %period
' for !i=1 to !k step 1
'   genr co{!i}ma=@movav(co{!i}, 3)
' next
'
' smpl %period
' for !i=1 to !k step 1
'   genr co{!i}ma=@movav(co{!i}, 6)
' next
' /***** 수정할 부분 *****/
' /***** 보외 *****/
' /***** subroutine move*****/
' subroutine move(series coma, scalar k, scalar m)
'   if k=1 then
'     %date="1989:01"
'   else
'     %date="1990:01"
'   endif
'
' /*****미싱기간 보외 *****/
' /***** 수정할 부분 *****/
' for !j=2 to 1 step -1
'   smpl %date+!j-1+m %date+!j-1+m
'   genr coma=coma(1)*(coma(1)/coma(2)+coma(2)/coma(3))/2
' next
' endsub
'
' for !j=5 to 1 step -1

```

```

'
    smpl %date+!j-1+m %date+!j-1+m
    genr coma=coma(1)*(coma(1)/coma(2)+coma(2)/coma(3))/2
'
    next
'
endsub
'////////// 수정할 부분 //////////'

    for !i=1 to !k step 1
        if !i=!k then
'////////// 수정할 부분 //////////'
            call move(co{!i}ma, 1, 0)
'////////// 수정할 부분 //////////'
        endif
    next

'***** 추세조정계열(TC계열)의 대칭변화율(Symmetrical Percent Change) *****/
'***** subroutine symmetric *****/
subroutine symm(series sym_co, series coma)
    series sym_co
    genr sym_co=(coma-coma(-1))/(coma+coma(-1))*200
endsub

subroutine symm1(series sym_co, series coma)
    series sym_co
    genr sym_co=(coma-coma(-1))
endsub

for !i=1 to !k step 1
    series sym_co{!i}
next

smpl %period
    call symm(sym_co1, co1ma)

'***** 표준화인자(Standardization Factor) *****/
smpl %periodsf
    for !i=1 to !k step 1
        scalar sf_co{!i}=@mean(abs(sym_co{!i}))
    next

'***** 표준화증감률(Standardized Rates of change) *****/
'***** subroutine Standardized Rates *****/
subroutine strate(series strate_co, series sym_co, scalar sf_co)
    series strate_co
    genr strate_co=sym_co/sf_co
endsub

for !i=1 to !k step 1
    series strate_co{!i}
next

smpl %period
    call strate(strate_co1, sym_co1, sf_co1)

'***** 종합증감률(Summary Rates of Changes) *****/
smpl %period
    genr sum_rate=strate_co1

'////////// 구간별 지표수 산출 *****/

```

```

smpl %period
series count=0
for !i=1 to !k step 1
    genr num{!i}=strate_co{!i}<na
    genr count=count+num{!i}
next

'***** 평균증감률(Average Rates of Changes) *****
smpl %fmon+!kk+1 %lmon
    genr mean_rate=(sum_rate/count)

'***** 원지수 누적(Raw Index cumulation) *****/
smpl %fmon+!kk %fmon+!kk
    genr rawco_idx=100

smpl %fmon+!kk+1 %lmon
    genr rawco_idx=rawco_idx(-1)*((200+mean_rate)/(200-mean_rate))

'***** 원지수(Raw Index, Base Year '95=100) *****/
smpl %periodbs
    scalar bsco_raw=@mean(rawco_idx)

smpl %period
    genr rawcoci_idx=rawco_idx/bsco_raw*100
    genr rawcoci_rate=rawcoci_idx/rawcoci_idx(-1)*100-100

'***** 대전경기종합지수 그래프 옵션 *****/'
freeze(graph1) rawcoci_idx.line
graph1.name(1) <대전(%jj)>
graph1.addtext(t) <<대전의 동행종합지수(1개지표)>>
graph1.addtext(1t) 1995=100
graph1.shade 1988:1 1989:7
graph1.shade 1992:1 1993:1
graph1.shade 1996:3 1996:3
show graph1

'***** 대전경기종합지수 보조지표 산출 *****/'
'***** 순환변동치 산출(Cyclical Component of Coincident CI by HP-Filter) *****/'
smpl %period
do rawcoci_idx.HPF(24400) hptrend_ci
genr hpcycle=rawcoci_idx/hptrend_ci*100

'***** 전년동월비 산출 *****/'
smpl %period
genr ms_rate=(rawcoci_idx/rawcoci_idx(-12))*100-100

'***** 평활화 6개월전비 산출 *****/'
smpl %period
genr s6ms_ci=(rawcoci_idx(-12)+rawcoci_idx(-11)+rawcoci_idx(-10)+
rawcoci_idx(-9)+ rawcoci_idx(-8)+rawcoci_idx(-7)+rawcoci_idx(-6)+
rawcoci_idx(-5)+rawcoci_idx(-4)+rawcoci_idx(-3)+rawcoci_idx(-2)+rawcoci_id
x(-1))/12
genr s6ms_rate=(rawcoci_idx/s6ms_ci)^(12/6.5)*100-100

'***** 평활화 12개월전비 산출 *****/'
smpl %period

```

```

    genr s12ms_ci=(rawcoci_idx(-17)+rawcoci_idx(-16)+rawcoci_idx(-15)+
rawcoci_idx(-14)+ rawcoci_idx(-13)+rawcoci_idx(-12)+rawcoci_idx(-11)+
rawcoci_idx(-10)+rawcoci_idx(-9)+rawcoci_idx(-8)+rawcoci_idx(-7)+rawcoci_idx(
-6))/12
    genr s12ms_rate=(rawcoci_idx/s12ms_ci)*100-100

'***** 대전경기종합지수 보조지표 그래프 옵션 *****/'
'***** 순환변동치 *****/'
freeze(graph2) hpcycle.line
graph2.name(1) <대전(%jj)>
graph2.addtext(t) <<대전 동행지수의 순환변동치(1개지표)>>
graph2.addtext(lt) trend=100
graph2.shade 1988:1 1989:7
graph2.shade 1992:1 1993:1
graph2.shade 1996:3 1996:3

'***** 전년동월비 *****/'
freeze(graph3) ms_rate.line
graph3.name(1) <대전(%jj)>
graph3.addtext(t) <<대전 동행지수의 전년동월비(1개지표)>>
graph3.addtext(lt) %
graph3.shade 1992:1 1993:1
graph3.shade 1996:3 1996:3

'***** 평활화 6개월전비 *****/'
freeze(graph4) s6ms_rate.line
graph4.name(1) <대전(%jj)>
graph4.addtext(t) <<대전 동행지수의 평활화 6개월전비(1개지표)>>
graph4.addtext(lt) %
graph4.shade 1992:1 1993:1
graph4.shade 1996:3 1996:3

'***** 평활화 12개월전비 *****/'
freeze(graph5) s12ms_rate.line
graph5.name(1) <대전(%jj)>
graph5.addtext(t) <<대전 동행지수의 평활화 12개월전비(1개지표)>>
graph5.addtext(lt) %
graph5.shade 1992:1 1993:1
graph5.shade 1996:3 1996:3

'/////***** 그래프를 그룹화 *****/////'
freeze(group1) graph1 graph2 graph3 graph4 graph5
show group1

