

CHAPTER 2.

데이터 만들기

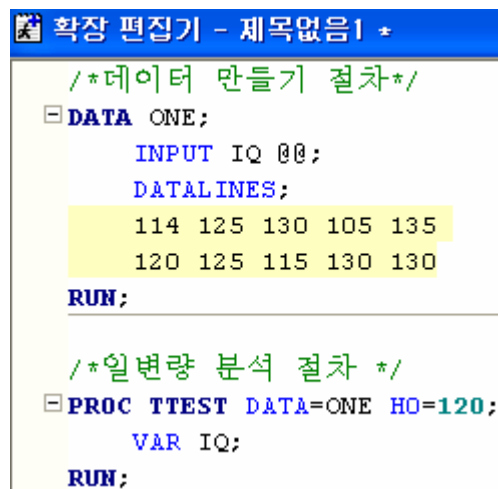
SAS를 이용하여 원하는 분석(PROC 이용)을 하려면 SAS 데이터가 만들어져 있어야 한다. SAS 데이터를 만드는 방법은 다음과 같다. 이들 방법 중 가장 권하고 싶은 것은 방법 ③에서 확장 편집기의 프로그램 이용 방법이다. 절차가 간편하고 가장 많이 사용되는 방법이다.

- ① 프로그램 에디터(확장 편집기)에서 입력하기
- ② SAS 메뉴의 분석 창(analyst)에서 입력하기 (스프레드시트 형태 데이터 입력)
- ③ 외부 데이터 불러오기 (메뉴 방법과 확장 편집기 프로그램 방법)

2.1 프로그램 구조

SAS 프로그램은 데이터를 만들거나 변환하는 데이터 단계(DATA step), 데이터를 처리하여 분석 작업을 실행하는 PROC(EDURE) 단계로 구성되어 있다. 각각은 DATA, PROC에 의해 시작되고 끝은 RUN;에 의해 단계가 종료된다. RUN을 매 단계 사용하지 않고 프로그램 마지막에만 사용되어도 되나 프로그램 구조의 이해를 돕기 위하여 매 단계 사용하기를 권장한다.

다음은 10명 학생들의 IQ를 조사한 후 이들의 IQ가 전 학생들의 평균 IQ 120과 같은지 알아보기 위하여 작성한 프로그램이다. 이 프로그램은 하나의 데이터 단계와 하나의 PROC 단계로 구성되어 있다.



```

/*데이터 만들기 절차*/
DATA ONE;
  INPUT IQ @@;
  DATALINES;
  114 125 130 105 135
  120 125 115 130 130
RUN;

/*일변량 분석 절차 */
PROC TTEST DATA=ONE HO=120;
  VAR IQ;
RUN;
  
```

SAS 프로그램 작성 규칙을 정리하면 다음과 같다.

(1)* 문장 */은 주석문(comment statement)으로 사이의 문장은 실행 결과에 영향을 미치지 않는다. 작성된 프로그램을 설명하기 위해 사용된다.

(2)모든 문장은 세미콜론(;)으로 끝난다. 문장이 여러 행에 걸쳐 작성되더라도 세미콜론이 없다면 하나의 문장으로 간주된다.

(3)PROC 단계에 사용되는 데이터는 데이터 단계를 통해 만들어져 있어야 한다.

(4)프로그램 안에서 대소문자 구별은 없다. 단 관측치가 문자인 경우만 구별된다.

(5)데이터 이름(ONE), 변수 이름(IQ)은 영어, 숫자, 그리고 특수문자 '_'로만 구성되어야 한다. 그리고 반드시 영어로 시작해야 한다.

(6)DATA, PROC, RUN은 진한 파란색, 데이터 부분은 노란색, 옵션은 파란색, 숫자와 주석 문은 초록색, 옵션 지정 내용은 검정색으로 나타난다. 오류 옵션이나 문장은 빨간색으로 표시된다.

2.2 데이터 만들기

다음은 실습을 위해 사용하게 될 예제 데이터이다. 암 종류(Bronchus기관지, Stomach위)에 따른 암 환자 생존 시간(일)을 조사한 것이다.

TIME	TYPE	TIME	TYPE	TIME	TYPE	TIME	TYPE	TIME	TYPE
124	Stomach	146	Stomach	64	Bronchus	1112	Stomach	450	Bronchus
42	Stomach	340	Stomach	155	Bronchus	46	Stomach	246	Bronchus
25	Stomach	396	Stomach	859	Bronchus	103	Stomach	166	Bronchus
45	Stomach	81	Bronchus	151	Bronchus	876	Stomach	63	Bronchus
412	Stomach	461	Bronchus	166	Bronchus	223	Bronchus	72	Bronchus
51	Stomach	20	Bronchus	37	Bronchus	138	Bronchus	245	Bronchus

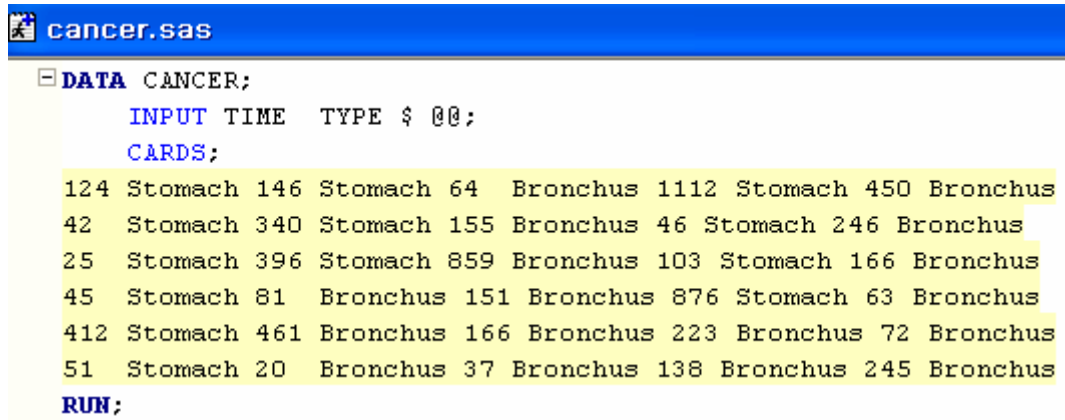
2.2.1 확장 편집기 이용

수집된 데이터를 SAS 데이터로 만드는 데이터 단계(DATA step)이다. 확장 편집기의 프로그램 명령어나 변수 이름은 대소문자 구별할 필요는 없으나 입력된 데이터의 관측치는 대소문자 구별해야 한다. DATA CANCER;나 data cancer;는 서로 동일하다. 또한 TIME과 time 역시 동일하다. 그러나 “Stomach”와 “stomach”는 SAS에서는 서로 다르게 인식됨을 유의하기 바란다.

SAS 프로그램에서 가장 중요한 것은 문장의 끝에는 반드시 세미콜론(;)으로 끝 맺어야 한다는 것이다. 세미콜론이 빠지면 SAS는 하나의 문장으로 인식하게 된다. 확장 편집기 위에 있는 CANCER.SAS는 프로그램 이름이다. 저장되지 않은 프로그램은 “제목없음”으로 표시된다. 프로그램을 저장하려면 CTRL+S를 누르거나 **파일(F) ▶ 다른 이름으로 저장(S)...**을

눌러 저장을 위한 팝업 창이 나타나면 폴더와 파일 이름을 설정하고 저장하면 된다.

2.1장에서 언급하였듯이 데이터를 컴퓨터 파일에 입력할 때는 열은 변수, 행은 개체의 관측치를 넣은 것이 일반적이다. 확장 편집기에서 자료를 입력하는



```

cancer.sas
DATA CANCER;
  INPUT TIME TYPE $ @@;
  CARDS;
124 Stomach 146 Stomach 64 Bronchus 1112 Stomach 450 Bronchus
42 Stomach 340 Stomach 155 Bronchus 46 Stomach 246 Bronchus
25 Stomach 396 Stomach 859 Bronchus 103 Stomach 166 Bronchus
45 Stomach 81 Bronchus 151 Bronchus 876 Stomach 63 Bronchus
412 Stomach 461 Bronchus 166 Bronchus 223 Bronchus 72 Bronchus
51 Stomach 20 Bronchus 37 Bronchus 138 Bronchus 245 Bronchus
RUN;
  
```

(1)DATA~RUN;

SAS 데이터 만드는 단계의 시작 문장과 끝 문장이다. 데이터 단계나 PROCEDURE 단계의 시작인 DATA나 PROCEDURE, 끝인 RUN은 진하게 파란색으로 표시된다. 다른 프로그램이 이어지면 RUN;을 생략해도 되나(즉 프로그램 마지막에만 이 문장이 있으면 OK) 프로그램 작성의 체계화를 위하여 매 단계 끝에 사용하길 권한다.

DATA 위의 "CANCER"는 만들어지는 SAS 데이터 이름이다. DATA 이름은 문자, 숫자, 특수문자(_정도 사용 가능) 혼용 사용해도 되나 첫 글자는 반드시 영어이고 공백은 사용하지는 안 된다. 예전에는 이름으로 8자가 최대였으나 이제는 8자 이상 사용이 가능하다.

SAS는 WORK.CANCER라는 데이터를 만들고 WORK 라이브러리에 임시 저장한다. 임시 저장의 의미는 SAS 종료와 동시에 사라짐을 의미한다. SAS 데이터 만드는 프로그램(CANCER.SAS)을 저장하였으니 이 프로그램을 다시 불러와 실행하면 CANCER SAS 데이터가 만들어지므로 굳이 영구 SAS 데이터(SAS 종료 후에도 남아 있음)를 만들 필요는 없다. 영구 SAS 데이터 만드는 방법은 2.3.2절에서 다룰 것이다.

(2)INPUT~;

변수명과 변수 타입을 정의하는 문장으로 변수명도 데이터 이름 부여 방법을 따르면 된

다. 읽어 오는 데이터가 문자이면 \$(문자열 구분)을 변수명 다음에 넣는다. 숫자는 콤마 없이, 문자는 공백 없이 입력하는 것이 읽어오는데 편리하다. 물론 콤마가 있는 숫자, 공백이 있는 문자를 읽어 오는 명령어도 존재한다. 이것은 2.4절에서 다룰 것이다.

변수이름은 영어와 숫자, 그리고 일부 특수문자 “_” 정도만 된다. 물론 변수 이름에는 공백이 있으면 안되고 데이터 이름과 같이 반드시 숫자로 시작해야 한다. 그러므로 “a~z”를 사용하거나 관측된 변수에 적절한 영어 이름을 부여하면 된다. 변수 이름으로 8자 이상 가능하다. 위 프로그램에서는 시간을 TIME, 암 환자 종류를 TYPE이라는 이름을 부여하였다.

변수 이름이 길어지면 프로그램 내에 다시 사용하는 것이 다소 번거롭고 그렇다고 알파벳을 사용하면 출력 결과를 이해하는데 어려움이 있다. 이 때에는 LABEL 문을 사용하면 된다. LABEL 문을 사용하면 한글 출력이 가능하다. 아래 프로그램을 실행하면 변수 이름 T는 “시간”으로 출력된다. 이처럼 LABEL 문은 데이터 설정 단계 이후에 사용되는 모든 PROCEDURE 결과에 변수 T가 “시간”으로 출력되게 한다.

```

cancer.sas *
DATA CANCEr;
  INPUT T TYPE $ @@;
  LABEL T="시간";
  CARDS;
124 Stomach 146 Stomach 64 Bronchus 1112 Stomach 450 Bronchus
PROC MEANS data=cancer MEAN STD;
RUN;

```

The MEANS Procedure

분석 변수 : T 시간

평균값	표준편차
243.8333333	274.4652586

변수 사이에는 하나의 공백을 둔다. 한 개체에 대한 데이터는 한 줄에 넣는 것이 원칙이다. 만약 한 줄에 여러 개체(사람)의 데이터를 넣는 경우 @@를 사용해야 된다. 만약 위의 프로그램에서 @@를 제외하면 CANCEr 데이터에는 2개 변수, 6개의 관측치가 생성된다. 다음 결과는 @@ 없이 프로그램을 실행해 생성한 CANCEr 데이터 출력 결과이다.

Obs	TIME	TYPE
1	124	Stomach
2	42	Stomach
3	25	Stomach
4	45	Stomach
5	412	Stomach
6	51	Stomach

(3) CARDS;

데이터 입력이 시작되었다고 SAS에게 알린다. DATALINES; 을 사용해도 된다. 입력된 데이터는 사용자가 구별하기 쉽도록 노란 바탕색으로 표시된다. SAS는 그 부분만을 데이터로 받아들인다는 의미이다.

(4) RUN;

DATA 단계, PROCEDURE 단계의 끝을 알린다. 프로그램 여러 단계로 구성된 경우 마지막 단계에만 RUN;을 사용하면 된다. 그러나 프로그램 체계화를 위하여 매 단계 사용하는 것이 효율적이다.

데이터 생성 프로그램(CANCER.SAS)이 실행한 후 SAS 데이터가 성공적으로 만들어졌는지 다음 방법을 통해 확인할 수 있다.

