

## SAS

- Ha un'impostazione ormai superata
- Ha compatibilità col passato
- E' in grado di gestire dati di grandi dimensioni
- Presenta moduli aggiuntivi che permettono di gestire diverse cose in più ambiti

## COMMENTO

- */\*commento\*/*
  - serve per scrivere dei commenti sul foglio di sas senza che vadano a incidere sul programma

## GENERALITA'

- La scheda **log** è importante
  - in questa scheda viene trascritto come sas interpreta il programma
- I dati sono organizzati in **righe** e **colonne**: ciò compone il **dataset**
  - righe = **osservazioni**
  - colonne = **variabili**
- Ogni istruzione termina con ";"

## DATA STEP & PROC STEP

- SAS è distinto in:
  - **passi di data**:
    - permettono di creare, modificare, elaborare e gestire un dataset
    - iniziano sempre con *DATA*
  - **passi di procedura**:
    - permettono l'analisi di un dataset senza modificarlo
    - iniziano sempre con *PROC*

## LIBRERIE

- Per creare una libreria va scritto su SAS:
  - *LIBNAME (nome cartella) "(percorso cartella)"*
- Il nome della cartella deve avere massimo 8 caratteri
- Dentro la libreria saranno raccolti tutti i dataset che compariranno come file

## INPUT & INFILE

- Si usano per importare i dati dai file di testo o csv nel programma SAS
- Si usano in un data step
- *DATA (nome libreria).(nome del programma);*  
*INFILE "(percorso del file)";*  
*INPUT (variabili);*  
*run;*

## FIRSTOBS

- Se c'è bisogno di saltare la prima riga nella lettura del file va aggiunto *FIRSTOBS=2* nella riga di *INFILE*:
- *DATA (nome libreria).(nome del programma);*  
*INFILE "(percorso del file)" FIRSTOBS=2;*  
*INPUT (variabili);*  
*run;*

## \$.99.

- E' un simbolo che permette di leggere variabili con valori alfanumerici
- Va scritto prima (o forse dopo) la variabile

## DLM & DSD

- Nei file csv i dati sono separati dalla virgola, perciò si usano *DLM*="," *DSD* nella riga *INFILE* per modificare il delimitatore
  - *DATA* (nome libreria).(nome del programma);  
*INFILE* "(percorso del file)" *FIRSTOBS*=2 *DLM*="," *DSD*;  
*INPUT* (variabili);  
*run*;
- Nel caso in cui il delimitatore fosse il punto e virgola si scrive *DLM*=";" *DSD*

## OBS

- *OBS* permette di prendere i dati fino a una determinata riga
- Si scrive sempre nella riga *INFILE*:
  - *DATA* (nome libreria).(nome del programma);  
*INFILE* "(percorso del file)" *FIRSTOBS*=2 *OBS*=1730 *DLM*="," *DSD*;  
*INPUT* (variabili);  
*run*;

## DATE

- Quando una variabile presenta date, un po' come per i valori alfanumerici va scritto da parte alla variabile
  - *:MMDDYY10.*
  - *:DDMMYY10.*
  - *:YYMMDD10.*
  - *:DDMMYY8.*
  - *:DATE9.*
  - *:DATE7.*

## PROC CONTEST

- Mostra un sunto degli aspetti fondamentali del dataset
- E' utile dopo aver importato un file per capire se è stato importato bene
- *PROC CONTEST DATA* = (nome libreria).(nome dataset);  
*INFILE* "(percorso del file)" *FIRSTOBS*=2 *OBS*=1730 *DLM*="," *DSD*;  
*INPUT* (variabili);  
*run*;

## COPIARE UN DATASET IN UNO NUOVO

- Per incollare un dataset copiato bisogna crearne uno nuovo e va scritto *SET* (nome dataset da copiare)
  - *PROC CONTEST DATA* = (nome libreria).(nome dataset);  
*SET* (nome dataset da copiare);  
*run*;
- Per copiare il dataset prendendo solo determinate variabili, sotto *SET* va scritto *KEEP* (variabili da tenere)
  - *PROC CONTEST DATA* = (nome libreria).(nome dataset);  
*SET* (nome dataset da copiare);  
*KEEP* (variabili da tenere);  
*run*;
- Per copiare il dataset escludendo alcune variabili, sotto *SET* va scritto *DROP* (variabili da escludere)
  - *PROC CONTEST DATA* = (nome libreria).(nome dataset);  
*SET* (nome dataset da copiare);  
*DROP* (variabili da escludere);  
*run*;
- Ovviamente *KEEP* e *DROP* non possono coesistere nello stesso proc step
- A *KEEP* e *DROP* sotto si può scrivere *IF* (condizione variabile) *THEN OUTPUT* per prendere o escludere le variabili solo in determinate situazioni
  - *PROC CONTEST DATA* = (nome libreria).(nome dataset);  
*SET* (nome dataset da copiare);  
*KEEP/DROP* (variabili da tenere/escludere);

```
IF (condizione variabile) AND/OR/NOT (condizione variabile) THEN OUTPUT
run;
```