save a:\pgm\tjnal2.wf1

```

2.3 후보지표 선정

※ 후보지표 선정기준

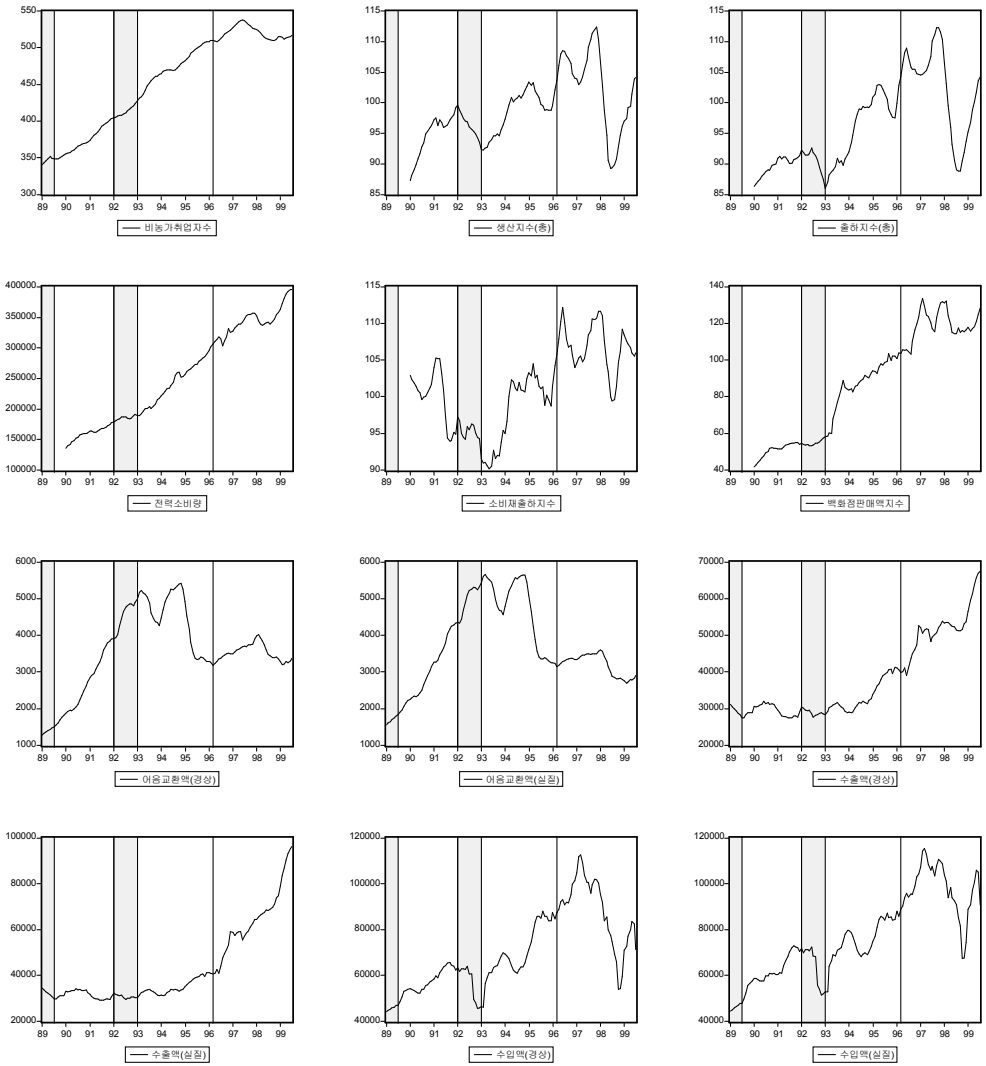
- **경제적 중요성(Economic Significance)**: 경기순환을 파악하는데 있어서 경제적 포괄범위가 넓고 경제활동의 한 부문을 대표할 수 있는 지표
- **통계적 적합성(Statistical Adequacy)**: 통계지표의 신뢰도를 평가하는 방법으로 표본의 크기, 조사대상 기간 및 주기, 통계작성방법, 시계열 장단, 시계열의 연속성 등이 평가기준이 됨
- **경기 대응성(Conformity)**: 개별지표의 순환과정 방향이 전국의 경기(기준순환일) 방향과 어느 정도로 일치하며 또한 일관성을 가지는지를 평가
- **경기 속보성(Currency)**: 통계자료가 시의성있게 정기적으로 발표되고 있는지를 평가

<대전지역 동행 후보지표 목록>

변수명	자료명	단 위	자료기간	작성기관	자료출처	MCD
TJ01 (A2504)	비농가취업자수	천명	89.1월~	통계청 사회통계과	KOSIS	1
TJ02 (B2501)	생산지수(총)	95=100	90.1월~	통계청 산업동향과	"	4
TJ03 (B2507)	출하지수(총)	95=100	90.1월~	"	"	4
TJ04 (B2513)	전력소비량 ¹⁾	10Mwh	90.1월~	한국전력	한전 영업처	3
TJ05 (C2504)	출하지수 (소비재)	95=100	90.1월~	통계청 산업동향과	KOSIS	6
TJ06 (C2507)	백화점 판매액지수	95=100	90.1월~	통계청 서비스업통계과	"	6
TJ07 (G2501)	어음교환액	10억원	89.1월~	한국은행 대전지점	"	3
TJ08 (G2501R)	어음교환액/ 생산자물가지수	10억원	89.1월~	"	"	3
TJ09 (H2501)	수 출 액	천불	89.1월~	무역협회 충남대전지부	대전시 경제정책과	6
TJ10 (H2501R)	수 출 액/ 수출물가지수(\$)	천불	89.1월~	무역협회/ 한국은행	"	6