① LOG 창

관측치 30개, 변수 2개인 SAS 데이터가 WORK 라이브러리 안에 CANCER 이름으로 성공적으로 만들어졌음을 알 수 있다.


```

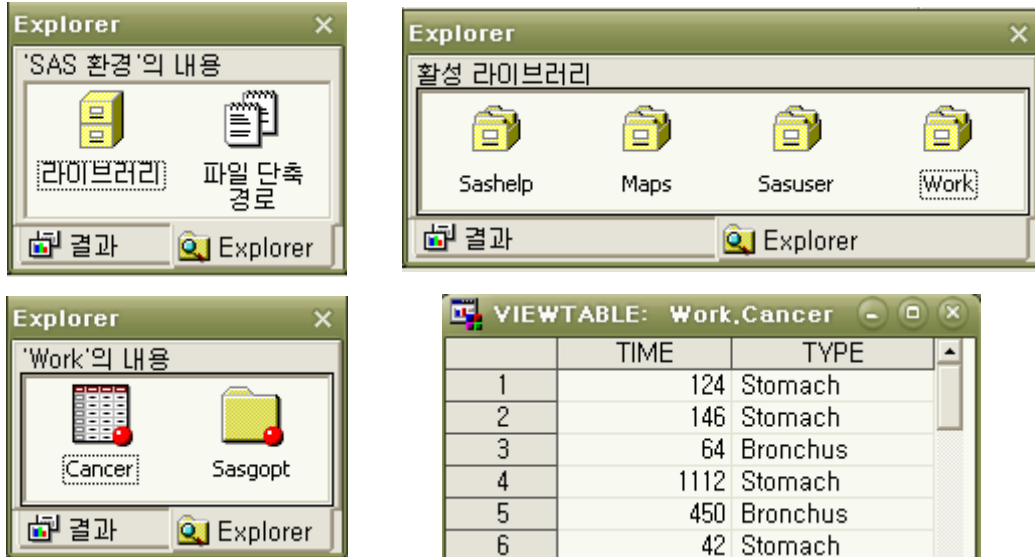
로그 - (제목없음)
69 DATA CANCER;
70     INPUT TIME TYPE $ @@;
71     CARDS;

NOTE: INPUT 문이 행의 끝에 도달했을 때 SAS는 다음 행으로 이동합니다.
NOTE: 데이터셋 'WORK.CANCER'은(는) 30개 관측치, 2개 변수를 가지고 있습니다.
NOTE: DATA 문장 실행:
      실행 시간      0.04 초
      cpu 시간      0.03 초

```

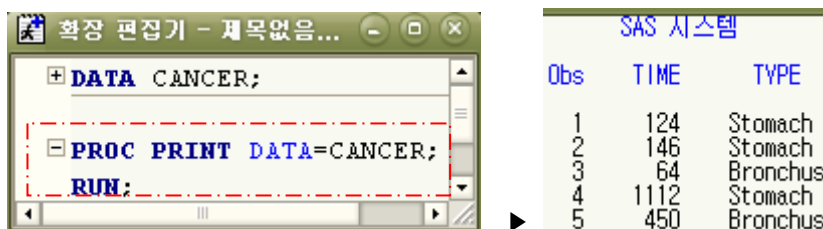
②탐색기 폴더에서

만들어진 SAS 데이터는 WORK 라이브러리에 임시적으로 저장된다. WORK 라이브러리에 저장된 SAS 데이터는 SAS 종료와 함께 사라진다. 이를 임시 데이터라 한다. SAS 데이터 CANCER의 내용을 보려면 을 두 번 클릭하면 VIEWTABLE 창이 열리고 데이터 내용이 출력된다.



③출력 창에서

확장 편집기에서 다음 프로그램을 실행하면(가장 일반적인 방법) 출력 창에 데이터 내용이 출력된다. 데이터 단계 첫 행을 보면 `DATA`에서 `DATA`로 바뀌어 있음을 알 수 있다. 데이터 단계의 문장 모두가 한 행에 숨겨진 경우 `DATA`로 표시된다. 이것은 프로그램이 긴 경우 프로그램 작업이 편하도록





출력 창 제목 바꾸기

출력 결과 제목의 디폴트(default)는 "SAS 시스템"이다. 같은 행에 시스템 날짜와 페이지 번호가 자동 출력된다. 출력 제목을 바꾸려면 TITLE 문장을 사용하면 된다.

cancer.sas *		CANCER DATA		
		Obs	TIME	TYPE
+ DATA CANCER;		1	124	Stomach
TITLE 'CANCER DATA';		2	146	Stomach
- PROC PRINT DATA=CANCER;		3	64	Bronchus
RUN;		4	1112	Stomach
		5	450	Bronchus
		6	42	Stomach

제목은 두 줄 이상으로 하려면 TITLE1과 TITLE2, ...을 사용하면 된다. 날짜나 페이지 번호가 제목에 출력되지 않게 하려면 OPTIONS 문장을 사용하면 된다. NODATE 옵션은 날짜, NONUMBER 옵션은 페이지 번호를 출력하지 말라는 옵션이다.

확장 편집기 - 제목없음1 *		CANCER DATA MAKING SAS DATA		
		Obs	TIME	TYPE
+ DATA CANCER;		1	124	Stomach
OPTIONS NODATE NONUMBER;		2	146	Stomach
TITLE1 'CANCER DATA';		3	64	Bronchus
TITLE2 'MAKING SAS DATA';		4	1112	Stomach
- PROC PRINT DATA=CANCER;		5	450	Bronchus
RUN;		6	42	Stomach
		7	340	Stomach
		8	155	Bronchus

제목 문장은 한 번 사용하면 SAS 종료할 때까지 다른 TITLE 문을 설정하지 않는 한 동일하게 출력된다. 제목 문을 Reset(제목이 출력되지 않음)하기 원하면 제목 설정 없이 TITLE; 문장을 사용하면 된다.

프로그램 중 일부만 실행하고 싶으면 아래와 같이 마우스로 선택한 후 실행하면 된다. 아래 프로그램은 TITLE문은 없어지게 하고 CANCER 데이터의 TIME 변수의 평균, 분산, 최대값, 최소값이 출력되는 프로그램이다.


```

+ DATA CANCER;
TITLE;
- PROC MEANS DATA=CANCER;
  VAR TIME;
RUN;

```

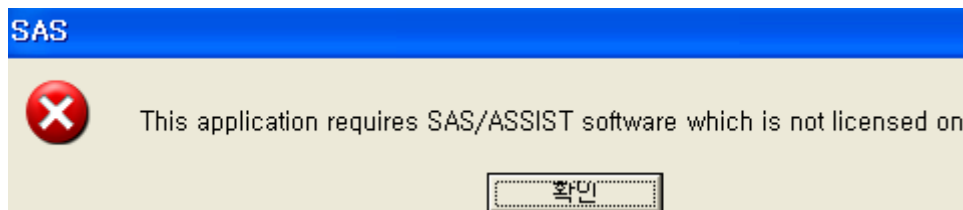
The MEANS Procedure

분석 변수 : TIME

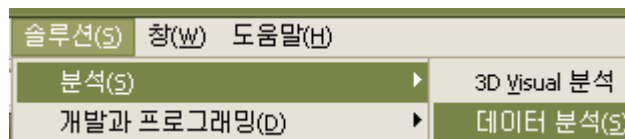
N	평균값	표준편차	최소값	최대값
30	243.8333333	274.4652586	20.0000000	1112.00

2.2.2 메뉴 분석 창 이용

엑셀과 같이 스프레드시트 형식으로 데이터를 입력하려면 SAS/ASSIST 제품에 대한 라이선스가 있어야 한다. 라이선스가 없는 경우 아래 팝업 창이 뜨고 실행되지 않는다.



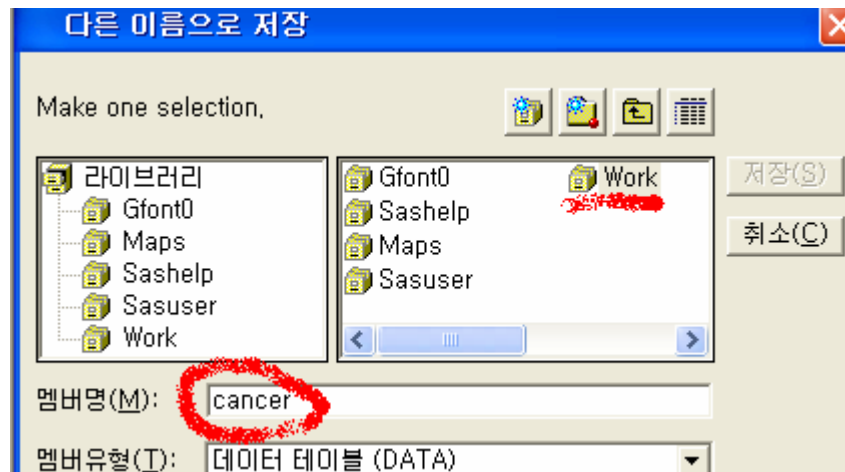
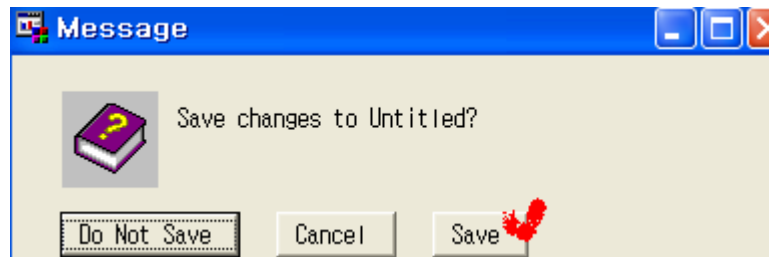
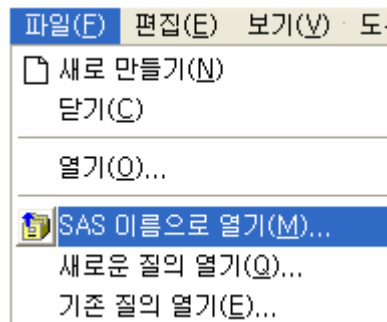
SAS 메뉴서 솔루션(S) ▶ 데이터 분석(S)을 선택하면 왼쪽 화면에는 제목, 오른쪽 화면에는 자료를 입력할 수 있는 스프레드시트 창이 뜬다. 데이터 입력은 엑셀에서와 같이 하면 된다. 반드시 열은 변수, 행은 개체의 관측치를 입력해야 한다.



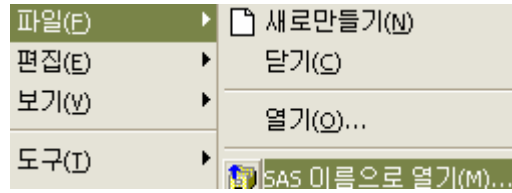
Analyst: (new project)

Untitled (NEW)		
	A	B
1	124	Stomach
2	42	Stomach
3		

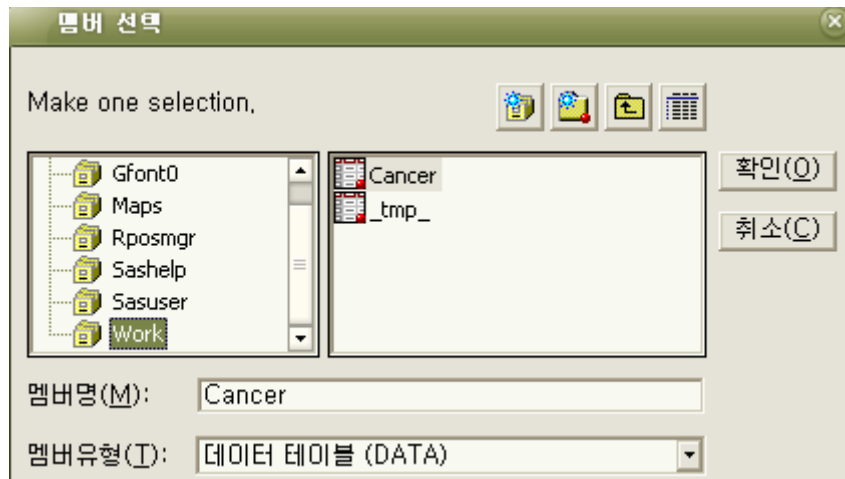
데이터 입력이 끝나면 다음 절차에 의해 SAS 데이터에 저장하면 된다. WORK 라이브러리에 “CANCER”라는 이름으로 저장된다.



이미 만들어진(기존의) SAS 데이터를 이용하여(예를 들어 CANCER데이터) SAS/ASSIST를 이용하려면 스프레드시트 창에서 오른쪽 마우스를 눌러 팝업 창이 나타나도록 한다.



SAS 이름으로 열기를 선택하고 아래와 같이 CANCER 데이터를 선택하면 된다.



