3-3강

포맷수정자: & 를 이용하여 관찰값 읽기

```
DATA club2;
INPUT indo name & $18. team $ stwgt endwgt;
CARDS;
1023 David Shaw  red 189 165          /*David Shaw  red에서 Shaw 와 red 사이의 공백이 2칸*/
1049 Amelia Serrano  yellow 145 124
;
RUN;

PROC PRINT DATA=club2;
RUN;
```

SAS 시스템

OBS	indo	name	team	stwgt	endwgt
1	1023	David Shaw	red	189	165
2	1049	Amelia Serrano	yellow	145	124

SAS 시스템

OBS	indo	name	team	stwgt	endwgt
1	1023	David Shaw red 189	1049	.	.

공백이 한 칸일 경우

예 : 한 한개의 공백문자만을 문자변수의 일부로 읽기

```
DATA one;
LENGTH lastname $ 15 name1 $8;
/*LENGTH는 자릿수의 default를 알려줌*/
INPUT lastname $ name1 $;
CARDS;
Longlastname1  John
Mc Allister  Mike
Longlastname3  Jim
;
RUN;
```

VIEWTABLE: Work.One

	lastname	name1
1	Longlastname1	John
2	Mc	Allister
3	Longlastname3	Jim

```
INPUT lastname $15. name1 $;
```

```
INPUT lastname $13. name1 $CHAR6.;
/*CHAR은 공백문자를 포함한다.*/
```

VIEWTABLE: Work.One2

	lastname	name1
1	Longlastname1	John
2	Mc Allister M	ike
3	Longlastname3	Jim

```
INPUT lastname & $15. name1 $;
```

VIEWTABLE: Work.One3

	lastname	name1
1	Longlastname	1 Joh
2	Mc Allister	Mike
3	Longlastname	3 Jim

VIEWTABLE: Work.One5

	lastname	name1
1	Longlastname1	John
2	Mc Allister	Mike
3	Longlastname3	Jim

자유 포맷 & 표준데이터 유형 (LIST INPUT)

데이터 유형	01 1 Male 1 1 1 02 2 Man 3 3 3 03 4 Female 3 3 1 04 4 Man 3 3 2 05 4 M 1 1 1 06 5 Female 2 . . 07 3 MR 1 1 1 08 5 Famme 1 1 1 09 5 Man 1 1 3 10 2 Female 2 3 2	1 1 Male 1 1 1 2 2 Man 3 3 3 3 4 Female 3 3 1 4 4 Man 3 3 2 5 4 M 1 1 1 6 5 Female 2 . . 7 3 MR 1 1 1 8 5 Famme 1 1 1 9 5 Man 1 1 3 10 2 Female 2 3 2	1,1,M,1,1,1 2,2,M,3,3,3 3,4,F,3,3,1 4,4,M,3,3,2 5,4,M,1,1,1 6,5,F,2,.,. 7,3,M,1,1,1 8,5,F,1,1,1 9,5,M,1,1,3 10,2,F,2,3,2	1,1,Male,1,1,1 2,2,Man,3,3,3 3,4,Female,3,3,1 4,4,Man,3,3,2 5,4,M,1,1,1 6,5,Female,2,.,. 7,3,MR,1,1,1 8,5,Famme,1,1,1 9,5,Man,1,1,3 10,2,Female,2,3,2
특징	**구분자(공백)로 분리됨**		**구분자(COMMA)로 분리됨**	
기타 특징	● **결측값은 . 으로 표시되어 있다.** ● **문자열은 공백을 포함하지 않고, 8자 이하이다.**			
문법	● 입력방법은 변수명 변수유형 ex) age gen $ ● 구분자 지정은 ex) INFILE fileref DLM=', ' ; (SAS SYSTEM Default Delimiter : 공백) ● **8자를 초과하는 문자데이터가 있는 변수에 대해서는 LENGTH 문장으로 미리 선언/지정 : LENGTH name $ 10 ;**			