TJ11 (H2502)	수 입 액	천불	89.1월~	무역협회 충남대전지부	"	6
TJ12 (H2502R)	수 입 액/ 수입물가지수(\$)	천불	89.1월~	무역협회/ 한국은행	"	6

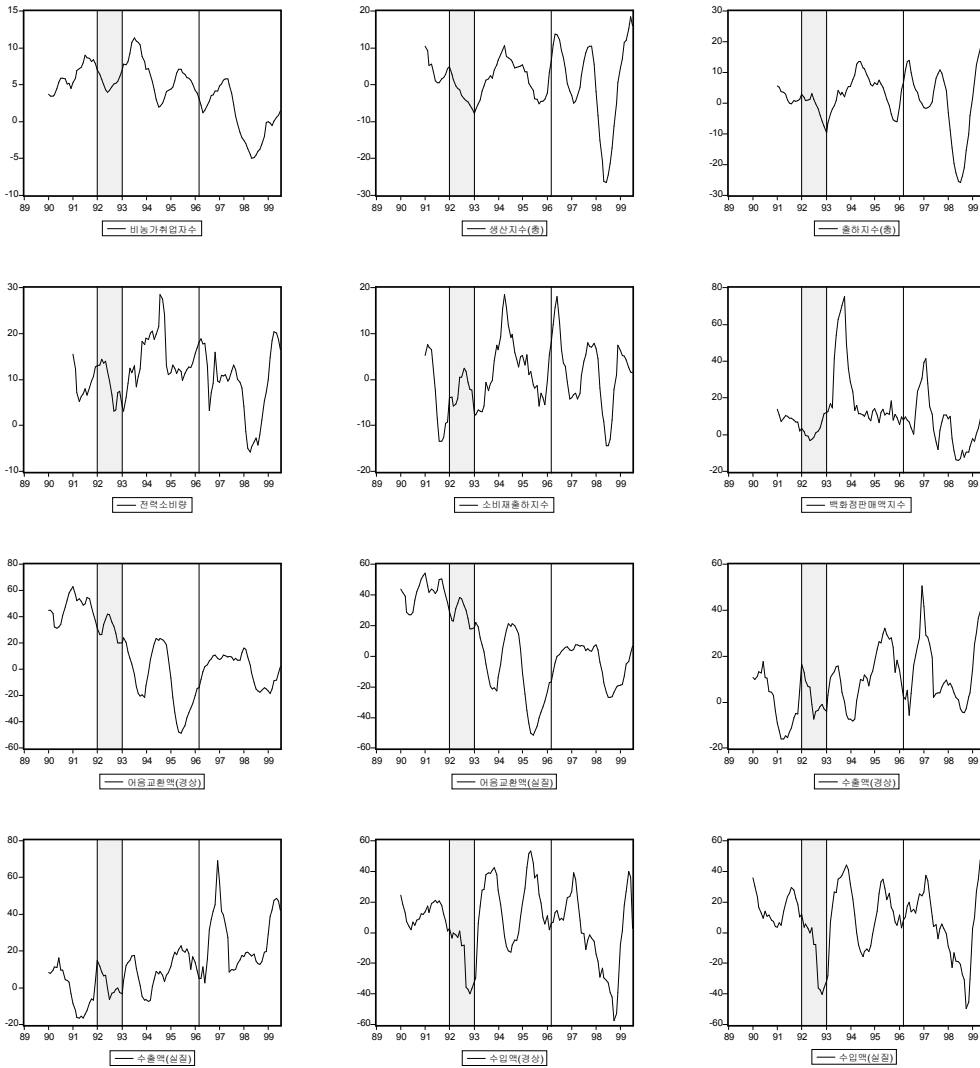
주1) 전력소비량은 행정구역별 통계를 사용하였음.

<대전지역 후보지표들의 추세순환계열>



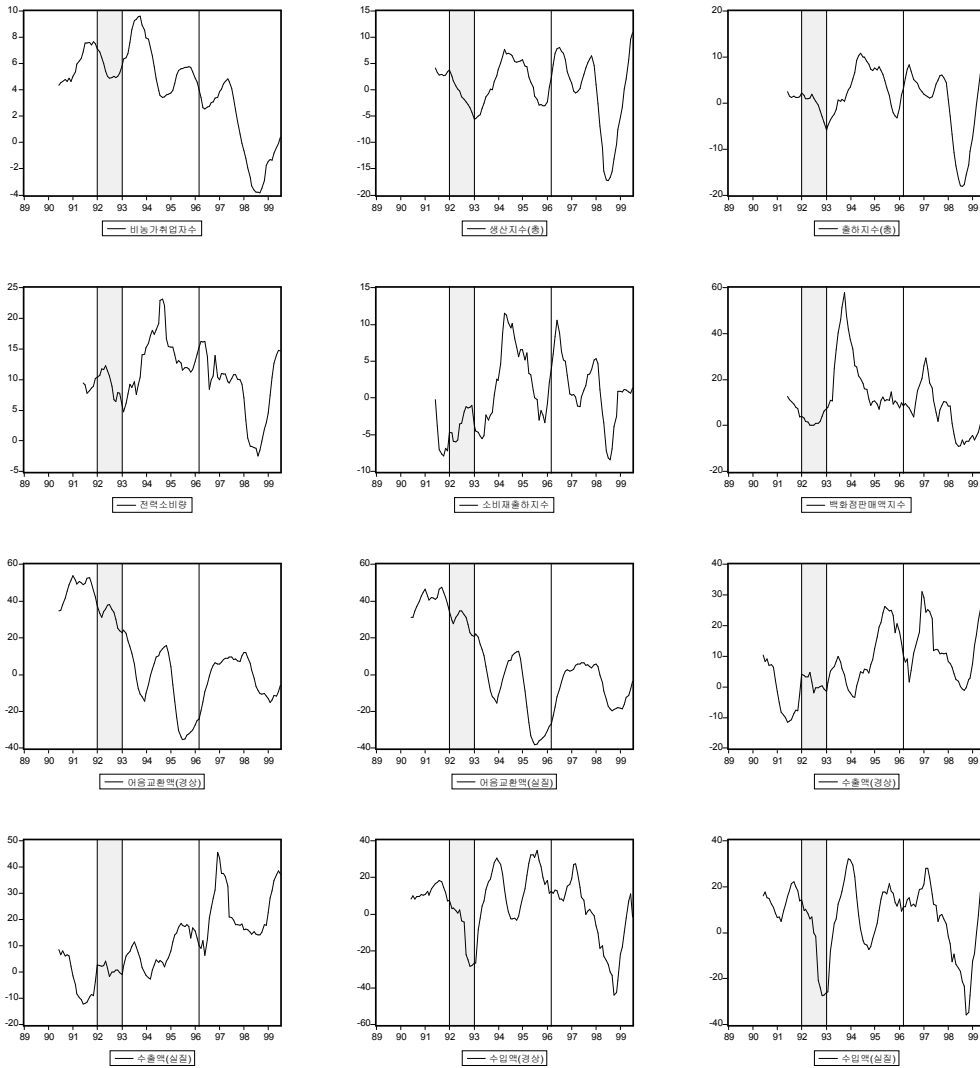
주) 음영기간은 우리나라의 경기불황기를 의미

<대전지역 후보지표들의 평활화 6개월전비>



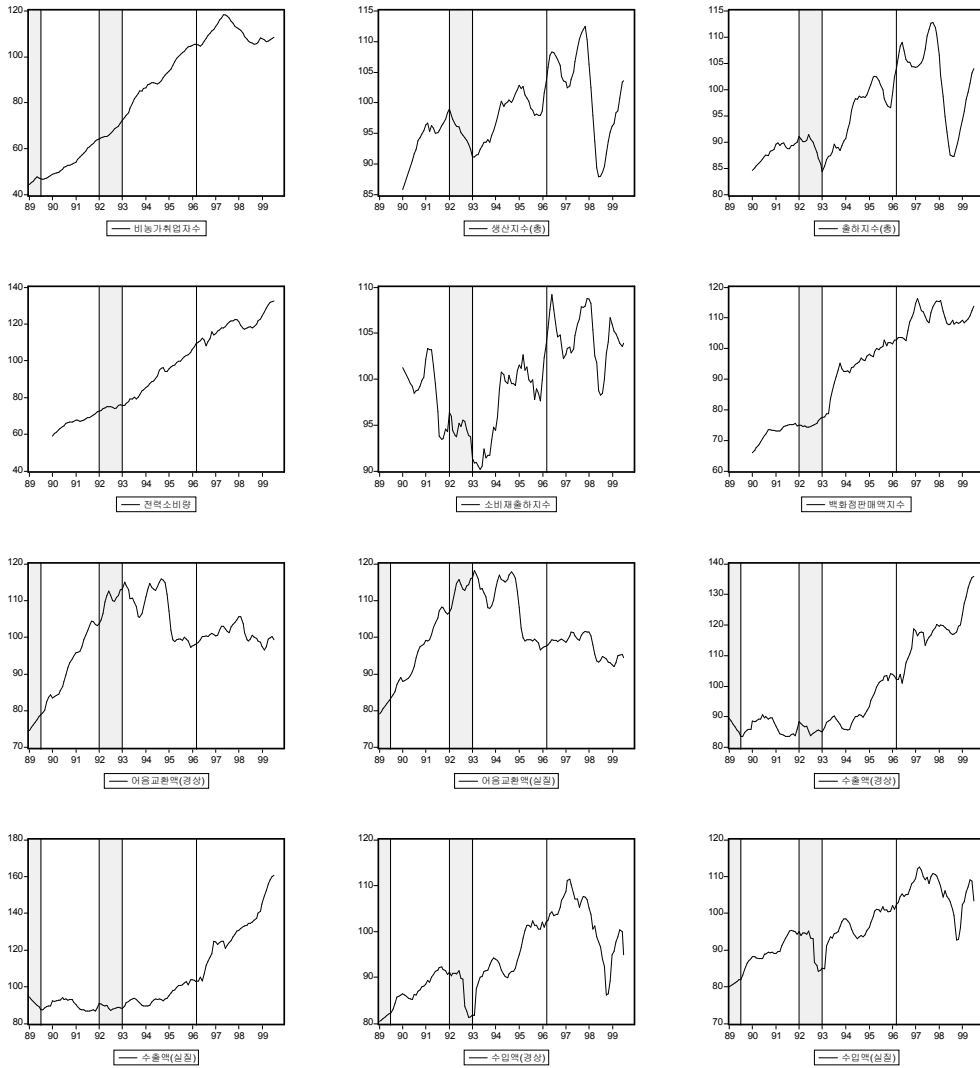
주) 음영기간은 우리나라의 경기불황기를 의미

<대전지역 후보지표들의 평활화 12개월 전비>



주) 음영기간은 우리나라의 경기불황기를 의미

<대전지역 후보지표들의 경기종합지수>

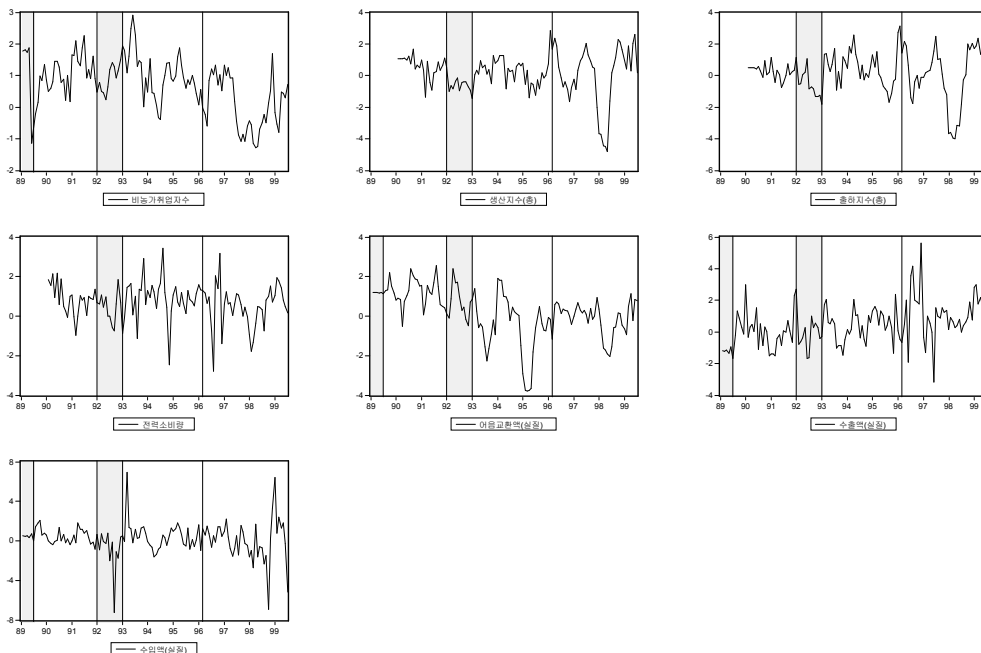


주) 음영기간은 우리나라의 경기불황기를 의미

3. 동행종합지수 시산 및 최종 구성지표 선정

- 대전지역의 경우 12개 후보지표를 이용하여 시산한 결과 다음 7개 지표를 조합하는 것이 전국의 경기변동과 최근의 경기동향 등을 감안할 때 설명력이 가장 큰 것으로 나타났음
 - 최종 구성지표(7개): 비농가취업자수, 산업생산지수, 생산자출하지수, 전력소비량, 어음교환액(실질), 수출액(실질), 수입액(실질)
- ♠ 시산작업(Simulation): 최적의 구성지표 조합(Combination)을 도출하기 위해 경제부문별로 선정된 후보지표를 이용하여 지표의 조합내용을 변경시켜가면서 CI를 작성하는 과정
- ※ 최종 구성지표 선정방법
 - 조합한 구성지표들로 CI를 작성하였을 때 전국의 기준순환일과 비교하여 경기국면이 일치하는 지를 평가

<대전지역 동행종합지수 구성지표의 표준화증감률>



주) 음영기간은 우리나라의 경기불황기를 의미

3.1 동행종합지수 시산 프로그램