## CREARE UNA NUOVA VARIABILE

- Per creare una nuova variabile basta scrivere il nome della variabile e l'operazione a essa associata, per esempio:
  - *PROC CONTEST DATA = (nome libreria).(nome dataset);*  
*SET (nome dataset da copiare);*  
*KEEP/DROP (variabili da tenere/escludere);*  
*Somma = Var\_A + Var\_B*  
*Differenza = Var\_A - Var\_B*  
*IF Differenza < 3 AND/OR/NOT Somma > 7 THEN OUTPUT*  
*run;*

## FUNZIONI ED ESPRESSIONI

- **MINIMO**
  - *DATA (nome libreria).(nome del programma);*  
*SET (nome dataset da copiare);*  
*Minimo= MIN (variabile1, variabile2,...);*  
*run;*
- **MASSIMO**
  - *DATA (nome libreria).(nome del programma);*  
*SET (nome dataset da copiare);*  
*Massimo= MAX (variabile1, variabile2,...);*  
*run;*
- **MEDIA**
  - *DATA (nome libreria).(nome del programma);*  
*SET (nome dataset da copiare);*  
*Media= MEAN (variabile1, variabile2,...);*  
*run;*
- **LUNGHEZZA**
  - *DATA (nome libreria).(nome del programma);*  
*SET (nome dataset da copiare);*  
*Lunghezza= LENGHT (variabile\_alfanumerica);*  
*run;*
- **SUPERFICIE**
  - *DATA (nome libreria).(nome del programma);*  
*SET (nome dataset da copiare);*  
*S = SUBSTR (variabile1, n, k);*  
*run;*
  - E' utile accoppiarlo a *INDEX*
- **INDEX**
  - *DATA (nome libreria).(nome del programma);*  
*SET (nome dataset da copiare);*  
*S = SUBSTR (variabile1, n, k);*  
*I = INDEX (variabile1, "a");*  
*run;*
- **Funzioni collegate alle date**
  - *DATA (nome libreria).(nome del programma);*  
*SET (nome dataset da copiare);*  
*Mese = MONTH (variabile);*  
*Anno = YEAR (variabile);*  
*Giorno = DAY (variabile);*  
*run;*
- Le funzioni sono sempre dentro un data step

## ESPORTAZIONE DI UN DATASET

- Si esporta attraverso un data step, dove si copia il dataset vecchio, poi con *FILE* e *PUT* si va a esportare il dataset selezionando anche le variabili da esportare
- *DATA (nome libreria).(nome del programma nuovo);*  
*SET (nome dataset da copiare);*  
*FILE (nuovo nome dataset) DLM=" ", DSD;*  
*PUT (variabili);*

run;

- Esiste un dataset che consente l'esportazione senza che venga creato un nuovo dataset:
  - DATA\_NULL\_

## PROC PRINT

- Si usa per stampare
  - PROC PRINT DATA = (nome del programma);
  - run;
- Si possono selezionare le variabili da stampare con VAR (variabile1, variabile2,...)
  - PROC PRINT DATA = (nome del programma);
  - VAR (variabile1, variabile2,...)
  - run;
- Si possono selezionare le variabili dove le osservazioni assumono un certo valore aggiungendo WHERE (condizione variabile)
  - PROC PRINT DATA = (nome del programma);
  - VAR (variabile1, variabile2,...)
  - WHERE (condizione variabile)
  - run;
- Si può anche dare un titolo alla stampa con TITLE "(titolo desiderato)"
  - PROC PRINT DATA = (nome del programma);
  - VAR (variabile1, variabile2,...)
  - WHERE (condizione variabile)
  - TITLE "(titolo desiderato)"
  - run;
- Per cambiare il formato, senza modificare il contenuto, si aggiunge FORMAT (variabile) (nuovo formato della variabile)
  - PROC PRINT DATA = (nome del programma);
  - VAR (variabile1, variabile2,...);
  - WHERE (condizione variabile);
  - TITLE "(titolo desiderato)";
  - FORMAT (variabile) (nuovo formato della variabile);
  - run;
  - PROC PRINT DATA = (nome del programma);
  - VAR (variabile1, variabile2,...);
  - WHERE (condizione variabile);
  - TITLE "(titolo desiderato)";
  - FORMAT cacca\_73 \$.99;
  - run;