- 구분자, delimeter가 기본으로는 공백( )이지만, 상황에 따라서는 콤마(,)이다.

```
INFILE file_directory DLM=','; /*DLM가 구분자를 지정*/
```

- 문자열은 기본적으로는 8자가 최대이지만, 이 이상을 넘어갈때는 LENGTH로 그 사실을 알려줘야 한다.

```
LENGTH name $ 10 ;
```

DATA 읽기_실습

- Column Input
  - 열 번호 지정
  - 자료값이 고정된 열을 가지고 있어야 함

```
DATA scores;
INPUT name $ 1-18 score1 25-27 score2 30-32 score3 35-37;
CARDS;
Joseph                  11   32   76
Mitchel                 13   29   82
Sue Ellen               14   27   74
;
RUN;
PROC PRINT DATA=scores;
RUN;
```

OBS	name	score1	score2	score3
1	Joseph	11	32	76
2	Mitchel	13	29	82
3	Sue Ellen	14	27	74

- Formatted Input

```
DATA discounts;
infile "d:\data\offers.txt";
INPUT
  @1 Cust_type 4.
  @5 Offe_dt mmddyy8.
  @14 Item_gp $8.
  @22 Discount percent3.
;
RUN;

PROC PRINT data=discounts;
RUN;
```

Description	Column
Customer Type	1-4
Offer Date(월일년순)	5-12
Item Group	14-21
Discount	22-24

OBS	Cust_type	Offe_dt	Item_gp	Discount
1	1014	17502	Outdoors	0.15
2	2020	17446	Golf	0.07
3	1030	17431	Shoes	0.10
4	1030	17431	Clothes	0.10
5	2020	17355	Clothes	0.15

- List Input
  - 데이터가 자유 포맷, 즉 하나 이상의 공백문자로 구분되었을 때 사용

```
DATA scores;
LENGTH name $ 12;
INPUT name $ score1 score2;
CARDS;
Riley 1132 1187
Henderson 1015 1102
;
RUN;

PROC PRINT data=scores;
RUN;
```

OBS	name	score1	score2
1	Riley	1132	1187
2	Henderson	1015	1102

Data 읽기

- 구분자
  - 기본 구분자는 공백임
  - 즉, 구분자를 기술하지 않을 경우 공백이 구분자로 인식됨
  - INFILE 문장의 DLM=옵션으로 구분자를 정의할 수 있음

```
INFILE "raw-data-file-name" dlm='구분자';
```

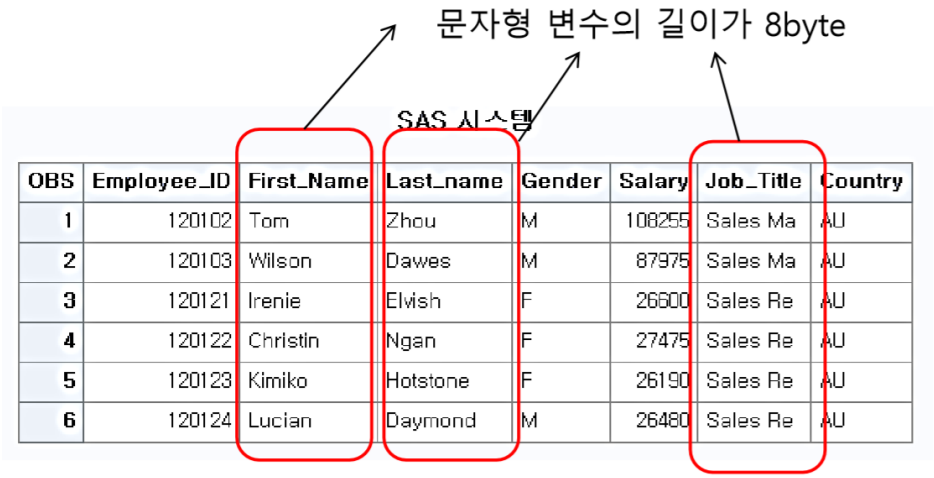
```
Data subset3;
Infile "c:\sas\sales.csv" dlm=',';
```

