Chapter 6. 데이터셋의 결합 및 관리

세로결합

❖ SET 명령문

- 이미 존재하고 있는 SAS 데이터 셋을 지정하여 그 데이터 셋의 전체 혹은 일부분을 읽거나
- 두 개 이상의 데이터 셋들을 세로로 결합하는데 사용

```
DATA new-data-set;
SET data-set-1 data-set-2 … ;
```

- 새로운 데이터 셋의 이름을 기존의 데이터 셋 이름과 동일한 것으로 지정하면 기존의 데이터 셋은 새로운 것으로 대체

세로결합

❖ 세로 결합(Concatenating)

- 여러 개의 데이터 셋을 동시에 지정하면, 지정된 데이터 셋의 각 개체들을 하나씩 차례로 읽어서 순서대로 쌓아 세로로 결합한 새로운 데이터 셋을 생성

SET 명령문을 이용한 데이터셋 생성

```
DATA mysas.male;
INPUT name $ sex $ mid final pre;
CARDS;
김철수 M 10 40 30
강민호 M 50 15 45
;
RUN;
```

```
DATA mysas.female;
INPUT name $ sex $ mid final;
CARDS;
이영희 F 15 10
박지수 F 20 .
;
RUN;
```

```
DATA mysas.concat;
SET mysas.male mysas.female;
IF final=. THEN final=mid;
RUN;
```

가로결합

❖ MERGE 명령문

- 여러 개의 데이터 셋을 동시에 지정하면 데이터 셋을 가로로 결합하여 새로운 데이터 셋을 생성
- 유의할 점
  - 여러 데이터 셋에 동일한 변수 이름이 존재하면 그 공통변수에 대한 개체들의 자료값은 MERGE 명령문의 뒤쪽에 지정된 데이터 셋의 자료값을 기준으로 대체

MERGE 명령문을 이용한 데이터셋 생성

```
DATA mysas.one;
INPUT a b c@@;
CARDS;
11 21 31 12 22 32 13 23 33 14 24 34 15 25 35
16 26 36
;
RUN;
```

```
DATA mysas.two;
INPUT c d@@;
CARDS;
41 51 42 52 . 53 44 54
;
RUN;
```

```
DATA mysas.combine;
MERGE mysas.one mysas.two;
RUN;
```

가로결합

- 대응 가로결합
  - BY 명령문
    - 명령문에 지정된 변수에 의해서 두 개 이상의 데이터 셋들이 결합
    - BY 명령문에 지정된 변수에 의해서 미리 정렬되어 있어야 함

MERGE와 BY 명령문을 이용한 대응 가로결합

```
DATA mysas.mid;
INPUT name $ sex $ pre mid;
CARDS;
김철수 M 30 10
강민호 M 45 50
이영희 F . 15
박지수 F . 20
;
RUN;
```

```
DATA mysas.final;
INPUT name $ sex $ pre final;
CARDS;
이영희 F 32 10
김철수 M . 40
박지수 F 15 20
강민호 M . 15
;
RUN;
```

```
PROC SORT DATA=mysas.mid;
BY name;RUN;
PROC SORT DATA=mysas.mid;
BY name;RUN;
DATA mysas.all;
MERGE mysas.mid mysas.final;
BY name;RUN;
```

가로결합

❖ 데이터의 갱신

- UPDATE 명령문
  - 나중에 지정된 데이터 셋에 결측값이 있는 경우에는 먼저 지정된 데이터 셋의 값을 그대로 유지할 필요가 있을 때 사용
  - 갱신에 앞서 두 데이터 셋이 기준이 되는 BY 변수에 의해 미리 정렬되어 있어야 함

UPDATE 명령문을 이용한 데이터 갱신

```
DATA mysas.all1;
UPDATE mysas.mid mysas.final;
BY name;
RUN;
```

데이터 셋 옵션의 사용

❖ 데이터 셋 옵션

- KEEP = variables: 데이터 셋에 포함될 변수들을 지정
- DROP = variables: 데이터 셋에 포함되지 않을 변수들을 지정
- RENAME = (oldvar=newvar): 변수 이름을 oldvar에서 newvar로 바꿈
- FIRSTOBS = n: n 번째 개체에서부터 데이터를 읽도록 지정
- OBS = n: n 번째 개체까지만 데이터를 읽도록 지정

