/\*Creating a SAS Program\*/

/\* Data Datasetname;

Input variable-list;

cards/datalines;

.

.Input Data

.

;

run; \*/

Data abc;

input a b c;

datalines;

1 2 3

4 5 6

7 8 9

;

run;

\* DISPLAY THE DATA PORTION OF DATA SET;

PROC PRINT DATA=abc;

RUN;

/\* CREATING A PERMANENT LIBRARY;

1) Using Libname

2) Using Graphical Interface \*/

/\*Libname libraryname 'Physical-Path';\*/

libname mylib '/home/u1048896';

Data mylib.abc;

input a b c;

datalines;

1 2 3

4 5 6

7 8 9

;

run;

PROC PRINT DATA=mylib.abc;

RUN;

\* Use of '$' ;

Data mylib.name;

input fname $ lname $ salary;

datalines;

saurabh jani 100

ram kapoor 200

rajesh sharma 300

;

run;

PROC PRINT DATA=mylib.name;

RUN;

\* Use of length statement;

Data mylib.emp;

length address $ 20;

input fname $ lname $ salary address $;

datalines;

saurabh jani 100 kormangala

ram kapoor 200 jaynagara

rajesh sharma 300 viveknagar

;

run;

PROC PRINT DATA=mylib.emp;

RUN;

\* Avoid use of length statement;

Data mylib.emp;

input fname $ lname $ salary address $ 20.;

datalines;

saurabh jani 100 kormangala

ram kapoor 200 jaynagara

rajesh sharma 300 viveknagar

;

run;

PROC PRINT DATA=mylib.emp;

RUN;

/\* Set Statement - Generate new dataset from existing dataset or

modification existing dataset \*/

data mylib.newdata;

set mylib.emp;

run;

PROC PRINT DATA=mylib.newdata;

RUN;

\* Creating a new variable in dataset;

data mylib.newdata;

set mylib.emp;

bonus=salary \* .1;

run;

PROC PRINT DATA=mylib.newdata;

RUN;

\* Creating muliple dataset from single dataset;

\* Creating single dataset from multiple dataset;

Data mylib.Student;

Input Name $ Age Gender $;

cards;

Alex 20 M

Sady 21 F

Amanda 22 F

John 21 M

Jane 19 F

Mike 18 M

Chris 22 M

Nayasa 20 F

;

run;

/\* Creating muliple dataset from single dataset \*/

Data mylib.Male mylib.Female;

set mylib.Student;

if Gender='M' then

output mylib.Male;

else if Gender='F' then

Output mylib.Female;

run;

PROC PRINT DATA=mylib.Student;

RUN;

PROC PRINT DATA=mylib.male;

RUN;

PROC PRINT DATA=mylib.female;

RUN;

/\* Creating single dataset from multiple dataset \*/

Data mylib.stud;

set mylib.Male mylib.Female;

run;

PROC PRINT DATA=mylib.stud;

RUN;

\* Variable Names are different.;

Data mylib.male;

set mylib.male;

a=age\*10;

run;

Data mylib.female;

set mylib.female;

b=age\*20;

run;

Data mylib.stud;

set mylib.Male mylib.Female;

run;

PROC PRINT DATA=mylib.stud;

RUN;

/\* New variable creation based on condition \*/

DATA mylib.EMP;

INPUT ID NAME $ DESIGNATION $ SALARY;

CARDS;

1 SAURABH ANALYST 200

2 ROHAN SENIOR 300

3 RAJ ANALYST 220

4 MOHAN SENIOR 340

5 RAMESH MANAGER 500

6 VINOD CONTRACT 100

;

run;

PROC PRINT DATA=mylib.emp;

RUN;

DATA mylib.NEWEMP;

SET mylib.EMP;

IF DESIGNATION='ANALYST' THEN

BONUS=50;

ELSE IF DESIGNATION='SENIOR' THEN

BONUS=80;

ELSE IF DESIGNATION='MANAGER' THEN

BONUS=100;

ELSE

BONUS=0;

run;

PROC PRINT DATA=mylib.NEWEMP;

RUN;

\* Multiple conditions ;

DATA mylib.NEWEMP;

SET mylib.EMP;

IF DESIGNATION='ANALYST' and Name='SAURABH' THEN

BONUS=50;

ELSE IF DESIGNATION='SENIOR' and Name='ROHAN' THEN

BONUS=80;

ELSE IF DESIGNATION='MANAGER' and Name='RAMESH' THEN

BONUS=100;

ELSE

BONUS=0;

run;

PROC PRINT DATA=mylib.NEWEMP;

RUN;

/\* Multiple operation in IF else statement \*/

/\* Use Do -- End statement \*/

DATA mylib.NEWEMP;

SET mylib.EMP;

IF DESIGNATION='ANALYST' THEN

DO;

BONUS=50;

NEWSALARY=SALARY+SALARY\*.10;

END;