```

/*=====*/
/*== 프로그램 명칭: 대전 동행지수 시산작업 프로그램(tc) ==*/
/*== 파일명: a:\pgm\simulation.prg ==*/
/*== 결과 파일명: a:\pgm\simulation.wf1 ==*/
/*== 프로그램 작성 날짜: 1999년 9월 16일(목) ==*/
/*== 프로그램 수정 날짜: 1999년 10월 15일(금) ==*/
/*=====*/

'////////// 시산(안) 프로그램, 경제지표 명 및 종류 //////////'
'////////// 수정할 부분 //////////'
%jj="tj01, tj02, tj03, tj04, tj08, tj10, tj12"
%jjj="21안"
'////////// 수정할 부분 //////////'

'////////// 자료 입력(Data Loading) 단계 //////////'

'////////// 새 workfile 생성 //////////'
create m 1989:01 2000:12

'////////// 시작 및 끝년도, 구간입력 //////////'
%fmon="1989:01"
%lmon="1999:07"
%period="1989:01 1999:07"
%period1="1990:01 1999:07"

%periodbs="1995:01 1995:12"
%periodsf="1989:01 1998:12"

%periodf="1989:01 1989:12"

'////////// 수정할 부분 //////////'
!k=7
'////////// 수정할 부분 //////////'

'////////// 자료 import //////////'
smpl %period
'///// 비농가취업자수(a2504) 89.1- =====/
read(b3, s=sheet1) a:\data\tjdata.xls a2504
read(c3, s=sheet1) a:\data\tjdata.xls a2504pf
read(d3, s=sheet1) a:\data\tjdata.xls a2504sf
genr a2504sa=a2504/a2504pf/a2504sf*100000000
genr tj1=a2504sa
genr co1=tj1

smpl %period1
'///// 생산(b2501, b2502s) 90.1- =====/
read(e15, s=sheet1) a:\data\tjdata.xls b2501
read(f15, s=sheet1) a:\data\tjdata.xls b2502s
genr tj2=b2502s
genr co2=tj2

'///// 출하(b2507, b2508s) 90.1- =====/

```

```

read(g15, s=sheet1) a:\data\tjdata.xls      b2507
read(h15, s=sheet1) a:\data\tjdata.xls      b2508s
      genr tj3=b2508s
      genr co3=tj3

smpl %period1
'/===== 전력소비량(b2513) 90.1- =====/
read(i15, s=sheet1) a:\data\tjdata.xls      b2513
read(j15, s=sheet1) a:\data\tjdata.xls      b2513pf
read(k15, s=sheet1) a:\data\tjdata.xls      b2513sf
      genr b2513sa=b2513/b2513pf/b2513sf*100000000
      genr tj4=b2513sa
      genr co4=tj4

'
smpl %period1
'/===== 소비재 출하지수(c2504) 90.1- =====/
read(l15, s=sheet1) a:\data\tjdata.xls      c2504
read(m15, s=sheet1) a:\data\tjdata.xls      c2504pf
read(n15, s=sheet1) a:\data\tjdata.xls      c2504sf
      genr c2504sa=c2504/c2504pf/c2504sf*100000000
      genr tj5=c2504sa
      genr co5=tj5

'
smpl %period1
'/===== 백화점판매액지수(c2507) 90.1- =====/
read(o15, s=sheet1) a:\data\tjdata.xls      c2507
read(p15, s=sheet1) a:\data\tjdata.xls      c2507pf
read(q15, s=sheet1) a:\data\tjdata.xls      c2507sf
      genr c2507sa=c2507/c2507pf/c2507sf*100000000
      genr tj6=c2507sa
      genr co5=tj6

'
smpl %period
'/===== 어음교환액(g2501, 경상) 89.1- =====/
read(b3, s=sheet2) a:\data\tjdata.xls      g2501
read(c3, s=sheet2) a:\data\tjdata.xls      g2501sf
      genr g2501sa=g2501/g2501sf*10000
      genr tj7=g2501sa
      genr co6=tj7

'
smpl %period
'/===== 어음교환액(g2501r, 실질) 89.1- =====/
read(b3, s=sheet2) a:\data\tjdata.xls      g2501
read(d3, s=sheet2) a:\data\tjdata.xls      proddf
read(e3, s=sheet2) a:\data\tjdata.xls      g2501rsf
      genr g2501r=g2501/proddf*100
      genr g2501rsa=g2501r/g2501rsf*10000
      genr tj8=g2501rsa
      genr co5=tj8

'
smpl %period
'/===== 수출액(h2501, 경상) 89.1- =====/
read(f3, s=sheet2) a:\data\tjdata.xls      h2501
read(g3, s=sheet2) a:\data\tjdata.xls      h2501pf
read(h3, s=sheet2) a:\data\tjdata.xls      h2501sf
      genr h2501sa=h2501/h2501pf/h2501sf*100000000
      genr tj9=h2501sa
      genr co7=tj9

```

```

,
smpl %period
' /===== 수출액(h2501r, 실질) 89.1- =====/
read(f3, s=sheet2) a:\data\tjdata.xls h2501
read(i3, s=sheet2) a:\data\tjdata.xls exdf
read(j3, s=sheet2) a:\data\tjdata.xls h2501rpf
read(k3, s=sheet2) a:\data\tjdata.xls h2501rsf
genr h2501r=h2501/exdf*100
genr h2501rsa=h2501r/h2501rpf/h2501rsf*100000000
genr tj10=h2501rsa
genr co6=tj10

,
smpl %period
' /===== 수입액(h2502, 경상) 89.1- =====/
read(l3, s=sheet2) a:\data\tjdata.xls h2502
read(m3, s=sheet2) a:\data\tjdata.xls h2502sf
genr h2502sa=h2502/h2502sf*10000
genr tj11=h2502sa
genr co8=tj11

,
smpl %period
' /===== 수입액(h2502r, 실질) 89.1- =====/
read(l3, s=sheet2) a:\data\tjdata.xls h2502
read(n3, s=sheet2) a:\data\tjdata.xls imdf
read(o3, s=sheet2) a:\data\tjdata.xls h2502rsf
genr h2502r=h2502/imdf*100
genr h2502rsa=h2502r/h2502rsf*10000
genr tj12=h2502rsa
genr co7=tj12

' /***** 3 또는 6개월 이동평균(3 or 6 Month Moving Average) <2.0 TC> *****/
' /////////////////////////////////////////////////////////////////// 수정할 부분 ////////////////////////////////////
smpl %period
for !i=1 to !k step 1
if !i=1 or (!i=4 (!i=5)) then genr co{!i}ma=@movav(co{!i}, 3)
else genr co{!i}ma=@movav(co{!i}, 6)
endif
next
' /////////////////////////////////////////////////////////////////// 수정할 부분 ////////////////////////////////////

' /***** 보외 *****/
' /***** subroutine move*****/
subroutine move(series coma, scalar k, scalar m)
if k=1 then
%date="1989:01"
else
%date="1990:01"
endif

' /*****미싱기간 보외 *****/
' /////////////////////////////////////////////////////////////////// 수정할 부분 ////////////////////////////////////
if !i=1 or (!i=4 (!i=5)) then
for !j=2 to 1 step -1
smpl %date+!j-1+m %date+!j-1+m
genr coma=coma(1)*(coma(1)/coma(2)+coma(2)/coma(3))/2
next
else

```