데이터 셋 옵션의 사용

```
DATA mysas.all2;
UPDATE mysas.mid(KEEP=name mid pre OBS=4) mysas.final(DROP=sex RENAME=(pre=pre1));
BY name;
RUN;
```

```
DATA mysas.all3;
UPDATE mysas.mid mysas.final;
BY name;
RENAME mid=m_score final=f_score;
DROP sex pre;
RUN;
PROC PRINT DATA=mysas.all(FIRSTOBS=2 OBS=3);
RUN;
```

DO-END와 OUTPUT 명령문

- DO-END 명령문
  - DO와 이에 짝을 이루는 END 명령문 사이에 있는 내용을 반복적으로 수행
  - UNTIL이나 WHILE 명령어를 사용하여 DO 명령문이 수행될 조건을 지정할 수 있음
  - DO-END 명령문은 반복작업을 계속 수행하게 할 뿐 매 반복에서 작업한 내용을 생성되는 데이터 셋에 저장하지 않음
- OUTPUT 명령문
  - DO-END 명령문의 반복작업에 의해 발생한 결과를 생성되는 데이터 셋에 저장
  - OUTPUT 명령문을 사용하지 않으면 맨 마지막에 수행된 하나의 개체만 데이터 셋에 저장
  - 데이터 셋 이름을 함께 지정하면 각 개체를 각각 특정 데이터 셋에 저장할 수 있음

DO-END와 OUTPUT 명령문

DO-END와 OUTPUT 명령문의 사용

```
DATA mysas.survey;
INFILE “C:\SAS_Programming\Raw_Data\survey.txt”;
DO age=10 TO 50 by 10;
DO sex=‘Female’, ‘Male’;
INPUT size response@@;
OUTPUT;
END;END;
RUN;
```

```
DATA mysas.gender_m mysas.gender_f;
SET mysas.survey;
IF sex=‘Male’ THEN OUTPUT mysas.gender_m;
ELSE OUTPUT mysas.gender_f;
 RUN;
```

자동변수(Automatic Variable)

- 자동변수
  - 새로운 데이터 셋을 생성할 때, 작업에 필요한 여러 가지 정보
  - 자동변수는 개체를 읽어 들이는 동안에만 임시로 존재하며 생성된 데이터 셋에는 따로 저장되지 않음
  - 저장할 경우 새로운 변수 이름을 지정하여 저장
- _N_과 _ERROR_
  - _N_: 각 개체에게 그것이 만들어진 순서에 따라 1부터 시작하는 자연수의 값이 부여
  - _ERROR_: 오류 없이 수행된 경우에는 해당 개체에 0, 오류가 발생한 경우 1을 변수값으로 가짐

자동변수 **_N_**과 **_ERROR_**의 사용

```
DATA mysas.stat;
INPUT dept $ id $ score grade $@@;
IF _N_=1 THEN score=17;
obsnum=_N_;
Errornum=_ERROR_;
CARDS;
STAT S01 15 C STAT S02 40 A STAT S03 35 B
MATH M01 32 B MATH M02 54 A MATH M03 C C MATH M04 63 A
;
RUN;
```

자동변수(Automatic Variable)

- FIRST.BY와 LAST.BY 변수
  - 각 개체그룹의 시작과 끝을 나타내는 정보를 저장하고 있는 변수
  - 개체그룹
    - BY 변수에 의해 크기 순으로 정렬되어 있는 경우 일반적으로 동일한 BY 변수의 값을 가지는 여러 개체 개체들의 집합
  - 개체 그룹의 첫 번째 개체에는 FIRST.BY 변수가 1
  - 개체 그룹의 마지막 개체에는 LAST.BY 변수가 1

자동변수 FIRST.BY와 LAST.BY의 사용

```
PROC SORT DATA=mysas.stat;
BY dept score;
RUN;
DATA mysas.stat_n;
SET mysas.stat;
BY dept;
IF FIRST.dept=1 OR LAST.dept=1;
RUN;
```

텍스트 데이터 출력하기

❖ PUT과 FILE 명령문

- PUT 명령문
  - 데이터 셋의 내용을 텍스트 파일로 바꾸어 외부로 보내고자 할 때 출력하는 형식을 지정하는 명령문
  - INPUT 명령문에서 사용할 수 있는 다양한 포맷형식을 사용할 수 있음
  - 문자열을 출력하고자 할 때 인용부호로 문자열을 표시해주면 됨
- FILE 명령문: 데이터를 출력할 외부 파일을 지정하는 명령문
- FILENAME 명령문: 파일에 대한 별칭을 등록하기 위해 사용