ELSE IF DESIGNATION='SENIOR' THEN

DO;

BONUS=80;

NEWSALARY=SALARY+SALARY\*.15;

END;

ELSE IF DESIGNATION='MANAGER' THEN

DO;

BONUS=100;

NEWSALARY=SALARY+SALARY\*.20;

END;

ELSE

DO;

BONUS=0;

NEWSALARY=SALARY+SALARY\*.05;

END;

run;

PROC PRINT DATA=mylib.NEWEMP;

RUN;

\* Data set options;

\* Drop statement;

DATA mylib.NEWEMP;

SET mylib.EMP;

drop ID DESIGNATION;

run;

PROC PRINT DATA=mylib.EMP;

RUN;

PROC PRINT DATA=mylib.NEWEMP;

RUN;

\* Keep statement;

DATA mylib.NEWEMP;

SET mylib.EMP;

keep ID DESIGNATION;

run;

PROC PRINT DATA=mylib.EMP;

RUN;

PROC PRINT DATA=mylib.NEWEMP;

RUN;

\* Mathematical operations;

data mylib.inputd;

input ID a b c;

cards;

1 2 3 4

4 5 6 7

;

run;

PROC PRINT DATA=mylib.inputd;

RUN;

DATA mylib.output1;

SET mylib.inputd;

sum\_col=sum(a, b, c);

drop a b c;

run;

PROC PRINT DATA=mylib.inputd;

RUN;

PROC PRINT DATA=mylib.output1;

RUN;

\* Use set and drop statment together;

DATA mylib.output1;

SET mylib.inputd(drop=a b c);

sum\_col=sum(a, b, c);

run;

PROC PRINT DATA=mylib.inputd;

RUN;

PROC PRINT DATA=mylib.output1;

RUN;

DATA mylib.output1(drop=a b c);

SET mylib.inputd;

sum\_col=sum(a, b, c);

run;

PROC PRINT DATA=mylib.inputd;

RUN;

PROC PRINT DATA=mylib.output1;

RUN;

\* Rename options;

DATA mylib.output1;

SET mylib.inputd;

sum\_col=sum(a, b, c);

rename ID=empid a=First;

run;

PROC PRINT DATA=mylib.inputd;

RUN;

PROC PRINT DATA=mylib.output1;

RUN;

\* Rename with data step;

DATA mylib.output1(rename=(ID=empid));

SET mylib.inputd;

sum\_col=sum(a, b, c);

run;

PROC PRINT DATA=mylib.inputd;

RUN;

PROC PRINT DATA=mylib.output1;

RUN;

\* Why to use math functions instead of math operator;

data mylib.sumop;

input a b c;

cards;

. 2 3

. 5 6

7 . 9

1 3 4

;

run;

PROC PRINT DATA=mylib.sumop;

RUN;

data mylib.sumopout;

set mylib.sumop;

sum\_1=sum(a, b, c);

sum\_2=a+b+c;

run;

PROC PRINT DATA=mylib.sumop;

RUN;

PROC PRINT DATA=mylib.sumopout;

RUN;

/\* Operator

= EQ

>= GE

<= LE

> GT

< LT

~= or ^= NE

\*/

Data mylib.output2;

set mylib.emp;

\* where id in (1, 4, 5);

/\* (id=4) or (id=1) or (id=5) \*/

\* where salary between 200 and 500;

/\*(Salary >= 200) and (Salary<=500) \*/

where DESIGNATION ~='ANALYST';

/\* DESIGNATION NE 'ANALYST' \*/

run;

Proc Print data=mylib.emp;

run;

Proc Print data=mylib.output2;

run;

Data mylib.output2;

set mylib.emp;

where not (id in (1, 4, 5));

/\* (id NE 4) and (id NE 1) and (id NE 5) \*/

run;

Proc Print data=mylib.emp;

run;

Proc Print data=mylib.output2;

run;

/\* Missing Value in SAS

Numberic data = . (dot)

Char data = '' (blank)

\*/

data mylib.test;

input a b c;

cards;

1 . 3

4 5 6

7 . 9

;

run;

Data mylib.output3;

set mylib.test;

where b is not missing;

run;

Proc print data=mylib.test;

run;

Proc print data=mylib.output3;

run;

Data mylib.output2;

set mylib.emp;

where name contains 'AN';

run;

Proc print data=mylib.emp;

run;

Proc print data=mylib.output2;

run;

Data mylib.output2;

set mylib.emp;

where name like 'R%';

run;

Proc print data=mylib.emp;

run;

Proc print data=mylib.output2;

run;

Data mylib.output2(where=(name like 'R%'));

set mylib.emp;

run;

Proc print data=mylib.emp;

run;

Proc print data=mylib.output2;

run;

Data mylib.output2;

set mylib.emp;

where name like '\_I\_\_\_';

run;

Proc print data=mylib.emp;

run;

Proc print data=mylib.output2;

run;