

SAS CHEAT SHEET

DATA STATEMENT

1. Modifying data- Numerical changes

```
DATA LIBNAME.NEWDATA;  
SET LIBNAME.OLDDATA;  
FORMAT X Y 10.2;  
X= LOG (AGE); Y= HEIGHT^2; RUN;
```

2. STRING FUNCTIONS

To get the Length of the variable

```
LEN(Variable)
```

To concatenate two variables

```
Variable1 || Variable2
```

To remove extraneous spaces

```
STRIP(Variable)
```

```
COMPRESS(var1 | var2)
```

3. SUBSTRING FUNCTIONS

- **To extract 3 objects from the first position**

```
SUBSTR(VAR, 1, 3)
```

- **UPCASE(Var)/LOWCASE(Var)**

- **Extraction from a variable[Value=123]**

```
SCAN (VAR, 1, 3)
```

```
RESULT- 12
```

4. FILTERING DATA

KEEP /DROP

```
DATA LIB.NEW (DROP = VAR1 VAR2);
```

```
SET LIB.OLD (KEEP = VAR1 VAR2 VAR3 VAR4);
```

```
RUN;
```

5. DATE FUNCTIONS

```
DAY(A)
```

```
MONTH(A)
```

```
YEAR(A)
```

```
QTR(A)
```

```
WEEKDAY(A)
```

```
MDY(Y, X, Z)
```

```
TODAY()
```

To know the End Date when start date and duration is known

```
INTNX("UNIT", START DATE, DURATION)
```

To get the date of the end of the month/week

```
INTNX("UNIT", START DATE, DURATION,'E')
```

To get the date of the start of the month/week

```
INTNX("UNIT", START DATE, DURATION,'S')
```

To know the duration when End Date and Start Date is given

```
INTCK("UNIT", DATE1, DATE2)
```

PROC STATEMENT

1. STRUCTURE OF THE DATA

```
PROC CONTENTS LIBNAME.DATASET;  
RUN;
```

2. TO VIEW THE DATA IN THE OUTPUT WINDOW

```
PROC PRINT DATA LIBNAME.NEW;  
VAR VARIABLE1 VARIABLE2; RUN;
```

TO OBSERVE THE FIRST 10 OBS.

```
PROC PRINT DATA NEW (OBS = 10);  
RUN;
```

3. SORTING

BY VARIABLE

```
PROC SORT DATA = LIBNAME.NEW;
```

```
BY VAR1 DESCENDING VAR2;RUN;
```

4. DESCRIPTIVE STATISTICS

[BY DEFAULT A SUMMARY IS CREATED IN THE OUTPUT WINDOW]

```
PROC MEANS DATA=LIB.DATA; RUN;
```

FOR A SPECIFIC MEASURE

```
PROC MEANS DATA=LIB.DATA MEAN;
```

```
RUN;
```

FOR A SPECIFIC CONDITION

```
PROC MEANS DATA=LIB.DATA;
```

```
WHERE SEX= "F";RUN;
```

BY STATEMENT

```
PROC MEANS DATA=LIB.DATA MEAN;
```

```
BY SEX; RUN;
```

CLASS STATEMENT

```
PROC MEANS DATA=LIB.DATA MEAN;
```

```
CLASS SEX; RUN;
```

```
PROC SUMMARY DATA=LIB.DATA MEAN;
```

```
VAR X; CLASS SEX;
```

```
OUTPUT OUT= LIB.OUT; RUN;
```

6. IF-THEN-ELSE

```
DATA NEW; SET OLD;
IF CONDITION THEN ACTION;
ELSE IF CONDITION THEN ACTION;
ELSE IF CONDITION THEN ACTION;
...;
ELSE ACTION; RUN;
```

7. MERGING

- **X & Y**

```
DATA LIB.M;
MERGE LIB.A(IN = X) LIB.C(IN = Y);
BY M;
IF X & Y; RUN;
```

- **X & NOT(Y)**

```
DATA LIB.M;
MERGE LIB.A(IN = X) LIB.C(IN = Y);
BY M;
IF X & NOT(Y); RUN;
```

- **IF X**

```
DATA LIB.M;
MERGE LIB.A(IN = X) LIB.C(IN = Y);

BY M;
IF X; RUN;
```

- **IF Y**

```
DATA LIB.M;
MERGE LIB.A(IN = X) LIB.C(IN = Y);
BY M;
IF Y; RUN;
```

5. SQL IN SAS

```
PROC SQL;
CREATE TABLE DATANAME AS SELECT *
FROM SOURCE DATA
WHERE CONDITION; QUIT;
```

DISTINCT/COUNT

```
PROC SQL;
CREATE TABLE DATANAME AS SELECT
COUNT(DISTINCT(VAR)) AS CNT
FROM SOURCE DATA; QUIT;
```

MERGING USING SQL

```
PROC SQL;
CREATE TABLE DATANAME AS SELECT *
FROM DATA1.A X DATA2.B Y
WHERE X.M=Y.L;QUIT;
[UNIQUE COLUMN OF X IS M AND Y IS M]
```

6. CREATING MACROS

```
%MACRO UNI(DT, VA, OB)

PROC UNIVARIATE DATA= &DT
NEXTROBS= &OB; VAR &VA;RUN;MEND;
```

7. REMOVAL OF DUPLICATES

BY VARIABLES

```
PROC SORT DATANAME NODUPKEY;
OUT= LIB.O DUPOUT= LIB.D; BY SEX;
RUN;
```

BY ALL OBSERVATIONS

```
PROC SORT DATANAME NODUP;
OUT= LIB.O DUPOUT=LIB.D;BY SEX;
RUN;
```

8. CREATING PLOT

```
PROC UNIVARIATE DATANAME PLOT;

VAR A; RUN;

PROC GPLOT DATANAME; PLOT A*B; RUN;
```