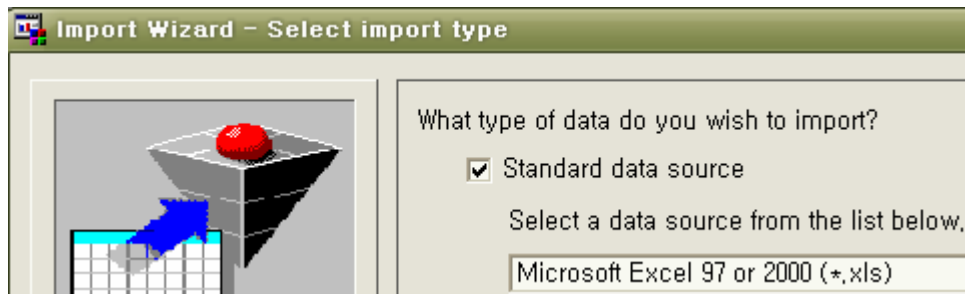
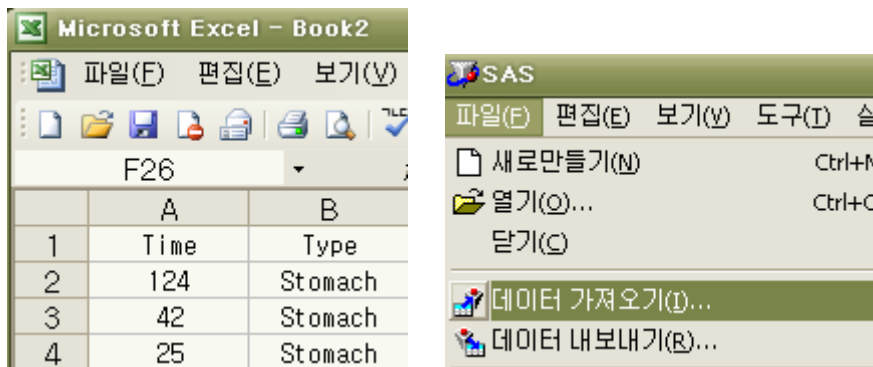
	TIME	TYPE
1	124	Stomach
2	146	Stomach
3	64	Bronchus
4	1112	Stomach
5	450	Bronchus
6	42	Stomach
7	340	Stomach
8	155	Bronchus

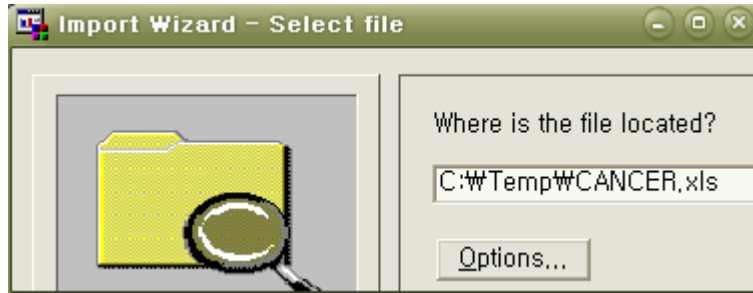
2.2.3 외부 데이터 불러오기

엑셀 데이터

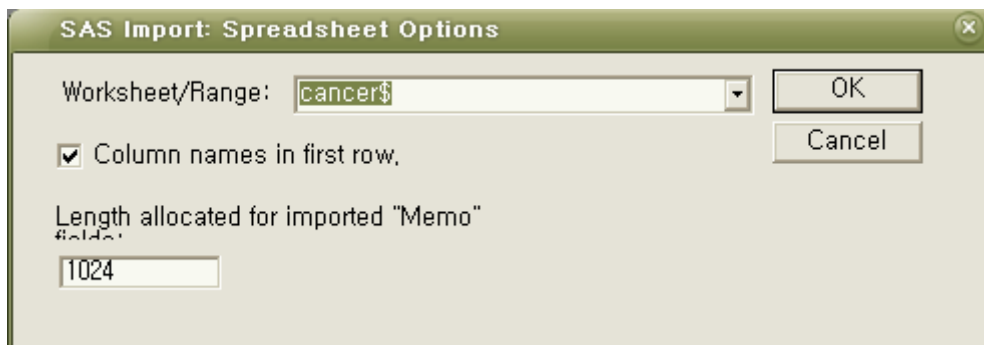
엑셀에 데이터를 입력하는 경우 열은 변수, 행은 개체에 대한 관측치를 입력하면 된다. 첫 행은 SAS에서 사용하게 될 변수 이름을 지정하는 것이 좋다. 엑셀의 첫 행에 지정된 변수 이름이 그대로 SAS 데이터 변수 이름으로 사용되므로 변수 이름을 부여할 때 SAS 변수 이름 설정 규칙을 지켜야 한다. 첫 행을 변수 이름 설정으로 사용하지 않았거나 SAS 변수 이름 설정 규칙을 따르지 않는 경우 SAS는 변수 이름으로 `_COL1`, `_COL2`,... 로 자동 설정하게 된다.

엑셀 데이터가 C:\TEMP 폴더에 `CANCER.XLS` 이름으로 저장되어 있다고 하자.

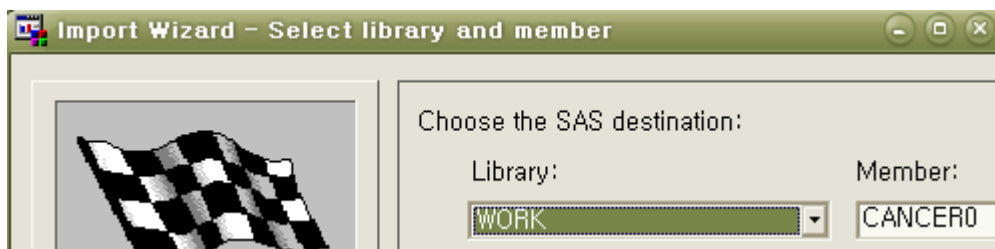




Options...을 선택하여 다음과 같이 설정한다. 엑셀 문서 내 시트(sheet)가 2개 이상이거나 첫 행에 변수 이름을 지정하지 않은 경우 사용한다. 엑셀 문서 내 데이터 시트가 하나이고 첫 행을 변수 이름으로 사용하였다면 조건 변경 없이 디폴트 설정을 이용하면 된다.

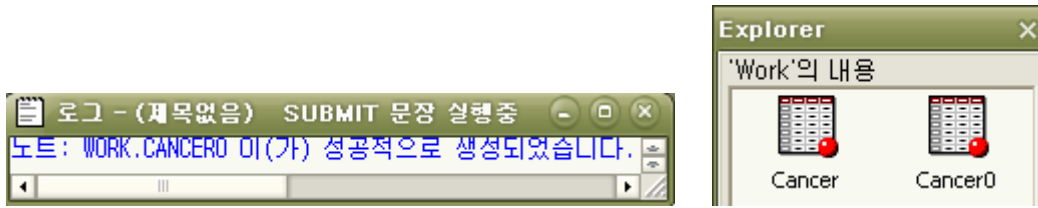


Library:에는 만들어진 SAS 데이터를 저장할 라이브러리를 설정하고 **Member:**에는 SAS 데이터 이름을 준다. 아래 화면을 실행하면 WORK 라이브러리에 "CANCER0" SAS 데이터가 저장된다.



다음 화면에서는 아무 설정 없이 **Finish** 옵션을 누른다. SAS 데이터가 성공적으로 만들어지면 로그 창에 다음 노트가 출력된다. 탐색기 폴더의 WORK 라이브러리에도

당연히 CANCERO SAS 데이터가 있다.



엑셀 데이터 SAS 데이터 만들기 쉬운 방법

확장 편집기에서 SAS 데이터를 생성하는 것과는 달리 엑셀 데이터 불러오기의 경우 이런 방법을 사용하면 SAS 종료 후에 다시 이 데이터를 이용하려면 위 과정을 반복해야 하는 번거로움이 있다. 이를 해결하기 위해서는 다음 2가지 방법이 있다.

- ①영구 데이터 만들기: 2.3.2절 참고
- ②확장 편집기에서 프로그램 실행

SHEET 옵션은 엑셀 데이터에 시트가 여러 개 있는 경우 사용한다. 시트가 하나 밖에 없거나 가장 앞에 있는 시트 데이터를 사용하려면 사용하지 않아도 된다. **GETNAMES** 옵션은 엑셀 데이터 첫 행을 변수 이름으로 사용하느냐에 대한 설정이다. 첫 행을 변수 이름으로 사용한 경우에는 이 옵션(**GETNAMES=YES**)을 설정하지 않아도 된다.

이 프로그램만 저장해 놓으면 엑셀 데이터 불러오기 작업 없이 엑셀 데이터를 SAS 데이터로 만들 수 있다.

```

cancer.sas *
PROC IMPORT DATAFILE="D:\TEMP\CANCER.xls"
              OUT=WORK.CANCER1 DBMS=EXCEL;
              GETNAMES=NO;
RUN;
  
```

엑셀 데이터를 SAS 데이터로 만들려면 SAS/ACCESS 라이선스가 있어야 한다. 라이선스가 없는 경우에는 엑셀에서 다음 형식으로 데이터 파일을 저장한 후 텍스트 데이터 SAS

데이터 만들기 방법을 사용하면 된다.

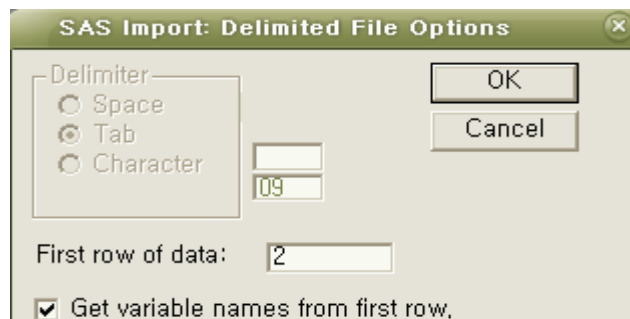
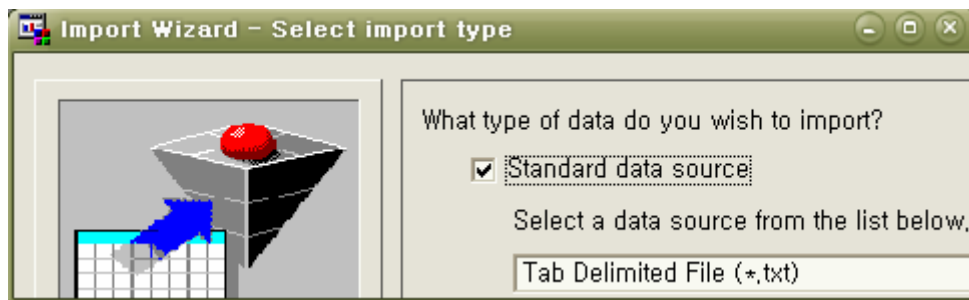
- ① 파일 형식(T): 텍스트 (탭으로 분리) (*.txt)
- ② 파일 형식(T): 텍스트 (공백으로 분리) (*.prn)

텍스트 데이터 (TAB 키를 사용한 경우)

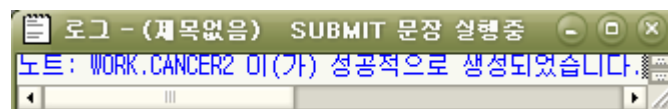
cancer - 모장		
파일(E)	편집(E)	서식(O)
Time	Type	
124	Stomach	
42	Stomach	

▶ 첫 행은 변수명이 있다.

다음 화면을 제외하고는 엑셀 데이터 가져오기 방법과 동일하다.

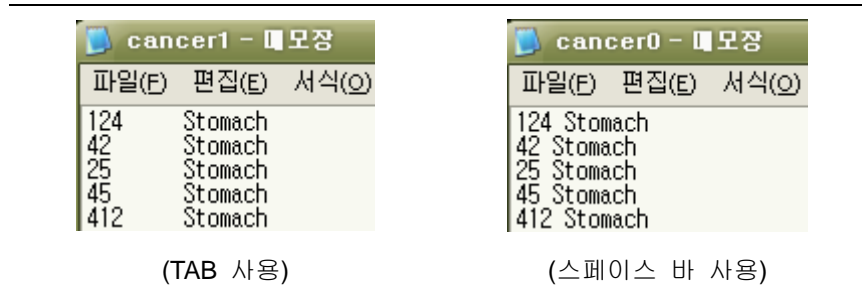


첫 행이 변수명이었다. 로그 창에 다음 메시지가 나타난다.



텍스트 데이터 (공백 사용한 경우)

텍스트 형식의 데이터이고 변수 사이에 하나 이상의 공백을 사용하여 데이터가 텍스트 형식으로 입력된 경우 확장 편집기에서 프로그램을 사용하는 방법이 가장 편리한 방법이다. 프로그램만 저장해 놓으면 SAS 데이터 만들기는 계속할 수 있다.



스페이스 바를 이용하여 데이터가 입력한 경우에는 다음 프로그램을 사용하면 된다.

```

DATA CANCER2;
  INFILE "C:\TEMP\CANCERO.TXT";
  INPUT TIME TYPE $;
RUN;

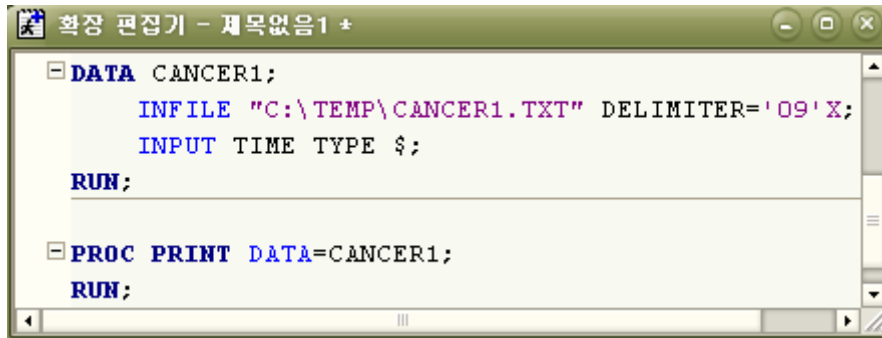
PROC PRINT DATA=CANCER2;
RUN;

```

(1) INFILE;

텍스트 파일이 있는 위치를 설정해 준다. 위 프로그램에서는 텍스트 데이터가 C:\TEMP 폴더 아래 "CANCERO.TXT"라는 이름으로 저장되어 있다는 것을 의미한다. INPUT 문장은 2.2.1절과 동일하다.

