**Chapter XIX - Creating a Single Observation from Multiple Records**

1. 在写入多个records时，**INPUT** statement可以将这些records写入同一个observation中

* 方法1：

**INPUT** Lname $ 1-8 Fname $ 10-15;

**INPUT** Department $ 1-12 JobCode $ 15-19;

**INPUT** Salary comma10.;

* 方法2：

**INPUT** #1 Lname $ 1-8 Fname $ 10-15

#2 Department $ 1-12 JobCode $ 15-19

#3 Salary comma10.;

1. Using Line Pointer Controls

* *Input Statements for Column Specifications and Column Pointer Controls*

|  |  |
| --- | --- |
|  | **Example** |
| **Column Specifications** | **INPUT** Name $ 1-12 Age 15-16 Gender $ 18; |
| **Column Pointer Controls** | **INPUT** Name $12. @15 Age 2. @18 Gender $1.; |

* Two types of line pointer controls
* The forward slash (/) specifies a line location that is relative to the current one.
* The #n specifies the absolute number of the line to which you want to move the pointer.

1. Reading Multiple Records Sequentially

* Using the / Line Pointer Control

The / advances the input pointer to the next record. The / line pointer control moves the input pointer forward only and must be specified after the instructions for reading the values in the current record （相当于/是一个换行符号）

* Eg:

**INPUT** Lname $ 1-8 Fname $ 10-15 / Department $ 1-12 JobCode $ 15-19 / Salary comma10.;



* Number of Records per Observation

The raw data file must contain the same number of records for each observation. 如果其中有record的数量和其他的不相同，在写入observation时SAS会将后面的内容提前到missing的地方，导致数据错位。

Eg:

**DATA** perm.members;

**INFILE** memdata;

**INPUT** Fname $ Lname $ / Address $ 1-20 / City & $10. State $ Zip $;

***Raw Data File Memdata Verifying the Number of Records for Each Observation***

 



1. Reading Multiple Records Non-Sequentially

* The #*n* Line Pointer Control

The #n specifies the absolute number of the line to which you want to move the input pointer. The #n pointer control can read records in any order; therefore, it must be specified before the instructions for reading values in a specific record. (优先读取#后specify的一行)

Eg:

**DATA** perm.patients;

**INFILE** patdata;

**INPUT** #4 ID $5.

#1 Fname $ Lname $

#2 Address $23.

#3 City $ State $ Zip $

#4 @7 Doctor $6.;

**RUN**;

**PROC** **PRINT** data=perm.patients **NOOBS**;

**RUN**;



1. Combining Line Pointer Controls

Eg（same as the example above，output will exactly same as above）：

**DATA** perm.patients;

**INFILE** patdata;

**INPUT** #4 ID $5.

#1 Fname $ Lname $ / Address $23. / City $ State $ Zip $ / @7 Doctor $6.;

**RUN**;

1. Sample Program

**LIBNAME** perm 'c:\records\empdata';

**FILENAME** personel 'c:\records\empdata\new.dat';

**DATA** perm.emplist3;

**INFILE** personel;

**INPUT** #2 Department $ 5-16

#1 @16 ID $4. @1 Name $14. / JobCode 3. / Salary **COMMA9**.;

**RUN**;

**PROC** **PRINT** data=perm.emplist3;

**RUN**;

练习

1. Which INPUT statement correctly reads the values for ID in the fourth record, then returns to the first record to read the values for Fname and Lname?



****

1. How many records will be read for each iteration of the DATA step?

****

**DATA** spring.sportswr;

**INFILE** newitems;

**INPUT** #1 Item $ Color $

#3 @8 Price comma6.

#2 Fabric $

#3 SKU $ 1-6;

**RUN**;

1. one
2. two
3. three
4. four
5. Which INPUT statement correctly reads the values for City, State, and Zip?



1. **INPUT**#3 City $ State $ Zip $;
2. **INPUT** #3 City & $11. State $ Zip $;
3. **INPUT** #3 City $11. +2 State $2. + 2 Zip $5.;
4. all of the above
5. Which INPUT statement reads the values for Lname, Fname, Department, and Salary(in that order)?