```

        for !j=5 to 1 step -1
            smpl %date+!j-1+m %date+!j-1+m
            genr coma=coma(1)*(coma(1)/coma(2)+coma(2)/coma(3))/2
        next
    endif
endsub

for !i=1 to !k step 1
    if !i=2 or (!i=3) then
        call move(co{!i}ma, 2, 0)
    else if !i=4 then
        call move(co{!i}ma, 2, 0)
    else
        call move(co{!i}ma, 1, 0)
    endif
endif
next

'////////// 수정할 부분 //////////'

'***** 추세조정계열(TC계열)의 대칭변화율(Symmetrical Percent Change) *****/
'***** subroutine symmetric *****/
subroutine symm(series sym_co, series coma)
    series sym_co
    genr sym_co=(coma-coma(-1))/(coma+coma(-1))*200
endsub

subroutine symm1(series sym_co, series coma)
    series sym_co
    genr sym_co=(coma-coma(-1))
endsub

for !i=1 to !k step 1
    series sym_co{!i}
next

'////////// 수정할 부분 //////////'
    smpl %period
        call symm(sym_co1, co1ma)
        call symm(sym_co2, co2ma)
        call symm(sym_co3, co3ma)
        call symm(sym_co4, co4ma)
        call symm(sym_co5, co5ma)
        call symm(sym_co6, co6ma)
        call symm(sym_co7, co7ma)
'////////// 수정할 부분 //////////'

'***** 표준화인자(Standardization Factor) *****/
    smpl %periodsf
        for !i=1 to !k step 1
            scalar sf_co{!i}=@mean(abs(sym_co{!i}))
        next

'***** 표준화증감률(Standardized Rates of change) *****/
'***** subroutine Standardized Rates *****/
subroutine strate(series strate_co, series sym_co, scalar sf_co)
    series strate_co

```



```

        genr strate_co=sym_co/sf_co
    endsub

    for !i=1 to !k step 1
        series strate_co{!i}
    next

'////////// 수정할 부분 //////////'
    smpl %period
        call strate(strate_co1, sym_co1, sf_co1)
        call strate(strate_co2, sym_co2, sf_co2)
        call strate(strate_co3, sym_co3, sf_co3)
        call strate(strate_co4, sym_co4, sf_co4)
        call strate(strate_co5, sym_co5, sf_co5)
        call strate(strate_co6, sym_co6, sf_co6)
        call strate(strate_co7, sym_co7, sf_co7)
'////////// 수정할 부분 //////////'

'***** 종합증감률(Summary Rates of Changes) *****
    series sum_rate=0
    for !i=1 to !k step 1
        genr nstrate_co{!i}=@nan(strate_co{!i}, 0)
        genr sum_rate=sum_rate+nstrate_co{!i}
    next

''/***** 구간별 지표수 산출 *****/
smpl %period
    series count=0
    for !i=1 to !k step 1
        genr num{!i}=strate_co{!i}<>na
        genr count=count+num{!i}
    next

'/***** 평균증감률(Average Rates of Changes) *****
    smpl %fmon+1 %lmon
        genr mean_rate=(sum_rate/count)

'/***** 원지수 누적(Raw Index cumulation) *****/
    smpl %fmon %fmon
        genr rawco_idx=100

    smpl %fmon+1 %lmon
        genr rawco_idx=rawco_idx(-1)*((200+mean_rate)/(200-mean_rate))

'/***** 원지수(Raw Index, Base Year '95=100) *****/
    smpl %periodbs
        scalar bsco_raw=@mean(rawco_idx)

    smpl %period
        genr rawcoci_idx=rawco_idx/bsco_raw*100
        genr rawcoci_rate=rawcoci_idx/rawcoci_idx(-1)*100-100

'/***** 대전경기종합지수 그래프 옵션 *****/
    freeze(graph1) rawcoci_idx.line
    graph1.name(1) <대전(%jj)>
    graph1.addtext(t) <<대전의 동행종합지수(%jjj)>>

```

```

graph1.addtext(1t) 1995=100
graph1.shade 1988:1 1989:7
graph1.shade 1992:1 1993:1
graph1.shade 1996:3 1996:3
show graph1

' /***** 대전경기종합지수 보조지표 산출 *****/
' /***** 순환변동치 산출(Cyclical Component of Coincident CI by HP-Filter) *****/
smpl %period
do rawcoci_idx.HPF(24400) hptrend_ci
genr hpcycle=rawcoci_idx/hptrend_ci*100

' /***** 전년동월비 산출 *****/
smpl %period
genr ms_rate=(rawcoci_idx/rawcoci_idx(-12))*100-100

' /***** 평활화 6개월전비 산출 *****/
smpl %period
genr s6ms_ci=(rawcoci_idx(-12)+rawcoci_idx(-11)+rawcoci_idx(-10)+
rawcoci_idx(-9)+ rawcoci_idx(-8)+rawcoci_idx(-7)+rawcoci_idx(-6)+
rawcoci_idx(-5)+rawcoci_idx(-4)+rawcoci_idx(-3)+rawcoci_idx(-2)+rawcoci_idx(-1))/12
genr s6ms_rate=(rawcoci_idx/s6ms_ci)^(12/6.5)*100-100

' /***** 평활화 12개월전비 산출 *****/
smpl %period
genr s12ms_ci=(rawcoci_idx(-17)+rawcoci_idx(-16)+rawcoci_idx(-15)+
rawcoci_idx(-14)+ rawcoci_idx(-13)+rawcoci_idx(-12)+rawcoci_idx(-11)+
rawcoci_idx(-10)+rawcoci_idx(-9)+rawcoci_idx(-8)+rawcoci_idx(-7)+rawcoci_idx(-6))/12
genr s12ms_rate=(rawcoci_idx/s12ms_ci)*100-100

' /***** 대전경기종합지수 보조지표 그래프 옵션 *****/
' /***** 순환변동치 *****/
freeze(graph2) hpcycle.line
graph2.name(1) <대전(%jj)>
graph2.addtext(t) <<대전 동행지수의 순환변동치(%jjj)>>
graph2.addtext(1t) trend=100
graph2.shade 1988:1 1989:7
graph2.shade 1992:1 1993:1
graph2.shade 1996:3 1996:3
show graph2

' /***** 전년동월비 *****/
freeze(graph3) ms_rate.line
graph3.name(1) <대전(%jj)>
graph3.addtext(t) <<대전 동행지수의 전년동월비(%jjj)>>
graph3.addtext(1t) %
graph3.shade 1988:1 1989:7
graph3.shade 1992:1 1993:1
graph3.shade 1996:3 1996:3
show graph3

' /***** 평활화 6개월전비 *****/
freeze(graph4) s6ms_rate.line
graph4.name(1) <대전(%jj)>
graph4.addtext(t) <<대전 동행지수의 평활화 6개월전비(%jjj)>>
graph4.addtext(1t) %

```

```

graph4.shade 1988:1 1989:7
graph4.shade 1992:1 1993:1
graph4.shade 1996:3 1996:3
show graph4

' /***** 평활화 12개월전비 *****/
freeze(graph5) s12ms_rate.line
graph5.name(1) <대전(%jj)>
graph5.addtext(t) <<대전 동행지수의 평활화 12개월전비(%jj)>>
graph5.addtext(lt) %
graph5.shade 1988:1 1989:7
graph5.shade 1992:1 1993:1
graph5.shade 1996:3 1996:3
show graph5