텍스트 데이터이고 TAB 키를 이용하여 데이터가 입력된 경우에는 delimiter 옵션을 사용하여 SAS 데이터를 만들면 된다.



```

DATA CANCER1;
  INFILE "C:\TEMP\CANCER1.TXT" DELIMITER='09' X;
  INPUT TIME TYPE $;
RUN;

PROC PRINT DATA=CANCER1;
RUN;

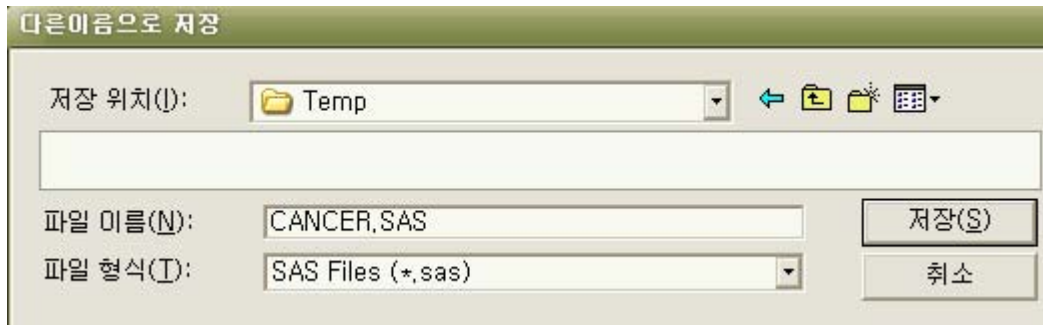
```

2.3 데이터 저장하기

2.3.1 프로그램 저장하기

SAS 데이터는 굳이 저장할 필요는 없다. 2.2절에서 확장 편집기를 이용하면 내부 입력은 물론 외부데이터(엑셀 데이터, 텍스트 데이터) 등도 읽어올 수 있음을 알았다. 다음과 같이 데이터 만들기 프로그램만 저장해 놓고 SAS 실행 후 이 프로그램을 확장 편집기로 불러와 실행해 주면 SAS 데이터가 만들어진다. 아래 과정에 의하면 확장 편집기에 작성된 프로그램이 C:\TEMP 폴더 아래 CANCER.SAS로 저장된다.

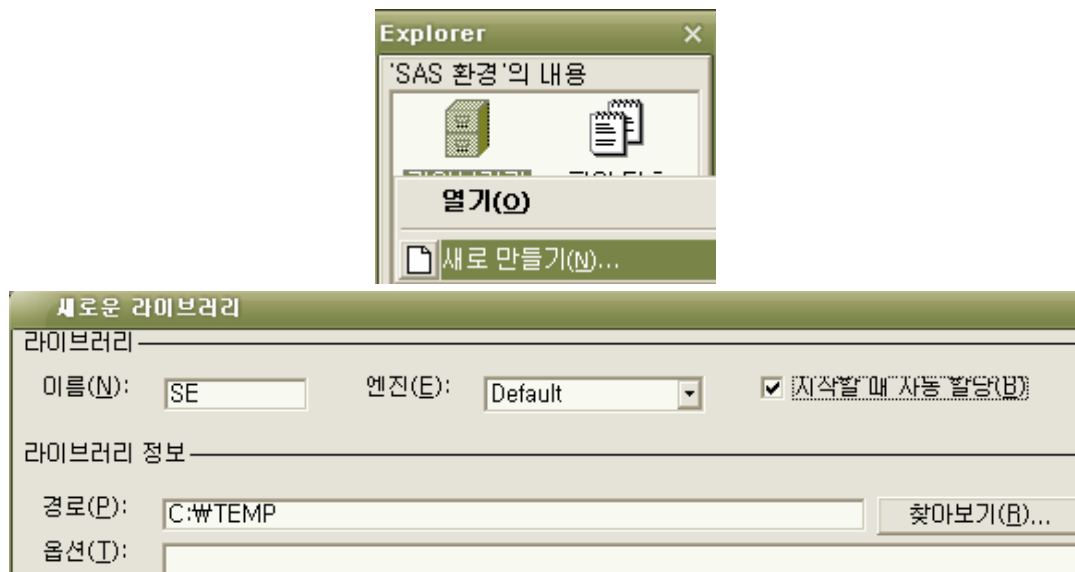




2.3.2 영구 데이터 만들기

WORK 라이브러리에 있는 저장된 SAS 데이터는 임시 데이터로 SAS가 종료되면 사라진다. 그러므로 SAS 종료 후에도 남아 있는 SAS 데이터를 영구 데이터라 하는데 아래 과정을 거쳐 만들어진다. 2.2절에서 살펴보았듯이 SAS 데이터 만들기는 확장 편집기에서 가능하므로 DATA 단계 프로그램만 저장해 했다가 다시 실행하면 굳이 영구 SAS 데이터를 만들지 않아도 된다. SAS 영구 데이터를 만드는 경우는 이례적이다.

탐색기에서 라이브러리 이름이 있는 곳까지 가서 오른쪽 마우스 버튼을 누르면 팝업 창이 나타나는데 여기서 새로 만들기 메뉴를 선택한다.

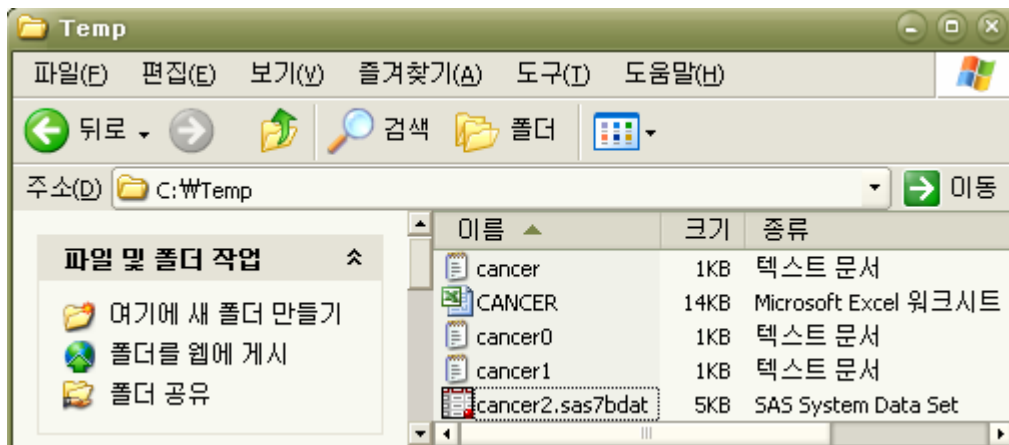


시작할 때 자동 할당을 체크하면 SAS 시작 시 “Se” 이름의 라이브러리가 생기고

C:\TEMP 폴더가 지정된다.



Se 라이브러리에 CANCER2 데이터가 저장된다. 윈도우 탐색기(창+E)를 열어 C:\TEMP 폴더에 가면 다음 파일이 있다. SAS 데이터의 확장자는 "SAS7BDAT"이다.

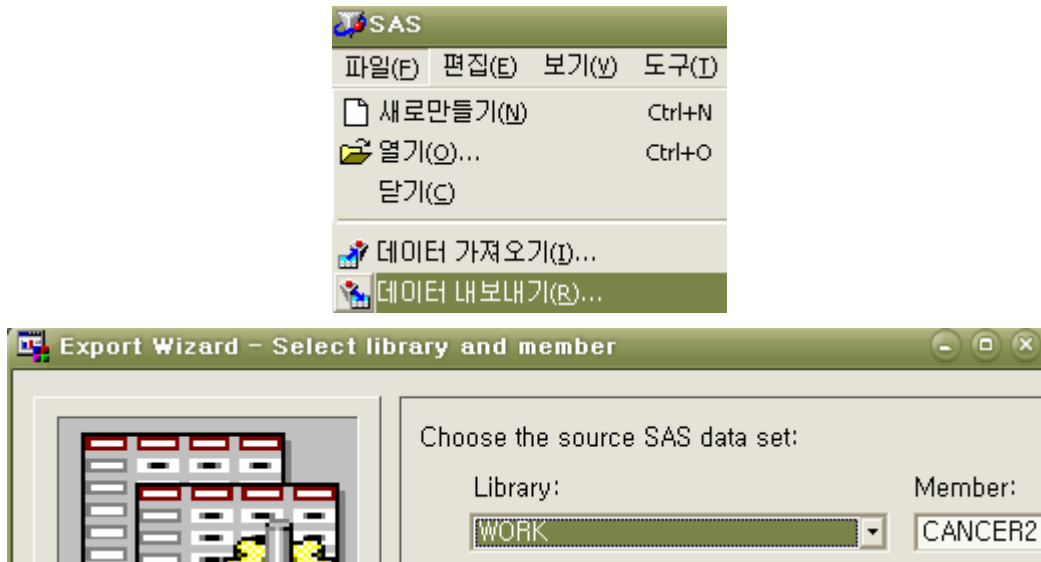


SAS를 종료하고 다시 시작하면 탐색기에 "SE" 라이브러리가 있고 그 안에 SAS 데이터

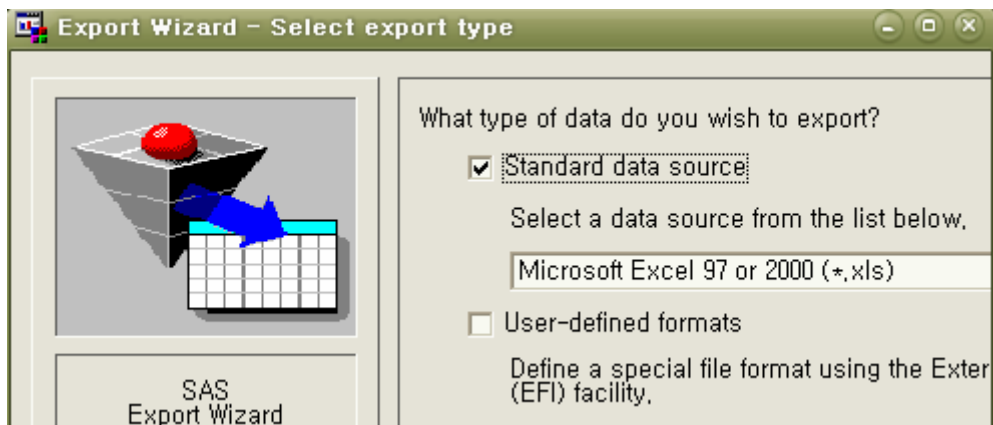
CANCER2가 있다.

2.3.3 데이터 다른 형식 저장하기

SAS 데이터를 다른 형식으로 저장하려면 다음과 같이 실행한다.



위 창에서 Next를 클릭하면 다음과 같이 Export Wizard 창이 뜬다.



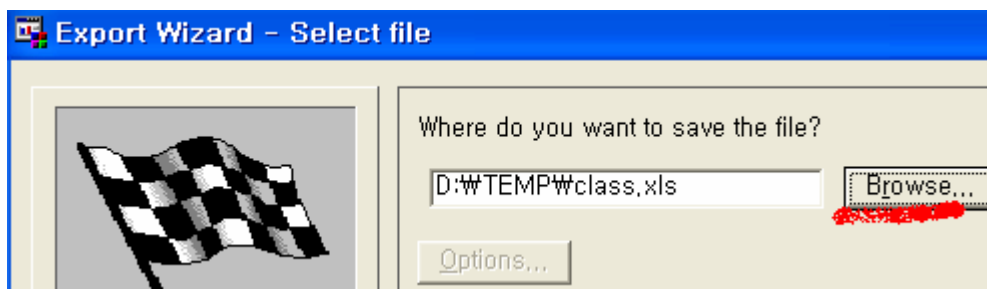
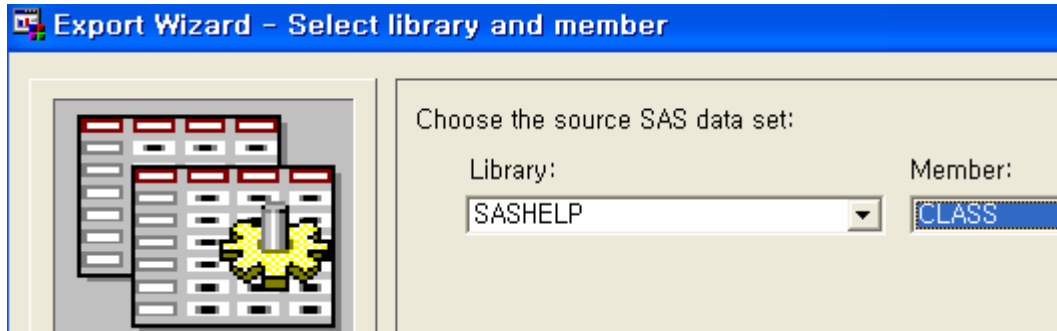
위 창에서 파일 형식을 엑셀 형식으로 지정하고 Next를 클릭하고 저장하고 싶은 파일 경

로와 이름을 지정해주면 엑셀 파일로 변환된 자료가 생성된다.



EXAMPLE: SAS 데이터 엑셀 저장

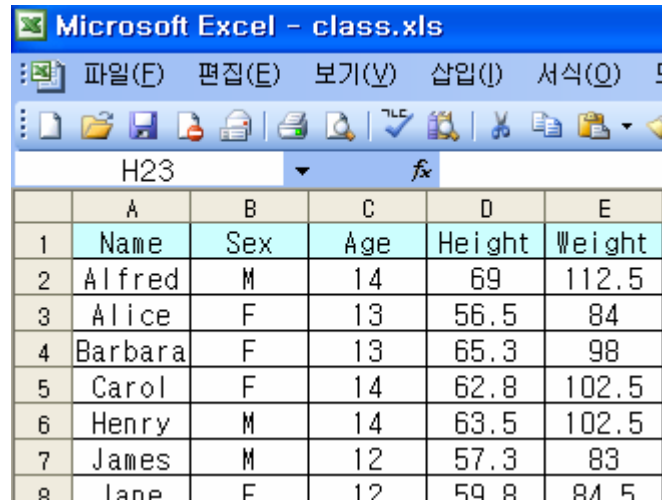
SASHELP 라이브러리에 있는 예제 데이터 CLASS를 엑셀 데이터로 저장하여 보자.



Finish 선택하면 로그 창에는 다음 메시지 출력되고 D:\TEMP 폴더 안에 CLASS.xls가 저장되어 있다.

로그 - (제목없음) SUBMIT 문장 실행중
 NOTE: 'D:\TEMP\class.xls'이(가) 성공적으로 생성되었습니다.

CLASS.xls를 열어 보자.