Data 읽기_실습

- List Input : 표준데이터 처리

```
DATA subset3;
infile 'd:\data\sales.txt' dlm=',';
INPUT Employee_ID First_Name $ Last_name $ Gender $ Salary
  Job_Title $ Country $;
RUN;

PROC PRINT data=subset3;
RUN;
```



OBS	Employee_ID	First_Name	Last_name	Gender	Salary	Job_Title	Country
1	120102	Tom	Zhou	M	108255	Sales Ma	AU
2	120103	Wilson	Dawes	M	87975	Sales Ma	AU
3	120121	Irenie	Elvish	F	26600	Sales Re	AU
4	120122	Christin	Ngan	F	27475	Sales Re	AU
5	120123	Kimiko	Hotstone	F	26190	Sales Re	AU
6	120124	Lucian	Daymond	M	26480	Sales Re	AU

→ LENGTH문이 선언이 안 되어 있다.

```
DATA subset;
infile 'F:\data\sales.txt' dlm=',';
LENGTH First_name $ 10;         /*LABEL을 선언한 변수가 가장 먼저 나오게 된다.*/
INPUT Employee_ID First_Name $ Last_name $ Gender $ Salary
  Job_Title $   Country $ ;
RUN;

DATA a;
RETAIN employee_ID First_name;  /*RETAIN은 변수의 위치를 고정시킨다.*/
set subset;
run;

PROC PRINT data=a;
RUN;
```

- RETAIN은 변수의 위치를 고정시킨다.
  - 첫번째가 employee_ID, 두번째가 First_name, 그 뒤에 오는 변수들은 그대로 둔다.
  - RETAIN 명령어는 문자형이라고 해도 $를 쓰지 않는다.

OBS	First_name	Employee_ID	Last_name	Gender	Salary	Job_Title	Country
1	Tom	120102	Zhou	M	108255	Sales Ma	AU
2	Wilson	120103	Dawes	M	87975	Sales Ma	AU
3	Irenie	120121	Elvish	F	26600	Sales Re	AU
4	Christina	120122	Ngan	F	27475	Sales Re	AU
5	Kimiko	120123	Hotstone	F	26190	Sales Re	AU
6	Lucian	120124	Daymond	M	26480	Sales Re	AU

OBS	employee_ID	First_name	Last_name	Gender	Salary	Job_Title	Country
1	120102	Tom	Zhou	M	108255	Sales Ma	AU
2	120103	Wilson	Dawes	M	87975	Sales Ma	AU
3	120121	Irenie	Elvish	F	26600	Sales Re	AU
4	120122	Christina	Ngan	F	27475	Sales Re	AU
5	120123	Kimiko	Hotstone	F	26190	Sales Re	AU
6	120124	Lucian	Daymond	M	26480	Sales Re	AU

- List Input : 비표준데이터 처리

```
DATA subset3;
infile ‘d:\data\sales.txt' dlm=',';
INPUT Employee_ID First_Name $ Last_name $ Gender $ Salary Job_Title $ Country $
      Birth_Date :date. Hire_Date :mmddyy.
;
RUN;

PROC PRINT data=subset3; /* Birth_Date date9. Hire_Date :mmddyy. */
RUN;                     /* Birth_Date :date. Hire_Date mmddyy10. */
```