' /*****/
freeze(group1) graph1 graph2 graph3 graph4 graph5
show group1

save a:\pgm\simulation.wf1

```

4. 동행종합지수의 산출

- 최종적으로 선정된 7개 구성지표들을 MCD(Months for Cyclical Dominance)에 따른 불규칙성을 감안하여 3 또는 6개월 이동평균방법으로 동행종합지수를 작성
 - 3 또는 6개월 이동평균방법: $MCD \leq 3$ 이면 3개월 이동평균
 $MCD \geq 4$ 이면 6개월 이동평균

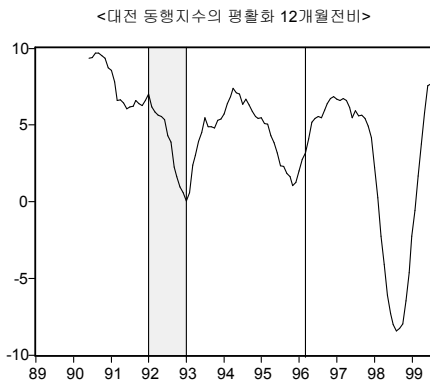
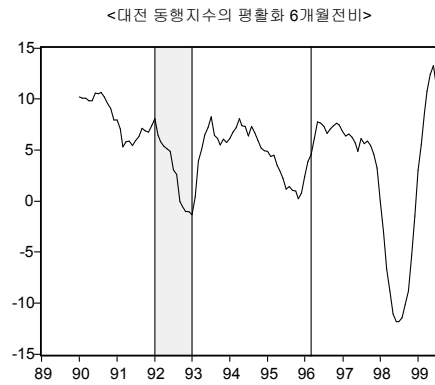
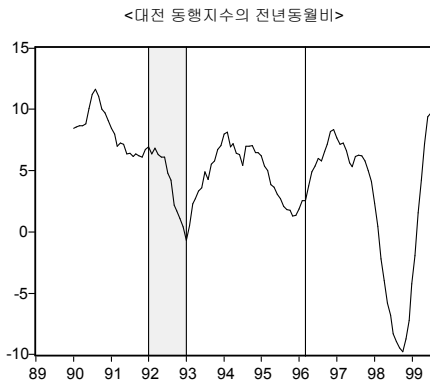
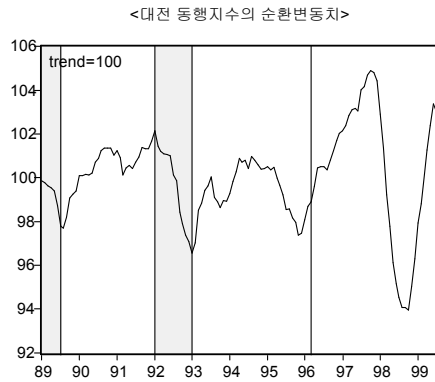
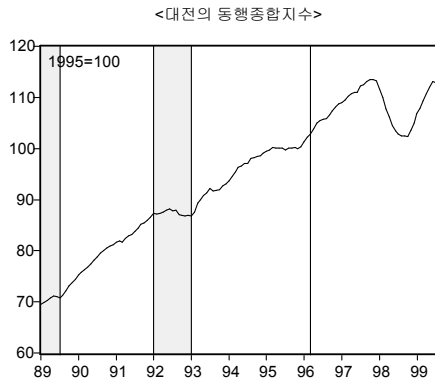
<대전지역 구성지표의 불규칙 정도(MCD)>

(단위: 개월)

지 표 명	비농가 취업자수	생산 지수	출하 지수	전력 사용량	어음 교환액	수출액	수입액
추세순환계열(TC) /불규칙계열(I)	1	4	4	3	3	6	6

- 대전지역의 경기종합지수 작성시 3개월 이동평균보다는 불규칙정도를 고려한 3 또는 6개월 이동평균에 의한 경기종합지수 작성방법이 경기국면 파악에 용이한 것으로 나타남
- 과거 고도성장의 영향으로 종합지수상에서 경기전환점(turning point)과 경기 확장 및 수축국면 파악이 어렵기 때문에 성장순환(growth cycle)과정을 판단할 수 있는 보조지표를 동시에 작성

▣ 대전의 동행종합지수 및 보조지표



주) 음영기간은 우리나라의 경기불황기를 의미

▣ 대전지역 동행종합지수의 평가

- 대전지역의 동행종합지수는 추세가 강하여 경기의 순환과정을 파악하기 어려우므로 보조지표를 사용하여야 함
 - 4개의 보조지표 중 평활화 12개월 변화율이 대전경기의 순환과정을 비교적 잘 설명하는 것으로 나타남
- 대전지역의 경기 상황은 전국의 경기국면과 거의 유사한 것으로 나타났으나, 93년 1월부터 96년 3월까지 전국 경기는 확장국면을 지속하였으나 대전지역의 경기는 기간중 일시적 수축국면(94 4월: 정점, 95년 5월: 저점)이 비교적 뚜렷한 것으로 나타났음
- 시산된 대전지역 동행종합지수에 의하면 대전지역의 최근 경기는 98년 8월에 경기저점을 시험하고 이후 경기상승세가 지속되고 있는 것으로 판단됨

<대전 동행지수의 평활화 12개월 변화율>



주) 음영기간은 우리나라의 경기불황기를 의미

4.1 지역 경기종합지수 작성기법

- 지역 동행종합지수를 작성하기 위한 기법은 통계청의 작성방법을 기본틀로 하고 있으며 통계자료의 특성상 다음과 같은 차이점이 있음
 - 통계시계열의 불규칙성으로 인한 불규칙조정방법
 - 통계청: 1 또는 3개월 이동평균
 - 지역: 3 또는 6개월 이동평균
 - 목표추세(target trend)인 지역내총생산(GRDP) 자료의 시의성문제로 인한 추세조정방법
 - 통계청: 국내총생산(GDP)으로 추세조정
 - 지역: 추세조정 하지않음
 - 지역 경기종합지수에서는 경기순환과정을 명확히 파악하기 어려워서 성장순환을 나타내도록 하는 보조지표 개발
 - 통계청: 순환변동치
 - 지역: 순환변동치, 전년동월비, 평활화 6(12)개월전비

가. 개별 구성지표의 비경기적 요인 제거

- ① 개별 구성지표들의 비경기적요인을 제거하기 위하여 계절조정(X-12-ARIMA) 및 불규칙조정(3 또는 6개월 말항 이동평균)을 하여 추세를순환계열($X_{i,t} = T \times C$)을 산출한다.

※ 3개월 말항 이동평균시 결항 보외 방법

3개월 말항 이동평균시 최초 년도의 2개월(초항 및 두번째항)은 결항이 되므로 보외하여 사용하는데 그 방법은 다음과 같다(6개월 말항 이동평균의 경우도 동일한 기법으로 하면 된다)

- ㉠ 수준지표 및 비율지표일 경우

$$\text{두번째항: } X_{i,2} = X_{i,3} \times \left(\frac{(X_{i,3}/X_{i,4}) + (X_{i,4}/X_{i,5})}{2} \right)$$

$$\text{초 항: } X_{i,1} = X_{i,2} \times \left(\frac{(X_{i,2}/X_{i,3}) + (X_{i,3}/X_{i,4})}{2} \right)$$

여기서, $i=1,2,\dots,N$ (구성지표)

㉠ 전년동월비자료의 경우는 그 산식을 $X_{i,t+12}/X_{i,t} \times 100$ 으로 계산하여 비율지표로 만든 후 ①과정으로 계산하고 다시 전시계열에서 100을 빼면 전년동월비의 3개월 말항 이동평균계열이 작성된다.

② 구성지표별로 전월대비 증감률(대칭변화율($Y_{i,t}$), symmetric percent changes)을 산출한다.

㉡ 수준지표(level indicators)일 경우

$$Y_{i,t} = \frac{X_{i,t} - X_{i,t-1}}{X_{i,t} + X_{i,t-1}} \times 200$$

※ 대칭변화율과 일반변화율의 차이

양(+)과 음(-)의 변화를 대칭적으로 처리하기 위하여 분모를 일반적으로 사용하는 전월치가 아닌 금월치와 전월치의 중간치로 사용한다.

〈일반변화율과 대칭변화율의 차이〉

	1월	2월	3월	4월	5월	평균	비 고
원계열	50	100	50	100	50	-	
일반변화율	-	100	-50	100	-50	25	분모=전월치
대칭변화율	-	66.7	-66.7	66.7	-66.7	0	분모=(금월치+전월치)/2

㉢ 비율지표(ratio indicators)와 영(0) 또는 음수가 포함된 경우는 차분을 이용

$$Y_{i,t} = X_{i,t} - X_{i,t-1}$$

여기서, $i=1,2,\dots,N$ (구성지표), $t=2,3,\dots,T$ (시점)

나. 구성지표의 표준화

- 지역 경기종합지수 구성지표들은 그 경제적 특성(변동성, Volatility)에 따라 경기변동에 반응하는 순환진폭이 서로 다르기 때문에 이것을 그대로 평균하면 진폭이 큰 구성지표에 의하여 경기종합지수의 움직임이 좌우될 수 있음

- 구성지표의 표준화란 이러한 현상을 방지하기 위하여 각 구성지표의 순환진폭이 평균적으로 모두 같아지도록 조정하는 것을 의미

③ 진폭(변동폭)이 큰 하나의 구성지표에 의해 종합지수가 좌우되지 않도록 각 구성지표의 대칭변화율($Y_{i,t}$)을 절대치평균인 표준화인자(S_i)로 나누어 표준화증감률($Z_{i,t}$)을 산출한다.

$$Z_{i,t} = \frac{Y_{i,t}}{S_i} \quad \text{단, } S_i (\text{표준화인자}) = \frac{\sum_{t=2}^T |Y_{i,t}|}{T-1}$$

④ 표준화증감률($Z_{i,t}$)의 합을 구성지표수로 나누어 평균증감률(AC_t)을 산출한다.

$$AC_t = \frac{\sum_{i=1}^N Z_{i,t}}{N}$$

다. 지수화

○ 지역 동행종합지수는 매월의 평균증감률을 누적하여 원지수를 계산한 뒤, 기준년도(1995년)로 조정하여 산출

⑤ 매월의 평균증감률(AC_t)을 누적하여 지역별 동행종합지수 원지수(RCI_t)를 산출한다.(시작년월=100)

$$RCI_t = RCI_{t-1} \times \frac{200 + AC_t}{200 - AC_t} \quad \text{단, } RCI_1 = 100$$

⑥ 동행종합지수 원지수(RCI_t)를 기준년도(1995년)로 조정하여 지역별 동행종합지수(CI_t)를 산출한다.(기준년도=100)

$$CI_t = \frac{RCI_t}{BASE} \times 100 \quad \text{단, } BASE \text{는 기준년도의 } RCI_t \text{의 평균}$$

라. 경기순환 판단을 위한 보조지표

- 지역별 동행종합지수도 과거 고도성장의 영향으로 강한 추세를 가지고 있어 경기 순환적인 움직임을 확인할 수 있는 다양한 보조지표 작성

- 보조지표: 순환변동치, 전년동월비, 평활화 6개월전비, 평활화 12개월전비 등을 사용하였음

※ 평활화 전년동월비 방법은 미국의 CIBCR 및 ECRI에서 우리나라의 성장순환(Growth Cycle)을 파악하기 위해 사용하고 있음

- ⑦ 지역별 동행종합지수(CI_t)는 상승추세가 너무 강하여 지수에서는 경기속도만 파악할 수 있으므로, 국면 및 전환점 등 경기순환과정을 파악하기 위해 다음과 같은 여러 가지 보조지표를 산출한다.

㉠ 순환변동치

$$C_t = \frac{CI_t}{T_t} \times 100$$

여기서, T_t 는 H-P Filters 방법으로 산출

㉡ 전년동월비

$$12MS_t = \left(\frac{CI_t}{CI_{t-12}} \right) \times 100 - 100$$

㉢ 평활화 6개월 전비

$$S6MS_t = \left(\frac{CI_t}{\frac{\sum_{t-12}^{-1} CI_t}{12}} \right)^{\frac{-12}{6.5}} \times 100 - 100$$

㉣ 평활화 12개월 전비

$$S12MS_t = \left(\frac{CI_t}{\frac{\sum_{t-17}^{-6} CI_t}{12}} \right) \times 100 - 100$$

4.2 지역 동행종합지수 작성과정 실제 예(대전)

	(1) 계절변동 및 불규칙변동 조정							
	A. 원계열				B. 계절요인			
	99.4월	5월	6월	7월	99.4월	5월	6월	7월
산업생산지수	109.4	111.4	117.4	96.6	-	-	-	-
전력소비량	379108	372704	384546	405950	9653	9426	9682	10268
수출액	92352	99592	98406	97380	9379	9912	10420	10267

주) 전력소비량은 명절효과를 반영하여 사전인자로 나눈 자료임.

수출액은 수입물가지수(\$기준)으로 실질화한 후 명절효과를 반영한 자료

	(1) 계절변동 및 불규칙변동 조정						
	C. 계절조정계열(A/B*10000)				D. 불규칙조정계열		
	99.4월	5월	6월	7월	99.6월	7월	
산업생산지수	102.2	109.9	113.5	97.7	103.9	104.1	
전력소비량	392736	395400	397176	395354	395104	395977	
수출액	98467	100477	94439	94848	95290	96156	

	(2) 구성지표의 표준화					
	E. 대칭변화율		F. 표준화인자 (과거증감률의 절대치 의 평균)	G. 표준화증감 률(E/F)	H. 종합증감 률(ΣG)	I. 평균증감률 (H/구성지표 수)
	6월	7월	7월	7월	7월	
산업생산지수	2.4531	0.1924	0.9384	0.2050		
전력소비량	0.6533	0.2207	1.3195	0.1672	0.8476	0.2825
수출액	2.3314	0.9041	1.9021	0.4753		

(3) 원지수 및 동행종합지수 산출	
계산과정별 결과	<ul style="list-style-type: none"> · 6월 지수=165.1이면 · 7월 지수=$165.1 \times \frac{200+0.2825}{200-0.2825}$ =165.1×1.0028=165.6 <div style="display: inline-block; vertical-align: top; width: 45%; margin-left: 20px;"> <ul style="list-style-type: none"> 기준년도(1995=126.9)를 100으로 하여 산출 · 7월 지수=165.6/126.9 =130.5 </div>

4.3 동행종합지수 산출 프로그램