	A	B	C	D	E
1	Name	Sex	Age	Height	Weight
2	Alfred	M	14	69	112.5
3	Alice	F	13	56.5	84
4	Barbara	F	13	65.3	98
5	Carol	F	14	62.8	102.5
6	Henry	M	14	63.5	102.5
7	James	M	12	57.3	83
8	Jane	F	12	59.8	81.5



EXAMPLE: SAS 데이터 엑셀 저장(확장 편집기에서)

엑셀 데이터 불러오는 경우 일반적으로 SAS 도구 메뉴를 이용하거나 확장 편집기에서 PROC IMPORT (2.2.3절) 이용하면 된다. SAS 데이터를 엑셀 파일로 저장할 때도 확장 편집기를 이용하면 된다. 첫 행과 두 번째 행은 하나의 문장이므로 첫 행 마지막에는 세미콜론이 없음을 유의하기 바란다. DBMS(Data Base Manage System) 옵션은 파일 형식을 지정하는 것이다. 변수 사이에 TAB이 있는 텍스트 파일로 저장하려면 DBMS=DLM을 사용하면 된다. 만약 MS-ACCESS 형식으로 저장하려면 DBMS=ACCESS로 하면 된다.

```

확장 편집기 - 제목없음2 *
PROC EXPORT DATA=SASHELP.CLASS OUTFILE='D:\TEMP\CLASS.XLS'
  DBMS=EXCEL REPLACE;
RUN;
  
```

엑셀 파일이 성공적으로 만들어지면 로그 창에 아래 노트가 출력되고 D:\TEMP 폴더\n 안에 CLASS.xls가 저장된다.

```
156 PROC EXPORT DATA=SASHELP.CLASS
157     OUTFILE='D:\TEMP\CLASS.XLS'
158     DBMS=EXCEL REPLACE;
NOTE: System DBCS translation table KOREAN registered.
NOTE: 'D:\TEMP\CLASS.XLS'이(가) 성공적으로 생성되었습니다.
```

2.4 특수한 데이터 읽어오기

텍스트 형식의 데이터에서 숫자나 공백 없이 입력된 문자는 데이터 단계 프로그램에서 INFILE, INPUT 문장에 의해 SAS 데이터 만드는데 문제가 없다.(2.2.3절 참고) 그러나 정형화 되지 않은 데이터를 읽어오려면 다소 어려움이 있다. 이런 문제들도 입력된 데이터를 엑셀로 보낸 후 셀 서식에서 적절한 작업을 거친 후 SAS 데이터 가져오기 메뉴에서 불러오면 문제가 해결된다. 훨씬 간편하고 시간 효율적인 방법이다.

그럼에도 불구하고 이 절에서는 정형화 되지 않은 데이터를 확장 편집기 프로그램을 이용하여 불러오는 방법을 설명하고자 한다.

2.4.1 예제 데이터

변수: 국가명, 국민소득(\$, 1960년), 날짜1, 날짜2, 시간

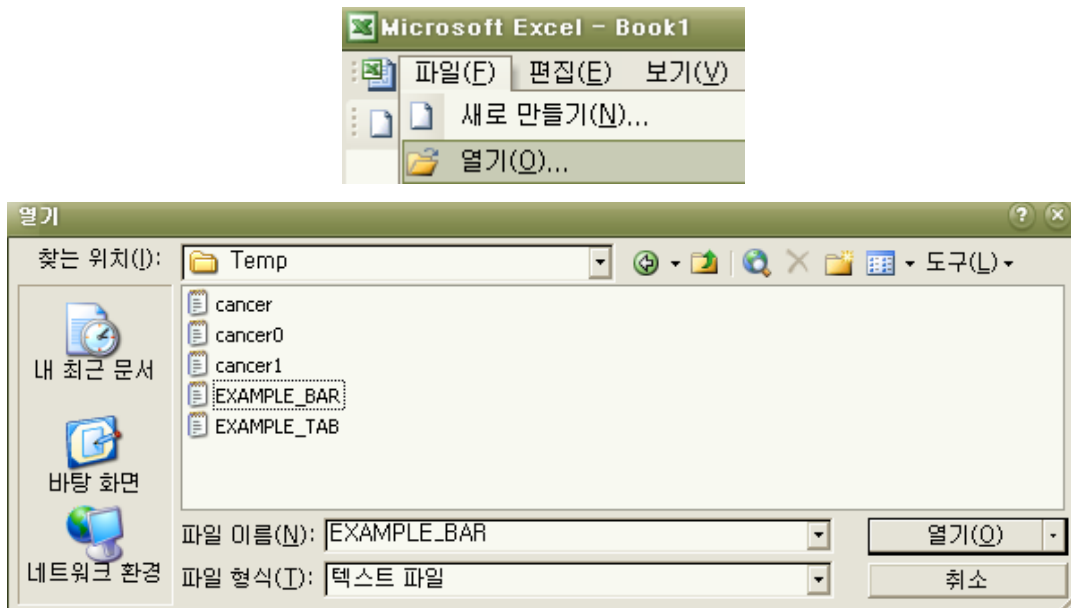
EXAMPLE_TAB .txt (tab 키 사용) / EXAMPLE_BAR .txt(space bar 사용)



정형화되지 않은 데이터를 SAS데이터로 만들 때는 우선 엑셀 파일로 변환한 후 SAS에서 2.2.2절 방법에 의해 SAS 데이터를 만드는 것이 편리하다.

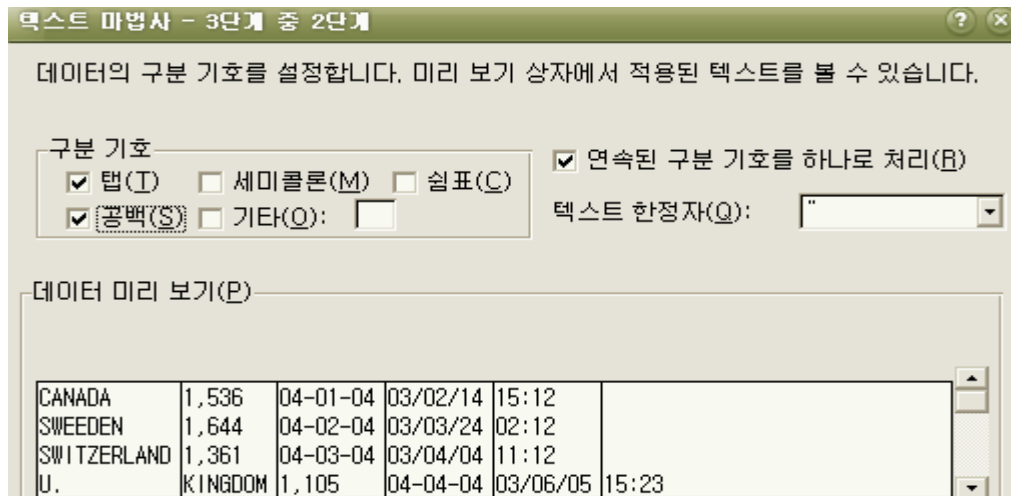
2.4.2 엑셀로 변환하기

EXAMPLE_BAR.txt를 엑셀 파일로 변환 해 보자. 두 텍스트 파일이 C:TEMP 폴더에 있다고 하자. 먼저 스페이스 바를 사용하여 입력한 데이터를 엑셀 데이터로 변환해 보자.

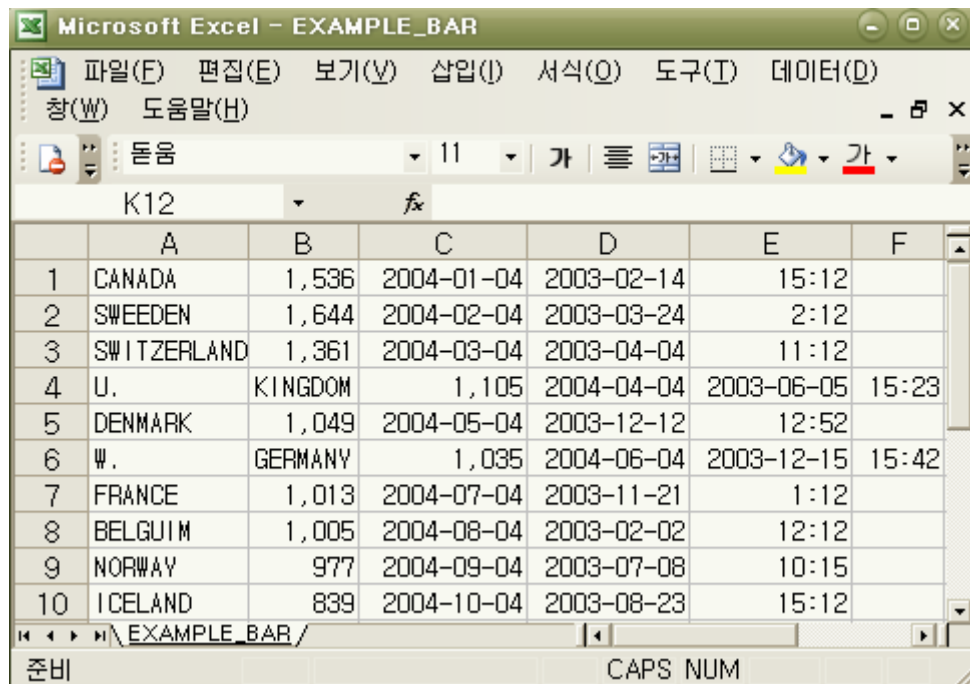


탭과 공백 부분을 표시하고 다음을 누른다. 그러나 문자에 공백이 있는 “U. KINGDOM”

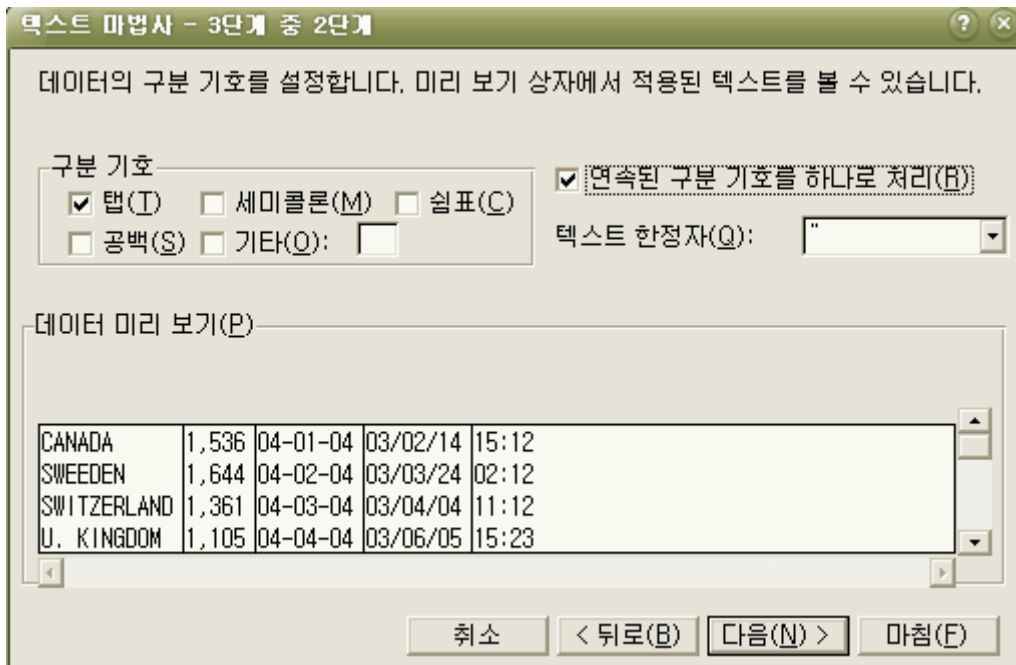
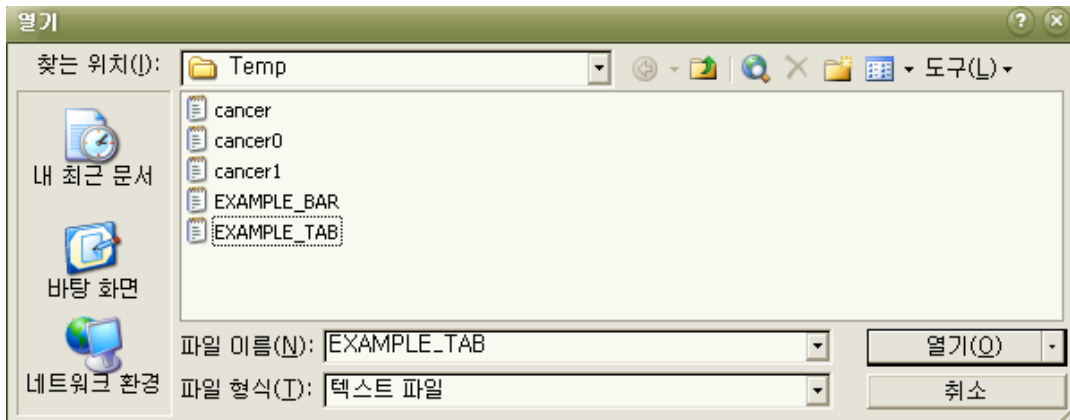
등은 문제이다. 해결 방법은 수작업 하는 방법 밖에 없다. 텍스트 마법사 - 3단계 중 1단계는 다음을 클릭한다.



여기서 마침을 클릭하면 다음과 같이 엑셀 데이터가 생성된다.



문자에 공백이 있는 경우에는 하나씩 수정해야 하는 문제가 발생한다. TAB 키를 이용하여 데이터를 입력한 EXAMPLE_TAB.txt를 엑셀 파일로 변환해 보자. 텍스트 마법사 - 3단계 중 1단계는 다음을 클릭한다.



여기서 마침을 클릭하면 다음과 같은 엑셀데이터가 생성된다.