텍스트 파일로 데이터 출력하기

```
DATA mysas.subject;
INPUT name $ dept $ math stat eng kor art@@;
CARDS;
김철수 Stat 5 5 1 2 1 최민지 Stat . 3 1 4 5 이영희 Stat 1 5 3 2 .
오인수 Stat 4 1 2 4 . 강민호 Econ 3 2 3 1 4
;
RUN;
FILENAME myfile “c:\SAS_Programming\Raw_Data\scoreout.txt”;
DATA _NULL_;
SET mysas.subject;
FILE myfile;
IF dept=‘Stat’;
PUT name 6. ‘,’ @11 math ‘,’ stat ‘,’ +3 eng ‘,’ kor ‘,’ art;
RUN;
```

EXPORT 프로시저 이용하기

❖ PROC EXPORT 명령문

- DATA=옵션: 내보낼 데이터 셋이 저장되어 있는 라이브러리와 SAS 데이터 셋 이름을 지정
- DBMS=옵션: 내보낼 파일의 형식을 지정
- REPLACE: 동일한 이름의 테이블이 있는 경우 이를 덮어쓰도록 지정
- OUTTABLE=옵션: 데이터가 내보내어질 테이블의 이름을 지정(ACCESS)
- DATABASE=명령어: 테이블이 저장될 MS ACCESS의 경로와 이름을 지정
- OUTFILE=옵션: 데이터 시트가 저장될 MS EXCEL 파일의 경로와 이름을 지정
- SHEET=명령문: 워크시트의 이름을 지정

EXPORT 프로시저(ACCESS)	EXPORT 프로시저(EXCEL)
PROC EXPORT DATA=mysas.htwt; OUTTABLE="htwtdata" DBMS=ACCESS REPLACE; DATABASE="C:\Sas_Programming\Raw_Data\export.mdb"; RUN;	PROC EXPORT DATA=mysas.subject DBMS=EXCEL REPLACE OUTFILE='C:\Sas_Programming\Raw_Data\subject.xls'; SHEET='subject'; RUN;

연습문제1

- 각 행은 병아리종류를, 각 열은 사료종류를 나타내는 다음의 데이터는 어린 병아리의 체중 증가량을 측정한 값을 기록한 것이다.

```
55 61 169 42
49 112 137 97
42 30 169 81
```

이 데이터를 읽어서 다음과 같은 내용을 가지는 SAS 데이터셋 chicken을 생성하기 위한 프로그램을 작성해 보아라. 단, 데이터셋 chicken에서 variety는 병아리종류를 saryo는 사료종류를 weight는 체중증가량을 나타내는 변수라고 하자.

Variety	Saryo	Weight
A	1	55
A	2	61
A	3	169
A	4	42
B	1	49
B	2	112
B	3	137
B	4	97
C	1	42
C	2	30
C	3	169
C	4	81

연습문제 2

- 두 개의 sas 데이터 셋 data1과 data2에 5개의 변수(id, gender, height, weight, year)가 포함되어 있고 두 데이터셋의 자료값들은 다음과 같다고 할 때,

  (1) 두 개의 데이터셋을 세로로 결합시킨 sas 데이터셋 total을 생성하여라.

  (2) 데이터셋 total로부터 변수 gender가 M 인 개체들만으로 새로운 데이터셋 male을 생성하여라.

Data1					Data2				
Id	Gender	Height	Weight	Year	Id	Gender	Height	Weight	Year
1	M	172	65	92	4	F	160	45	93
3	M	189	89	97	7	M	192	85	91
5	F	163	47	95	6	M	168	57	92
2	F	167	52	95	8	M	183	62	98

연습문제3

- 서로 다른 변수들을 가지고 있는 다음과 같은 두 개의 데이터 셋 infor와 score가 있을 때, 이 데이터 셋들을 기준변수 id에 의해 대응 가로결합하여 새로운 데이터 셋 combined를 생성하는 프로그램을 작성하시오.

Infor			**Score**			
Id	Gender	Class	Id	Dept	Mid	Final
1	M	NO	1	ENGL	30	50
5	F	NO	2	STAT	55	70
2	M	YES	3	ECON	62	90
3	F	NO	5	STAT	48	87