```

'/*=====*/
'/*== 프로그램 명칭: 대전 동행지수 산출 경상 프로그램 ==*/
'/*== 파일명: a:\pgm\coin99.prg ==*/
'/*== 결과 파일명: a:\pgm\coin99.wf1 ==*/
'/*== 프로그램 작성 날짜: 1999년 9월 16일(목) ==*/
'/*== 프로그램 수정 날짜: 1999년 10월 15일(금) ==*/
'/*=====*/

'//***** 자료 입력(Data Loading) 단계 *****//

'/****** 새 workfile 생성 *****/
create m 1989:01 2000:12

'/****** 시작 및 끝년도, 구간입력 *****/
%fmon="1989:01"
%lmon="1999:07"
%period="1989:01 1999:07"
%period1="1990:01 1999:07"

%periodbs="1995:01 1995:12"
%periodsf="1989:01 1998:12"

%periodf="1989:01 1989:12"

!k=8

'/****** 자료 import *****/
smpl %period
'/*===== 비능가취업자수(a2504) 89.1- =====/
read(b3, s=sheet1) a:\data\tjdata.xls a2504
read(c3, s=sheet1) a:\data\tjdata.xls a2504pf
read(d3, s=sheet1) a:\data\tjdata.xls a2504sf
genr a2504sa=a2504/a2504pf/a2504sf*100000000
genr tj1=a2504sa
genr co1=tj1

smpl %period1
'/*===== 생산(b2501, b2502s) 90.1- =====/
read(e15, s=sheet1) a:\data\tjdata.xls b2501
read(f15, s=sheet1) a:\data\tjdata.xls b2502s
genr tj2=b2502s
genr co2=tj2

smpl %period1
'/*===== 출하(b2507, b2508s) 90.1- =====/
read(g15, s=sheet1) a:\data\tjdata.xls b2507
read(h15, s=sheet1) a:\data\tjdata.xls b2508s
genr tj3=b2508s
genr co3=tj3

smpl %period1
'/*===== 전력사용량(b2513) 90.1- =====/

```

```

read(i15, s=sheet1) a:\data\tjdata.xls          b2513
read(j15, s=sheet1) a:\data\tjdata.xls          b2513pf
read(k15, s=sheet1) a:\data\tjdata.xls          b2513sf
genr b2513sa=b2513/b2513pf/b2513sf*100000000
genr tj4=b2513sa
genr co4=tj4

smpl %period1
'/===== 백화점판매액지수(c2507) 90.1- =====/
read(o15, s=sheet1) a:\data\tjdata.xls          c2507
read(p15, s=sheet1) a:\data\tjdata.xls          c2507pf
read(q15, s=sheet1) a:\data\tjdata.xls          c2507sf
genr c2507sa=c2507/c2507pf/c2507sf*100000000
genr tj6=c2507sa
genr co5=tj6

smpl %period
'/===== 어음교환액(g2501, 경상) 89.1- =====/
read(b3, s=sheet2) a:\data\tjdata.xls          g2501
read(c3, s=sheet2) a:\data\tjdata.xls          g2501sf
genr g2501sa=g2501/g2501sf*10000
genr tj7=g2501sa
genr co6=tj7

smpl %period
'/===== 수출액(h2501, 경상) 89.1- =====/
read(f3, s=sheet2) a:\data\tjdata.xls          h2501
read(g3, s=sheet2) a:\data\tjdata.xls          h2501pf
read(h3, s=sheet2) a:\data\tjdata.xls          h2501sf
genr h2501sa=h2501/h2501pf/h2501sf*100000000
genr tj9=h2501sa
genr co7=tj9

smpl %period
'/===== 수입액(h2502, 경상) 89.1- =====/
read(l3, s=sheet2) a:\data\tjdata.xls          h2502
read(m3, s=sheet2) a:\data\tjdata.xls          h2502sf
genr h2502sa=h2502/h2502sf*10000
genr tj11=h2502sa
genr co8=tj11