Microsoft Excel - EXAMPLE_TAB

파일(F) 편집(E) 보기(V) 삽입(I) 서식(O) 도구(T)
 데이터(D) 창(W) 도움말(H)

동음 11

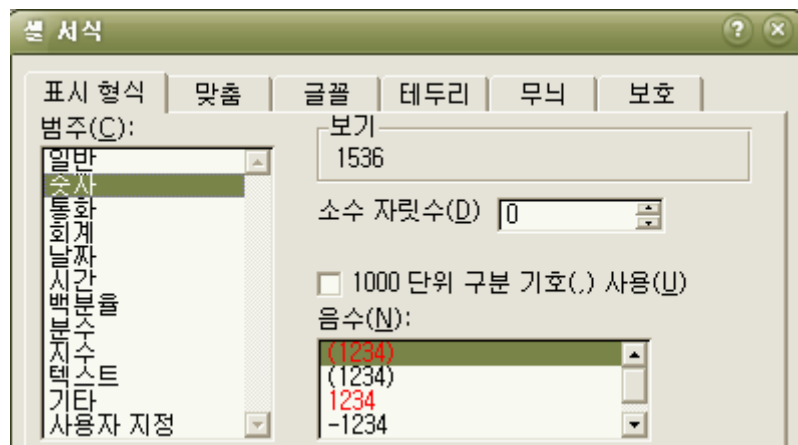
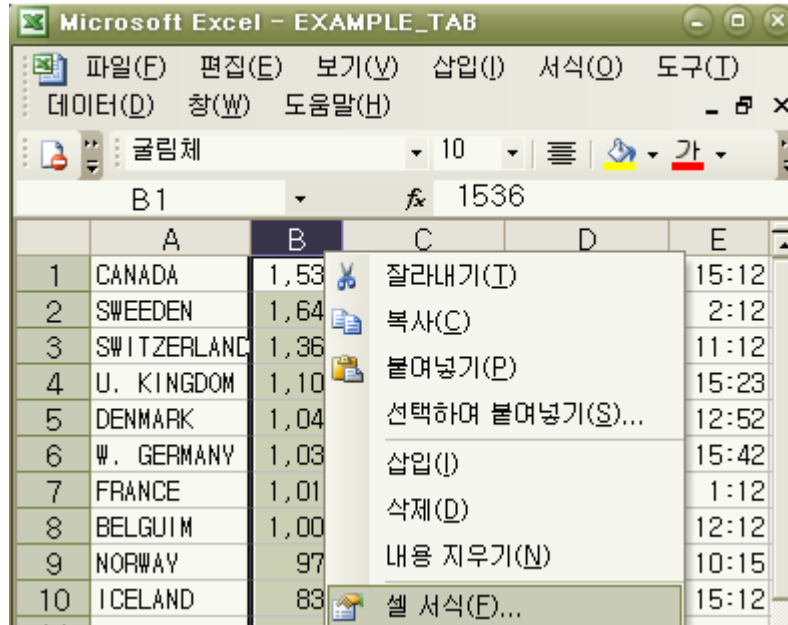
H4 fx

	A	B	C	D	E
1	CANADA	1,536	2004-01-04	2003-02-14	15:12
2	SWEEDEN	1,644	2004-02-04	2003-03-24	2:12
3	SWITZERLAND	1,361	2004-03-04	2003-04-04	11:12
4	U. KINGDOM	1,105	2004-04-04	2003-06-05	15:23
5	DENMARK	1,049	2004-05-04	2003-12-12	12:52
6	W. GERMANY	1,035	2004-06-04	2003-12-15	15:42
7	FRANCE	1,013	2004-07-04	2003-11-21	1:12
8	BELGUIM	1,005	2004-08-04	2003-02-02	12:12
9	NORWAY	977	2004-09-04	2003-07-08	10:15
10	ICELAND	839	2004-10-04	2003-08-23	15:12

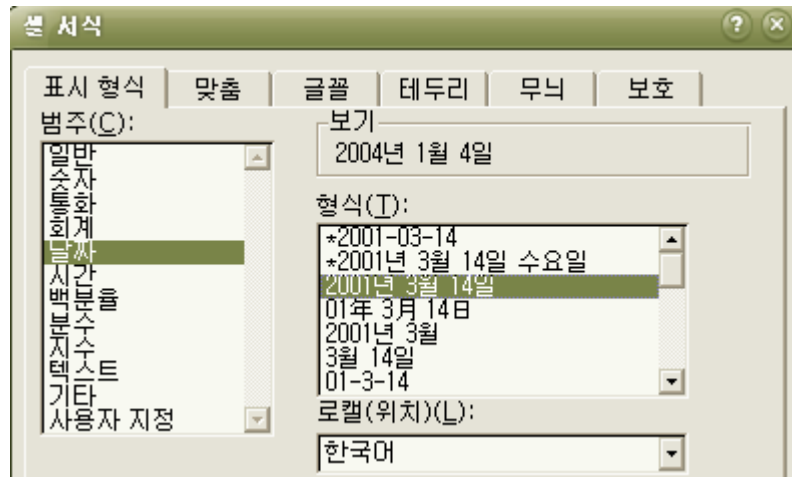
EXAMPLE_TAB /

준 CAPS NUM

성공적으로 변환되었다. 이제 변환된 엑셀 데이터의 셀 서식을 적절하게 바꾸자. 바꾸고 싶은 열 이름을 클릭하고 오른쪽 마우스 버튼을 클릭하면 팝업 창이 드는데 여기서 셀 서식 메뉴를 선택한다.



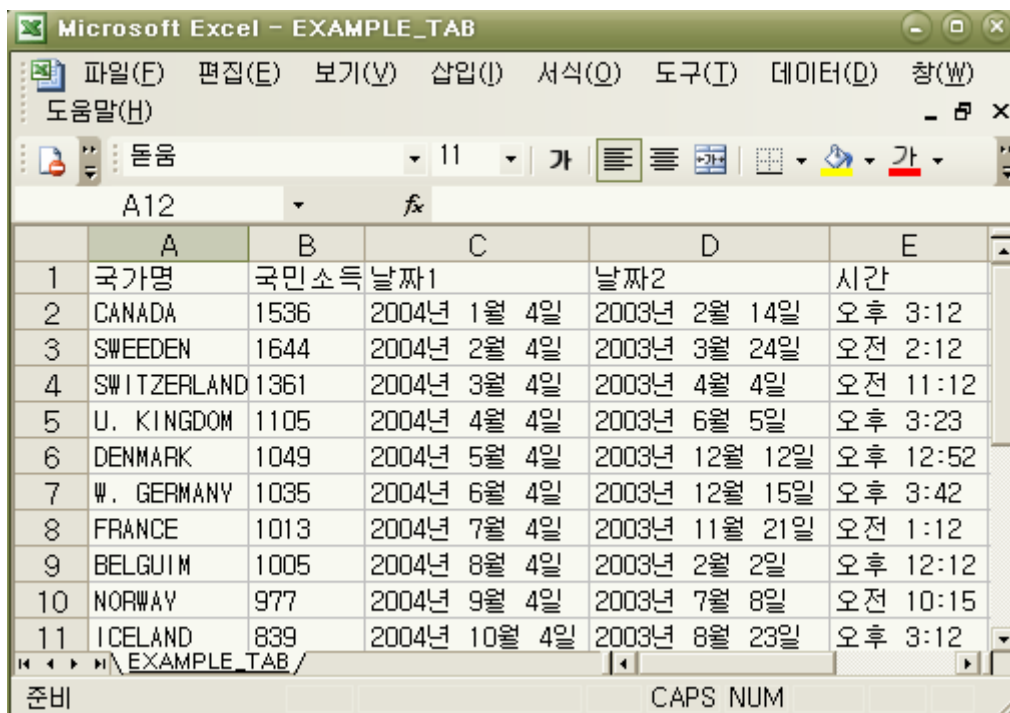
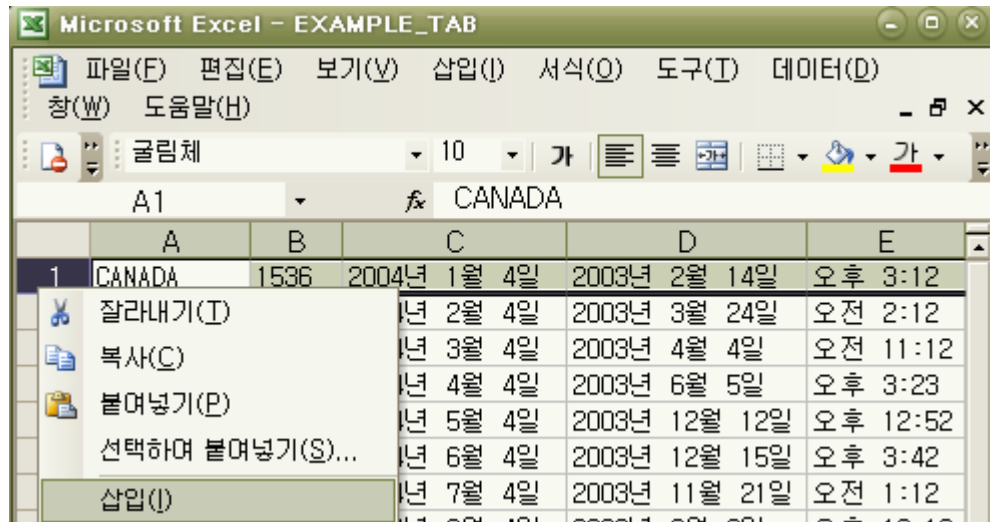
확인을 누르면 셀 서식이 바뀐다. 날짜도 동일한 방법으로 셀 서식을 지정한다.



동일한 방법으로 D열과 E열(시간에서 한다)을 다음과 같이 바꾸면 된다.

	A	B	C	D	E
1	CANADA	1536	2004년 1월 4일	2003년 2월 14일	오후 3:12
2	SWEEDEN	1644	2004년 2월 4일	2003년 3월 24일	오전 2:12
3	SWITZERLAND	1361	2004년 3월 4일	2003년 4월 4일	오전 11:12
4	U. KINGDOM	1105	2004년 4월 4일	2003년 6월 5일	오후 3:23
5	DENMARK	1049	2004년 5월 4일	2003년 12월 12일	오후 12:52
6	W. GERMANY	1035	2004년 6월 4일	2003년 12월 15일	오후 3:42
7	FRANCE	1013	2004년 7월 4일	2003년 11월 21일	오전 1:12
8	BELGUIM	1005	2004년 8월 4일	2003년 2월 2일	오후 12:12
9	NORWAY	977	2004년 9월 4일	2003년 7월 8일	오전 10:15
10	ICELAND	839	2004년 10월 4일	2003년 8월 23일	오후 3:12

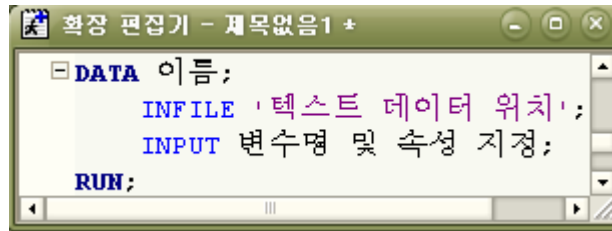
첫 행의 번호를 선택하고 오른쪽 마우스 버튼을 눌러 팝업 창에서 삽입을 선택하면 첫 행이 한 칸 아래로 밀리고 행이 삽입 된다. 다음과 같이 각 셀에 적절한 이름을 적는다.



2.4.3 확장 편집기 프로그램

이제 정형화 되지 않는 외부 데이터를 확장 편집기의 프로그램으로 한 번에 불러들이는

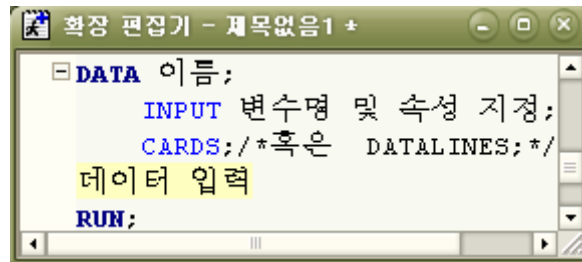
방법을 설명해 보자. 외부 텍스트(ASCII) 파일을 SAS 데이터 만들기 프로그램 구조(syntax)는 다음과 같다.



```

확장 편집기 - 제목없음1 *
DATA 이름;
  INFILE '텍스트 데이터 위치';
  INPUT 변수명 및 속성 지정;
RUN;
  
```

확장 편집기 내에서 데이터 입력하기 프로그램 구조는 다음과 같다. 외부 데이터가 텍스트 데이터의 경우 INFILE 문을 사용하기보다는 데이터를 확장 편집기에 복사한 후 CARDS; 문이나 DATALINES; 문을 사용하여 SAS 데이터를 만드는 것이 편리하다. 그러나 데이터가 많아지면(변수 10개 이상, 행 관측치 20~30개 이상) 프로그램 작성 시 입력 반응 속도가 늦어지는 단점이 있다.