## PROC SORT (ORDINAMENTO)

- Per ordinare, prima va copiato il dataset, poi si usa PROC SORT DATA = (nome dataset) con sotto BY/BY DESCENDING (variabile) e infine si stampa con PROC PRINT
  - BY = ordine crescente
  - BY DESCENDING = ordine decrescente
  - DATA (nuovo nome dataset);
  - SET (dataset da copiare);
  - run;
  - PROC SORT DATA = (nome dataset);
  - BY/BY DESCENDING (variabile);
  - run;
  - PROC PRINT DATA = (nome dataset);
  - run;

## COME CREARE UN NUOVO FORMATO

- Per creare un nuovo formato va creato un nuovo dataset dove si copierà il vecchio dataset e si aggiungerà una nuova variabile che assume un determinato valore in base alla condizione della variabile
  - DATA (nuovo nome dataset);
  - SET (dataset da copiare);
  - variabile\_nuova = 0
  - IF variabile1 > x THEN variabile\_nuova = 1;
  - run;
  - PROC FORMAT;

```

VALUE variabile1 0 = "piccolo"
VALUE variabile1 1 = "grande"
run;
/*ora si può stampare*/
PROC PRINT DATA = (nome dataset);
  FORMAT variabile_nuova variabile1.;
run;

```

### PROC MEANS (ESTRAZIONE DI DATI)

- Con la *PROC MEANS* si possono ottenere dati statistici relativi ai file
  - *PROC MEANS DATA = (nome dataset) MEANS MIN STD MAX N NMISS;*  
*VAR/CLASS (variabile1, variabile2,...);*  
*OUTPUT OUT = (nome dataset) MEAN = Media*  
*run;*
    - *VAR* si usa per variabili quantitative
    - *CLASS* si usa per le variabile qualitative
  - Come al solito dopo la proc means si può stampare con la *PROC PRINT*

### UNIONE DI 2 DATASET (DATA TOTALE)

- Per unire 2 dataset con stesse variabili si usa *DATA TOTALE*
  - *DATA TOTALE;*  
*SET (nome dataset 1) (nome dataset 2);*  
*run;*

### FUSIONE DI 2 DATASET

- E' importante che i nomi delle 2 variabili siano uguali
- Prima si stampano i 2 dataset, poi si rinomina il dataset, poi si ordinano per la stessa variabile e infine si fondono con *MERGE*
- *PROC PRINT DATA = (nome dataset 1);*  
*run;*  
*PROC PRINT DATA = (nome dataset 2);*  
*run;*  
*DATA (nome dataset);*  
*SET (nome dataset);*  
*RENAME (nome variabile vecchia) = (nome variabile nuova);*  
*run;*  
*PROC SORT DATA = (nome dataset 1)*  
 *BY (variabile nuova);*  
*run;*  
*PROC SORT DATA = (nome dataset 2)*  
 *BY (variabile nuova);*  
*run;*  
*DATA (nome dataset fuso);*  
*MERGE (dataset 1) (dataset 2);*  
 *BY (variabile nuova);*  
 *if (variabili) ne .;*  
*run;*

### PROC FREQ

- La *PROC FREQ* si usa per conteggiare quante osservazioni compaiono per variabile, dando frequenza relativa, assoluta e il numero assoluto e in percentuale
  - *PROC FREQ DATA = (nome dataset)*  
 *TABLES (variabile);*  
*run;*

### ODS

- ODS si usa per estrarre oggetti ed emetterli su una nuova destinazione
  - *ODS TRACE ON;*  
*PROC FREQ DATA = (nome dataset);*  
 *TABLES (variabile)\*(variabile);*  
*run;*  
*ODS TRACE OFF;*  
*PROC FREQ DATA = (nome dataset);*

```
TABLES (variabile)*(variabile);  
ODS OUTPUT (codice path) = (nome dataset desiderato);  
run;
```