'/***** 3 또는 6개월이동평균(3 or 6 Month Moving Average) <2.0 TC> *****/
smpl %period
for !i=1 to !k step 1
  if !i=1 or (!i=4 (!i=6)) then genr co{!i}ma=@movav(co{!i}, 3)
  else genr co{!i}ma=@movav(co{!i}, 6)
endif
next

'/*****보외 *****/
'/***** subroutine move*****/
subroutine move(series coma, scalar k, scalar m)
  if k=1 then
    %date="1989:01"
  else
    %date="1990:01"

```

```

endif
' /*****미싱기간 보외 *****/
if !i=1 or (!i=4 (!i=6)) then
  for !j=2 to 1 step -1
    smpl %date+!j-1+m %date+!j-1+m
    genr coma=coma(1)*(coma(1)/coma(2)+coma(2)/coma(3))/2
  next
else
  for !j=5 to 1 step -1
    smpl %date+!j-1+m %date+!j-1+m
    genr coma=coma(1)*(coma(1)/coma(2)+coma(2)/coma(3))/2
  next
endif
endsub

for !i=1 to !k step 1
  if !i=2 or (!i=3) then
    call move(co{!i}ma, 2, 0)
  else if !i=4 or (!i=5) then
    call move(co{!i}ma, 2, 0)
  else
    call move(co{!i}ma, 1, 0)
  endif
endif
next

' /***** 추세조정계열(TC계열)의 대칭변화율(Symmetrical Percent Change) *****/
' /***** subroutine symmetric *****/
subroutine symm(series sym_co, series coma)
  series sym_co
  genr sym_co=(coma-coma(-1))/(coma+coma(-1))*200
endsub

subroutine symm1(series sym_co, series coma)
  series sym_co
  genr sym_co=(coma-coma(-1))
endsub

for !i=1 to !k step 1
  series sym_co{!i}
next

smpl %period
  call symm(sym_co1, co1ma)
  call symm(sym_co2, co2ma)
  call symm(sym_co3, co3ma)
  call symm(sym_co4, co4ma)
  call symm(sym_co5, co5ma)
  call symm(sym_co6, co6ma)
  call symm(sym_co7, co7ma)
  call symm(sym_co8, co8ma)

' /***** 표준화인자(Standardization Factor) *****/
smpl %periodsf
  for !i=1 to !k step 1

```

```

        scalar sf_co{!i}=@mean(abs(sym_co{!i}))
    next

'/***** 표준화증감률(Standardized Rates of change) *****/
'/****** subroutine Standardized Rates *****/
    subroutine strate(series strate_co, series sym_co, scalar sf_co)
        series strate_co
        genr strate_co=sym_co/sf_co
    endsub

    for !i=1 to !k step 1
        series strate_co{!i}
    next

smpl %period
    call strate(strate_co1, sym_co1, sf_co1)
    call strate(strate_co2, sym_co2, sf_co2)
    call strate(strate_co3, sym_co3, sf_co3)
    call strate(strate_co4, sym_co4, sf_co4)
    call strate(strate_co5, sym_co5, sf_co5)
    call strate(strate_co6, sym_co6, sf_co6)
    call strate(strate_co7, sym_co7, sf_co7)
    call strate(strate_co8, sym_co8, sf_co8)

'/***** 종합증감률(Summary Rates of Changes) *****/
    series sum_rate=0
    for !i=1 to !k step 1
        genr nstrate_co{!i}=@nan(strate_co{!i}, 0)
        genr sum_rate=sum_rate+nstrate_co{!i}
    next

''/***** 구간별 지표수 산출 *****/
smpl %period
    series count=0
    for !i=1 to !k step 1
        genr num{!i}=strate_co{!i}<>na
        genr count=count+num{!i}
    next

'/***** 평균증감률(Average Rates of Changes) *****/
    smpl %fmon+1 %lmon
        genr mean_rate=(sum_rate/count)

'/***** 원지수 누적(Raw Index cumulation) *****/
    smpl %fmon %fmon
        genr rawco_idx=100

    smpl %fmon+1 %lmon
        genr rawco_idx=rawco_idx(-1)*((200+mean_rate)/(200-mean_rate))

'/***** 원지수(Raw Index, Base Year '95=100) *****/
    smpl %periodbs
        scalar bsco_raw=@mean(rawco_idx)

    smpl %period
        genr rawcoci_idx=rawco_idx/bsco_raw*100

```

```

    genr rawcoci_rate=rawcoci_idx/rawcoci_idx(-1)*100-100

'/****** 대전경기종합지수 그래프 옵션 *****/'
freeze(graph1) rawcoci_idx.line
graph1.name(1) <대전(%jj)>
graph1.addtext(t) <<대전의 동행종합지수(%jjj)>>
graph1.addtext(lt) 1995=100
graph1.shade 1988:1 1989:7
graph1.shade 1992:1 1993:1
graph1.shade 1996:3 1996:3
show graph1

'/****** 대전경기종합지수 보조지표 산출 *****/'
'/****** 순환변동치 산출(Cyclical Component of Coincident CI by
HP-Filter) *****/'
smpl %period
do rawcoci_idx.HPF(24400) hptrend_ci
genr hpcycle=rawcoci_idx/hptrend_ci*100

'/****** 전년동월비 산출 *****/'
smpl %period
genr ms_rate=(rawcoci_idx/rawcoci_idx(-12))*100-100

'/****** 평활화 6개월전비 산출 *****/'
smpl %period
genr s6ms_ci=(rawcoci_idx(-12)+rawcoci_idx(-11)+rawcoci_idx(-10)+
rawcoci_idx(-9)+rawcoci_idx(-8)+rawcoci_idx(-7)+rawcoci_idx(-6)+rawcoci_idx(-5)+
rawcoci_idx(-4)+rawcoci_idx(-3)+rawcoci_idx(-2)+rawcoci_idx(-1))/12
genr s6ms_rate=(rawcoci_idx/s6ms_ci)^(12/6.5)*100-100

'/****** 평활화 12개월전비 산출 *****/'
smpl %period
genr s12ms_ci=(rawcoci_idx(-17)+rawcoci_idx(-16)+rawcoci_idx(-15)+
rawcoci_idx(-14)+rawcoci_idx(-13)+rawcoci_idx(-12)+rawcoci_idx(-11)+rawcoci_idx(-10)+
rawcoci_idx(-9)+rawcoci_idx(-8)+rawcoci_idx(-7)+rawcoci_idx(-6))/12
genr s12ms_rate=(rawcoci_idx/s12ms_ci)*100-100

'/****** 대전경기종합지수 보조지표 그래프 옵션 *****/'
'/****** 순환변동치 *****/'
freeze(graph2) hpcycle.line
graph2.name(1) <대전>
graph2.addtext(t) <대전 동행지수의 순환변동치>
graph2.addtext(lt) trend=100
graph2.shade 1988:1 1989:7
graph2.shade 1992:1 1993:1
graph2.shade 1996:3 1996:3
show graph2

'/****** 전년동월비 *****/'
freeze(graph3) ms_rate.line
graph3.name(1) <대전>
graph3.addtext(t) <대전 동행지수의 전년동월비>
graph3.addtext(lt) %
graph3.shade 1988:1 1989:7
graph3.shade 1992:1 1993:1
graph3.shade 1996:3 1996:3
show graph3

```



```

' /***** 평활화 6개월전비 *****/
freeze(graph4) s6ms_rate.line
graph4.name(1) <대전>
graph4.addtext(t) <대전 동행지수의 평활화 6개월전비>
graph4.addtext(lt) %
graph4.shade 1988:1 1989:7
graph4.shade 1992:1 1993:1
graph4.shade 1996:3 1996:3
show graph4

' /***** 평활화 12개월전비 *****/
freeze(graph5) s12ms_rate.line
graph5.name(1) <대전>
graph5.addtext(t) <대전 동행지수의 평활화 12개월전비>
graph5.addtext(lt) %
graph5.shade 1988:1 1989:7
graph5.shade 1992:1 1993:1
graph5.shade 1996:3 1996:3
show graph5

' /*****/
freeze(group1) graph1 graph2 graph3 graph4 graph5
show group1

save a:\pgm\coin99.wf1

```