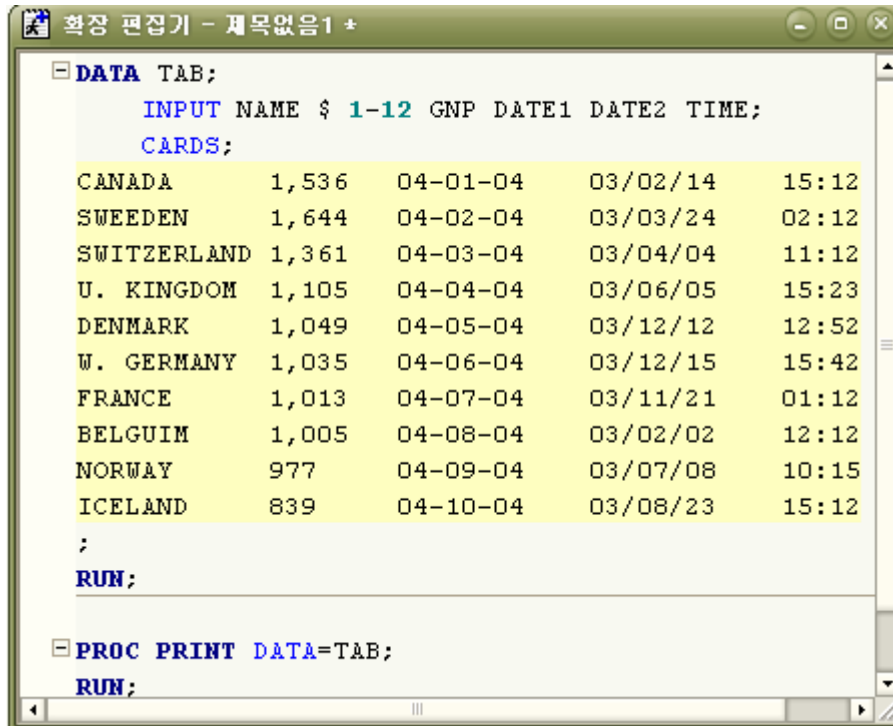


```

확장 편집기 - 제목없음1 *
DATA 이름;
  INPUT 변수명 및 속성 지정;
  CARDS; /*혹은 DATALINES;*/
  데이터 입력
RUN;
  
```

INPUT 문장에서 읽어 들이는 변수의 형식을 지정하는 것은 INFORMAT이라고 하고 출력 형식을 지정하는 것을 FORMAT 이라 한다.

공백이 있는 문자열 데이터



```

DATA TAB;
  INPUT NAME $ 1-12 GNP DATE1 DATE2 TIME;
  CARDS;
  CANADA      1,536   04-01-04   03/02/14   15:12
  SWEEDEN    1,644   04-02-04   03/03/24   02:12
  SWITZERLAND 1,361   04-03-04   03/04/04   11:12
  U. KINGDOM 1,105   04-04-04   03/06/05   15:23
  DENMARK    1,049   04-05-04   03/12/12   12:52
  W. GERMANY 1,035   04-06-04   03/12/15   15:42
  FRANCE     1,013   04-07-04   03/11/21   01:12
  BELGUIM   1,005   04-08-04   03/02/02   12:12
  NORWAY     977    04-09-04   03/07/08   10:15
  ICELAND    839    04-10-04   03/08/23   15:12
  ;
RUN;

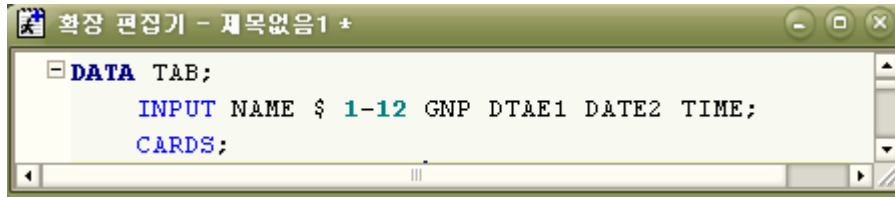
PROC PRINT DATA=TAB;
RUN;

```

위 프로그램을 실행시키면 다음과 같은 결과가 출력되는데 데이터가 제대로 읽히지 않음을 알 수 있다.

Obs	name	gnp	date1	date2	time
1	CANADA
2	SWEEDEN
3	SWITZERL
4	U.
5	DENMARK
6	W.
7	FRANCE
8	BELGUIM
9	NORWAY	977	.	.	.
10	ICELAND	839	.	.	.

1월부터 12월까지를 변수 NAME의 관측치로 읽는다. 이런 방식을 칼럼(column) 지정 방식이라 한다.



```

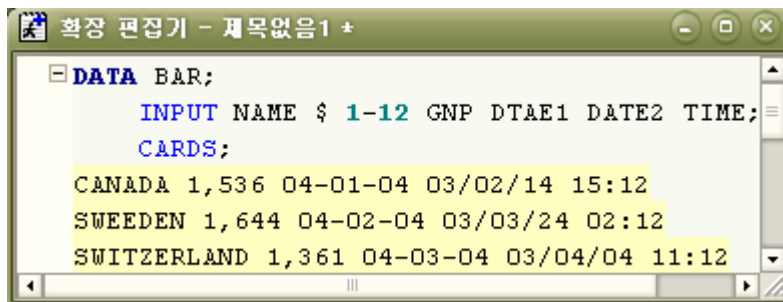
DATA TAB;
  INPUT NAME $ 1-12 GNP DTAE1 DATE2 TIME;
  CARDS;

```

INPUT 부분을 위와 같이 수정하고 실행시키면 결과는 다음과 같다.

Obs	name	gnp	date1	date2	time
1	CANADA
2	SWEEDEN
3	SWITZERLAND
4	U. KINGDOM

문자열에 공백이 있는 경우 열이 맞지 않으면 칼럼 지정 방식에 의해서도 불가능하다. 다른 방법이 있기는 하지만 여기서는 다루지 않겠다.



```

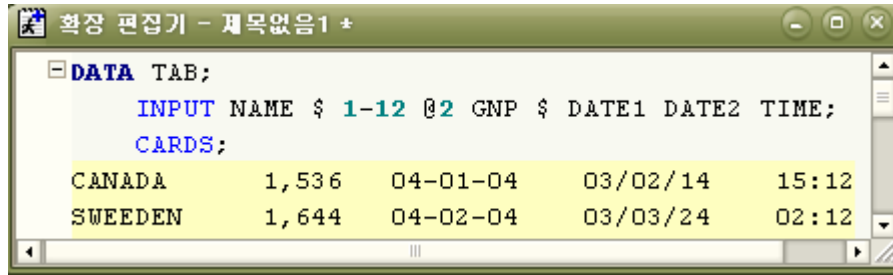
DATA BAR;
  INPUT NAME $ 1-12 GNP DTAE1 DATE2 TIME;
  CARDS;
  CANADA 1,536 04-01-04 03/02/14 15:12
  SWEEDEN 1,644 04-02-04 03/03/24 02:12
  SWITZERLAND 1,361 04-03-04 03/04/04 11:12

```

위 프로그램은 스페이스 바를 이용하여 입력한 자료를 읽는 프로그램이다.

Obs	name	gnp	date1	date2	time
1	CANADA 1,536
2	SWITZERLAND
3	U. KINGDOM 1
4	DENMARK 1,04	9	.	.	.

[참고] @ 사용하기. 커서를 강제로 2번째 열로 이동하여 관측치를 읽기 시작한다.



```

DATA TAB;
  INPUT NAME $ 1-12 @2 GNP $ DATE1 DATE2 TIME;
  CARDS;
  CANADA 1,536 04-01-04 03/02/14 15:12
  SWEEDEN 1,644 04-02-04 03/03/24 02:12

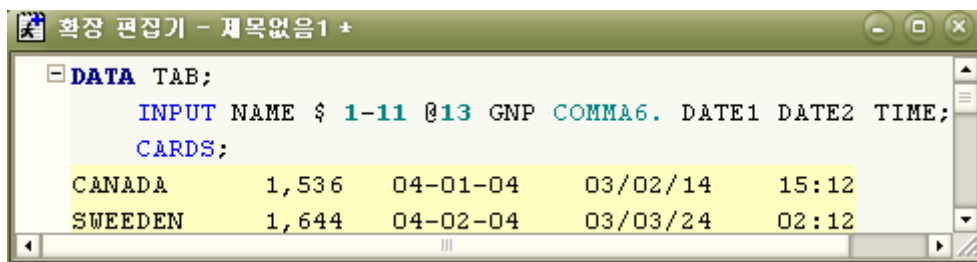
```

Obs	name	gnp	date1	date2	time
1	CANADA	ANADA	.	.	.
2	SWEEDEN	WEEDEN	.	.	.
3	SWITZERLAND	WITZERLA	.	.	.
4	U. KINGDOM
5	DENMARK	ENMARK	.	.	.
6	W. GERMANY
7	FRANCE	RANCE	.	.	.
8	BELGIUM	ELGIUM	.	.	.
9	NORWAY	ORWAY	977	.	.
10	ICELAND	CELAND	839	.	.

@를 사용해도 BAR 데이터는 문제를 해결할 수 없다.

coma(,) 데이터 읽기

COMMA가 있는 숫자 데이터를 읽을 때는 COMMAw.d 옵션을 사용한다. w(width)는 데이터 길이를 의미하며 d(digit)는 소수점 자리 수를 지정한다. 다음은 13번째 열부터 GNP 변수 관측치를 읽어 들이되逗마가 있는 데이터 6자리(소수점은 없음)로 데이터를 읽는 프로그램이다.



```

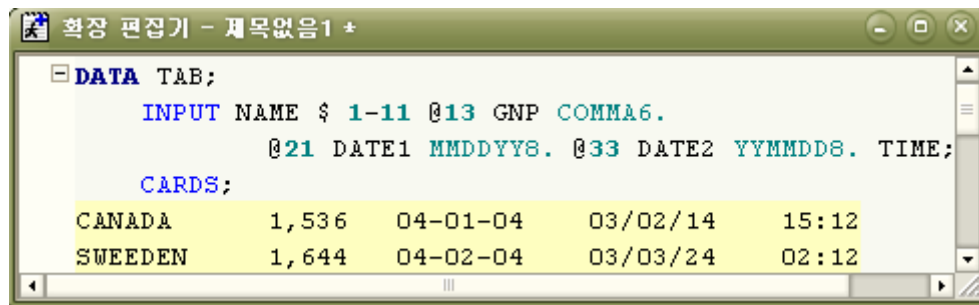
DATA TAB;
  INPUT NAME $ 1-11 @13 GNP COMMA6. DATE1 DATE2 TIME;
  CARDS;
  CANADA 1,536 04-01-04 03/02/14 15:12
  SWEEDEN 1,644 04-02-04 03/03/24 02:12

```

Obs	name	gnp	date1	date2	time
1	CANADA	1536	.	.	.
2	SWEEDEN	1644	.	.	.
3	SWITZERLAND	1361	.	.	.
4	U. KINGDOM	1105	.	.	.
5	DENMARK	1049	.	.	.
6	W. GERMANY	1035	.	.	.
7	FRANCE	1013	.	.	.
8	BELGIUM	1005	.	.	.
9	NORWAY	977	.	.	.
10	ICELAND	839	.	.	.

날짜, 시간 데이터 읽기

날짜 데이터를 지정하는 옵션은 다음과 같다. MMDDYYw. (MM은 월 두 자리, DD는 요일 두 자리, YY는 년도 두 자리를 의미한다. w는 데이터 길이를 의미한다. 8.을 사용한 것은 -나 /을 포함한 데이터 길이이다.



```

DATA TAB;
  INPUT NAME $ 1-11 @13 GNP COMMA6.
        @21 DATE1 MMDDYY8. @33 DATE2 YYMMDD8. TIME;
  CARDS;
CANADA 1,536 04-01-04 03/02/14 15:12
SWEEDEN 1,644 04-02-04 03/03/24 02:12

```

Obs	NAME	GNP	DATE1	DATE2	TIME
1	CANADA	1536	16162	15750	.
2	SWEEDEN	1644	16163	15788	.
3	SWITZERLAND	1361	16164	15799	.
4	U. KINGDOM	1105	16165	15861	.
5	DENMARK	1049	16166	16051	.
6	W. GERMANY	1035	16167	16054	.
7	FRANCE	1013	16168	16030	.
8	BELGIUM	1005	16169	15738	.
9	NORWAY	977	16170	15894	.
10	ICELAND	839	16171	15940	.

date1이나 date2 출력 결과를 보면 날짜 형식이 아니라 정수로 되어 있음에 놀랄 것이다. 우선 정수의 의미부터 설명해 보자. 1960년 1월 1일을 0으로 하여 매일 1씩 증가시켰다. 만약 출력 결과도 날짜로 보이려면 FORMAT을 지정해주면 된다. 10이나 8과 같은 숫자는 날

짜 길이를 지정하는 것이다. s는 slash의 약어이고 d는 dash를 의미한다.

2.4.4 형식 맞게 출력하기

만들어진 SAS 데이터를 읽어 들인 형태대로 출력하거나 원하는 형식으로 출력하려면 FORMAT 문을 사용하면 된다. 다음은 날짜 형식으로 읽어 들인 변수 DATE1, DATE2를 원하는 형식으로 출력하는 것이다.

```

DATA TAB;
  FORMAT DATE1 MMDDYY8. DATE2 YYMMDD8.;
  INPUT NAME $ 1-11 @13 GNP COMMA6.
         @21 DATE1 MMDDYY8. @33 DATE2 YYMMDD8. TIME;
CARDS;
CANADA      1,536    04-01-04    03/02/14    15:12
SWEEDEN     1,644    04-02-04    03/03/24    02:12
  
```

Obs	DATE1	DATE2	NAME	GNP	TIME
1	04/01/04	03-02-14	CANADA	1536	.
2	04/02/04	03-03-24	SWEEDEN	1644	.
3	04/03/04	03-04-04	SWITZERLAND	1361	.
4	04/04/04	03-06-05	U. KINGDOM	1105	.
5	04/05/04	03-12-12	DENMARK	1049	.
6	04/06/04	03-12-15	W. GERMANY	1035	.
7	04/07/04	03-11-21	FRANCE	1013	.
8	04/08/04	03-02-02	BELGIUM	1005	.
9	04/09/04	03-07-08	NORWAY	977	.
10	04/10/04	03-08-23	ICELAND	839	.