- ** : 포맷 문자형 자료를 읽을 때 지정길이에 관계없이 처음 공백이 나올 때까지 읽음

  :은 길이를 지정하지 않는다.fffffffffff

- 구분자 사례(&)

```
Region&State&Month&Expenses&Revenue
Southern&GA&JAN2001&2000&8000
Southern&GA&FEB2001&1200&6000
Southern&FL&FEB2001&8500&11000
Northern&NY&FEB2001&3000&4000
Northern&NY&MAR2001&6000&5000
Southern&FL&MAR2001&9800&13500
Northern&MA&MAR2001&1500&1000
```

```
PROC import datafile= “d:\data\special.txt” /*PROC IMPORT를 통해 input을 받을 수 있다.*/
out=mydata dbms=dlm replace; /*replace는 대체시키겠다는 말임*/
delimiter='&';
getnames=yes; /*첫줄에 변수명이 있어서 yes라고 해준다.*/
run;

options nodate ps=60 ls=80; /*option에 대한 이야기 이다. 나중에 더 자세히한다.*/
proc print data=mydata;
run;
```

OBS	Region	State	Month	Expenses	Revenue
1	Southern	GA	JAN2001	2000	8000
2	Southern	GA	FEB2001	1200	6000
3	Southern	FL	FEB2001	8500	11000
4	Northern	NY	FEB2001	3000	4000
5	Northern	NY	MAR2001	6000	5000
6	Southern	FL	MAR2001	9800	13500
7	Northern	MA	MAR2001	1500	1000

Data 읽기

- LENGTH 문장
  - 변수 길이를 지정 (자료값의 손실을 방지)
  - 숫자형은 2~8 바이트, 문자형은 1~32,767 까지 지정

```
LENGTH 변수명 $ length ;
```

```
DATA newlength;
SET mylib.internationaltours;
LENGTH Remarks $ 30;
if Vendor = 'Hispania' then Remarks = 'Bonus for 10+ people';
else if Vendor = 'Mundial' then Remarks = 'Bonus points';
else if Vendor = 'Major' then Remarks = 'Discount for 30+ people';
RUN;
```

Data 읽기 : 실습

- 외부 텍스트 데이터 읽기
  - 원자료가 보조기억장치에 있어서 자료를 SAS 프로그램과 함께 입력하지 않고, 따로 작성한 경우에는 INFILE문을 사용하여 외부데이터를 읽을 수 있음
  - 한가지 주의해야 할 것은 INFILE은 반드시 INPUT 문 앞에 와야 한다
- 예제
  - 데이터 : 20명의 통계학과 수강생들에 대한 기초조사 결과
  - 연령(세), 성별(1 남자, 2 여자), 키 (cm), 체중(kg)
  - 즐기는 음식(1 육류, 2 생선류, 3 채소류)을 조사한 결과

```
Data "d:\data\sample1.txt"
```

→ 외부파일(sample1)을 읽어오는 sas code를 작성하시오

Data 읽기

- INFILE 문에 사용되는 옵션
  - 외부파일명 : 인용부호(‘ 이나 “) 내에 파일이름 지정 (경로까지 모두 지정)
  - FIRSTOBS= 라인수 : 외부파일을 읽기 할 시작 위치(행번호)를 지정
  - OBS= 라인수 : 외부파일을 읽기 할 끝 위치(행번호)를 지정
  - LRECL=N : 읽고자 하는 레코드 폭을 지정(기본값은 132 열) (아직은 몰라도 된다.)
  - PAD : 가변 길이 레코드를 읽을 때， LRECL과 함께 사용 (아직은 몰라도 된다.)
  - MISSOVER : 자룟값을 읽지 못하는 변수에 대해서는 결측값으로 처리 하도록 지정

- STOPOVER : 읽고자 하는 레코드에 결측값이 있으면 데이터생산 중단

데이터의 길이가 매우 긴 경우 (136 컬럼을 넘어가는 경우)에는 LRECL의 값을 크게 줌
〔예를 들면, “LRECL=30000 PAD" 옵션을 줌〕

데이터 읽기 : 메뉴방식

- IMPORT WIZARD를 활용하여 읽기
  - 공백이 구분자인 텍스트 파일
  - [Data]d:\data\sample1.txt
  - [파일] > [데이터 가져오기]
  - [Standard data source]에 체크
  - 불러올 파일 종류를 선택 : 공백이 구분자인 파일이므로 Delimited File 선택