FORMAT 문은 숫자나 문자열에도 사용할 수 있다. FORMAT 문은 출력에 관련된 옵션이 다. GNP 변수는 8자리(소수점 2자리 포함) 콤마로 출력된다. NAME은 문자열 2자리이다. 출력은 이렇데 되더라도 SAS 데이터 내에는 온전한 데이터가 저장되어 있다.

```

DATA TAB;
  FORMAT DATE1 MMDDYY8. DATE2 YYMMDD8.;
  FORMAT GNP COMMA8.2 NAME $2.;
  INPUT NAME $ 1-11 @13 GNP COMMA6.
         @21 DATE1 MMDDYY8. @33 DATE2 YYMMDD8. TIME;
  CARDS;
CANADA      1,536    04-01-04    03/02/14    15:12
SWEEDEN     1,644    04-02-04    03/03/24    02:12

```

Obs	DATE1	DATE2	GNP	NAME	TIME
1	04/01/04	03-02-14	1,536.00	CA	.
2	04/02/04	03-03-24	1,644.00	SW	.
3	04/03/04	03-04-04	1,361.00	SW	.
4	04/04/04	03-06-05	1,105.00	U.	.
5	04/05/04	03-12-12	1,049.00	DE	.
6	04/06/04	03-12-15	1,035.00	W.	.
7	04/07/04	03-11-21	1,013.00	FR	.
8	04/08/04	03-02-02	1,005.00	BE	.
9	04/09/04	03-07-08	977.00	NO	.
10	04/10/04	03-08-23	839.00	IC	.

시간은 TIMEw.d 형식으로 읽으면 된다. 00:00:00은 시간:분:초로 00:00은 시간:분으로 인식하면 00:00:00을 d으로 하며 1초에 정수 하나씩 증가한다.

```

DATA TAB;
  FORMAT DATE1 MMDDYY8. DATE2 YYMMDD8.;
  FORMAT GNP COMMA8.2 NAME $2.;
  INPUT NAME $ 1-11 @13 GNP COMMA6.
         @21 DATE1 MMDDYY8. @33 DATE2 YYMMDD8.
         @45 TIME TIME6.;
  CARDS;
CANADA      1,536    04-01-04    03/02/14    15:12
SWEEDEN     1,644    04-02-04    03/03/24    02:12

```

Obs	DATE1	DATE2	GNP	NAME	TIME
1	04/01/04	03-02-14	1,536.00	CA	54720
2	04/02/04	03-03-24	1,644.00	SW	7920
3	04/03/04	03-04-04	1,361.00	SW	40320
4	04/04/04	03-06-05	1,105.00	U.	55380
5	04/05/04	03-12-12	1,049.00	DE	46320
6	04/06/04	03-12-15	1,035.00	W.	56520
7	04/07/04	03-11-21	1,013.00	FR	4320
8	04/08/04	03-02-02	1,005.00	BE	43920
9	04/09/04	03-07-08	977.00	NO	36900
10	04/10/04	03-08-23	839.00	IC	54720

출력도 시간 형태로 하려면 다음과 같이 하면 된다.

```

DATA TAB;
  FORMAT DATE1 MMDDYY8. DATE2 YYMMDD8.;
  FORMAT GNP COMMA8.2 NAME $2. TIME HHMM5.;
  INPUT NAME $ 1-11 @13 GNP COMMA6.
         @21 DATE1 MMDDYY8. @33 DATE2 YYMMDD8.
         @45 TIME TIME6.;
CARDS;
CANADA      1,536      04-01-04      03/02/14      15:12
SWEEDEN    1,644      04-02-04      03/03/24      02:12

```

Obs	DATE1	DATE2	GNP	NAME	TIME
1	04/01/04	03-02-14	1,536.00	CA	15:12
2	04/02/04	03-03-24	1,644.00	SW	2:12
3	04/03/04	03-04-04	1,361.00	SW	11:12
4	04/04/04	03-06-05	1,105.00	U.	15:23
5	04/05/04	03-12-12	1,049.00	DE	12:52
6	04/06/04	03-12-15	1,035.00	W.	15:42
7	04/07/04	03-11-21	1,013.00	FR	1:12
8	04/08/04	03-02-02	1,005.00	BE	12:12
9	04/09/04	03-07-08	977.00	NO	10:15
10	04/10/04	03-08-23	839.00	IC	15:12

2.4.5 두 행 이상 입력된 데이터 읽기

다음은 이름, 전화 번호, 월 사용료를 조사한 자료가 한 개체에 대해 2행으로 자료가 입력된 경우 SAS 데이터 만드는 프로그램이다. 여러 행인 경우 /을 계속하여 사용하면 된다.

```

DATA ONE;
  FORMAT BILL COMMA6.;
  INPUT NAME $ 1-9 PONE $
         11-23 /BILL COMMA6.;
  CARDS;
G. HONG    01X-123-4567
85,000
G. JANG    01X-234-5678
43,000
J. HWANG   01X-345-6789
84,000
;
RUN;

PROC PRINT DATA=ONE;
RUN;

```

SAS 시스템			
Obs	BILL	NAME	PONE
1	85,000	G. HONG	01X-123-4567
2	43,000	G. JANG	01X-234-5678
3	84,000	J. HWANG	01X-345-6789

한 편 한 행에 여러 개체의 데이터 입력된 경우에는 @@ 옵션을 사용하면 된다.

```

DATA TWO;
  INPUT GENDER $ IQ @@;
  CARDS;
M 120 F 115 M 110
F 130 F 115 M 128
;
RUN;

PROC PRINT DATA=TWO;
RUN;

```

SAS 시스템		
Obs	GENDER	IQ
1	M	120
2	F	115
3	M	110
4	F	130
5	F	115
6	M	128

2.4.6 연속하여 입력한 데이터 (설문 데이터)

설문 조사의 경우 데이터를 입력할 때는 항목(변수) 사이 빈칸 없이 연속하여 입력한다. 데이터 연속하여 입력하는 이유는 문자형 변수가 없고 데이터의 자리 수가 일정하므로 입

력 시간과 오류를 줄일 수 있기 때문이다. 예제 설문을 통하여 설문 데이터 입력 방법, SAS 데이터 만드는 방법을 알아보자.



설문 예제 데이터

1. 당신의 성별은? 남자() 여자 ()
2. 귀하의 전공은?

①통계학	②국제통상	③경영학	④경제학
⑤사회복지	⑥행정학	⑦교육학	⑧영문학
⑨ 철학	⑩사학	⑪문헌정보학	
3. 전공에 대한 만족도 어느 정도입니까?

①매우 불만족	②불만족	③보통	④만족	⑤매우 만족
---------	------	-----	-----	--------
4. 전공 선택 시 고려한 것은 무엇입니까? (복수 선택 가능)

①취업	②학문 우월성	③적성	④교수진
-----	---------	-----	------
5. 대학 생활 중 하고픈 일들에 대한 우선 순위를 부여하십시오. 1순위-5순위

①전공 공부()	②취업 준비()	③영어()	④학과 생활()	⑤동아리 활동()
-----------	-----------	--------	-----------	------------

설문지가 수거된 후 우선 부적절하고 불성실하게 응답된 설문지를 제외한다. 각 설문지에 일련 번호를 부여하고 첫 번째 설문지 문항에 변수 명을 적고 보기 번호가 없는 항목에 대해서는 숫자 번호를 부여한다.

각 문항에 변수 이름을 부여할 때 문항 내용에 따라 부여하는 것이 구별이 용이하다. Q1 대신 GENDER, Q2 대신 MAJOR 등으로... 그러나 설문 조사의 경우 문항 수가 많으므로 변수 이름 설정이 용이하지 않고 데이터 읽기, 프로그램 내에 사용하기 등에 문제가 있다. 이런 목적이라면 LABEL문을 사용하는 것이 적절하다. 그러므로 변수 이름은 차례로 Q1, Q2, ... 이런 식으로 부여하는 것이 적절하다. 4번 문항과 같이 복수 선택인 경우 가장 많이 선택한 응답자를 골라 그 수만큼 설정하거나 분석자가 임의로 2개로 지정하여 입력하면 된다. 5번 문항과 같이 우선 순위 문항은 순위 개수만큼 변수를 잡아 주어야 한다. Q5~QQ9

이렇게 잡아 주어도 되나 “_”에 의한 서브 번호를 사용하는 것이 문항 번호와 변수 번호가 일치하여 관리가 용이하다.

▶1번 문항: 보기 남자, 여자에 번호가 없으므로 남자=1, 여자=2 번호 부여

▶2번 문항: 보기가 11개 이므로 입력 자리 수는 2개이다. 그러므로 1번 선택한 경우에는 01로 입력해야 한다. 보기 개수가 10개이면 10번 보기를 0으로 입력하면 한자리만 입력해도 된다.

▶4번 문항: 최대 선택한 사람이 2개를 선택하였다. 혹은 분석자가 2개 선택한 것 까지만 유효하다고 인정한 경우이다. 2개만 유효한데(2개까지만 선택) 3개를 선택한 경우는 결측치 처리한다. 응답하지 않거나 결측치 처리 문항은 “.”(마침표)으로 입력한다.

▶5번 문항: 순위 개수만큼 변수를 지정한다. 입력 데이터는 보기 옆의 우선 순위이다.

Q1	1. 당신의 성별은? 남자() 여자 (✓) (1) (2)	
Q2	2. 귀하의 전공은? ①통계학 ②국제통상 ③경영학 ④경제학 ⑤사회복지 ⑥행정학 ⑦교육학 ⑧영문학 ⑨철학 ⑩사학 ⑪문헌정보학	2자리 부여
Q3	3. 전공에 대한 만족도 어느 정도입니까? ①매우 불만족 ②불만족 ③보통 ④만족 ⑤매우 만족	
Q4_1	4. 전공 선택 시 고려한 것은 무엇입니까? (복수 선택 가능)	
Q4_2	①취업 ②학문 우월성 ③적성 ④교수진	
Q5_1	5. 대학 생활 중 하고픈 일들에 대한 우선 순위를 부여하십시오. 1순위-5순위	
Q5_5	①전공 공부(1) ②취업 준비(4) ③영어(2) ④학과 생활(3) ⑤동아리 활동(5)	

입력한 데이터 보기이다.

데이터 입력	내용 설명
20152.14235	4번 문항에서 하나만 선택했으므로 나머지 하나는 결측치
11131221345	모든 문항 제대로 응답된 경우
2..24154321	2번 문항 무응답, 두 자리 입력 문항이므로 점을 두 번 입력
209.2..12345	4번 문항을 3개 응답하여 결측치 처리하였음
10142.....	우선 순위 문항을 제대로 응답하지 않은 경우

입력된 데이터가 C:\TEMP 폴더에 텍스트 형식으로 DATA.TXT로 저장되어 있다고 하자. SAS 데이터 만드는 프로그램은 다음과 같다. 4번, 5번과 같이 연속된 경우 반드시 ()로 묶어야 한다. 이런 형식은 개별 문항에 대해서도 사용할 수 있다. 6번 문항부터 10번 문항까지 한자리씩 입력하였다고 하자. 그런 경우 (Q6-Q10) (1.)을 사용하면 된다.

```

확장 편집기 - 제목없음1 *
DATA ONE;
  INFILE 'C:\TEMP\DATA.TXT';
  INPUT Q1 1. Q2 2. Q3 1. (Q4_1-Q4_2) (1.) (Q5_1-Q5_5) (1.);
RUN;

```

Q1, Q2, ... 변수가 무엇인지 구별하기 위해서는 LABEL 문을 사용하여 SAS 데이터를 만들면 PROC 단계의 결과에 Q1, Q2 옆에 이름이 출력된다.

```

확장 편집기 - 제목없음1 *
DATA ONE;
  INFILE 'C:\TEMP\DATA.TXT';
  INPUT Q1 1. Q2 2. Q3 1. (Q4_1-Q4_2) (1.) (Q5_1-Q5_5) (1.);
  LABEL Q1='성별' Q2='전공';
RUN;

```

각 문항의 보기에 대한 이름 지정은 FORMAT 문을 이용해야 한다. 이는 3.절에서 다루기로 한다.



시간/날짜 INFORMAT 및 FORMAT 문장 형식

INFORMAT명	의미	길이	표준 길이
DATEw.	ddmmyy 형식의 일자값(예 01 4 무59)	7-32]
DATETIMEw.	ddmmyy:hh:ss.ss 형식의 일자값	13-14	18
DDMMYYw.	ddmmyy 형식의 일자값	6-32	8
JULIANw.	yydd 형식의 유리무스력에 의한 일자값	5-32	5
MMDDYYw.	MMDDYY 형식의 일자값	6-32	8
MONYYw.	MMMyy 형식의 일자값	5-32	5
TIMEw.d	hh:mm:ss.ss 형식의 일자값	5-32	8
YYMMDDw.	yymmdd 형식의 일자값	6-32	8
YYQw.	yyQq 형식의 연도와 분기를 표시	4-32	4
FORMAT 명	의미	길이	표준 길이
DATAw.	ddmmmyy 형식의 일자값(예 01jan59)	5-9	7
DATETIMEw.	ddMMMyy:hh:ss.ss 형식의 날짜치	7-40	16
DDMMYYw.	ddMMyy 형식의 날짜치	2-8	8
HHWWw.d	hh:mm 형식의 날짜치	2-20	5
HOURw.d	hh.h 형식의 시간치 시각부분	2-20	2
JULIANw.	yydd 형식의 유리무스력에 의한 날짜치	5-7	5
MMDDYYw.	mmddy 형식의 날짜치	2-8	8
MMSSw.d	mm:ss 형식의 시간치	2-20	5
MONYYw.	MMMyy 형식의 날짜치	5-7	5
TIMEw.d	hh:mm:ss.ss 형식의 시간치	2-20	8
TODw.	hh:mm:ss 형식에 의한 일시치의 시간부분	2-20	8
WEEKDATEw.	날짜치 (요일-월-일-연 영어로 표기)	3-37	29
WORDDATEw.	날짜치 (월-일-연 형식 영어로 표기)	3-32	18
YYMMDDw.	yymmdd 형식의 날짜치	2-10	8
YYQw.	yyQq 형식의 연과 사분기의 날짜치	4-6	4