```
LIBNAME SSN5 '/home/debendra330/BATCH_202404/SESSION_5/A3.SAS_DATASET';
RUN;

/* HOW TO TRANSPOSE DATA FROM ROWS TO COLUMNS AND COLUMNS TO ROWS */
=================================================================================
```

```
DATA PROD_SALES;
INFILE CARDS DSD DLM='09'X;
INPUT PROD $ BANGALORE DELHI MUMBAI CHENNAI BHUBANESWAR KOLKATA ;
CARDS;
APPLE   43400   15600   48100   57900   10100   35500
DELL    54000   79500   26000   31700   34200   16200
ACER    15300   23500   60400   58200   22700   87300
ASUS    15500   24300   65900   57800   11100   74700
LENOVO  21000   85000   22500   51500   19300   64100
SAMSUNG 72000   12300   36400   21900   22700   79600
;
RUN;
```

```
PROC PRINT DATA=PROD_SALES;
RUN;

/* LETS TRANSPOSE THIS DATASET FROM COLUMN TO ROW BY PRODUCT */
```

```
PROC SORT DATA=PROD_SALES;
BY PROD;
RUN;
```

```
PROC TRANSPOSE DATA=PROD_SALES OUT=PROD_SALES_V1 (RENAME=(_NAME_=CITY COL1=SALES));
VAR BANGALORE DELHI MUMBAI CHENNAI BHUBANESWAR KOLKATA;
BY PROD;
RUN;
```

```
PROC PRINT DATA=PROD_SALES_V1;
RUN;

/* 1. WE USE OUT= TO CREATE A NEW DATASET WITH TRANSPOSED DATA */
/* VAR STATEMENT IS USED TO DECLARE ALL COLUMNS THAT NEED TO BE BROUGHT FROM COLUMN TO ROW */
/* BY STATEMENT IS USED TO DECLARE THE COLUMN THAT IS TO BE KEPT CONSISTENT */

/* EXAMPLE */
```

```
DATA STU_SCORE;
INPUT STU_NAME $ EXCEL VBA SQL SAS PYTHON;
CARDS;
LAXMI 80 90 40 50 80
DEV 40 90 20 40 60
LAXMAN 50 90 80 70 60
;
RUN;
```

```
PROC PRINT DATA=STU_SCORE;
RUN;
```

```
PROC SORT DATA=STU_SCORE;
BY STU_NAME;
RUN;
```

```
PROC TRANSPOSE DATA=STU_SCORE OUT=STU_SCORE_V1 (RENAME=(_NAME_=SUBJECT COL1=MARKS));
VAR EXCEL VBA SQL SAS PYTHON;
BY STU_NAME;
RUN;
```

```
PROC PRINT DATA=STU_SCORE_V1;
```

```
RUN;

/* TRANSPOSE DATA FROM ROW TO COLUMN */

DATA PROD_SALES;
INPUT PROD $ CITY $ SALES;
CARDS;
APPLE BGLR 89000
APPLE DL 60000
APPLE BBSR 56000
APPLE HYD 90000
DELL BGLR 83000
DELL DL 60000
DELL BBSR 50000
DELL HYD 95000
HP BGLR 89000
HP DL 65000
HP BBSR 56000
HP HYD 94000
;
RUN;

PROC PRINT DATA=PROD_SALES;
RUN;

PROC SORT DATA=PROD_SALES;
BY PROD;
RUN;

PROC TRANSPOSE DATA=PROD_SALES OUT=PROD_SALES_V1 (DROP=_NAME_);
ID CITY;
VAR SALES;
BY PROD;
RUN;

PROC PRINT DATA=PROD_SALES_V1;
RUN;

/* 1. ID STATEMENT - TELLS SAS WHICH ROW VALUES ARE CONVERTED INTO COLUMN */
/* 2. VAR STATEMENT - TELL SAS TO SELECT A PARTICULAR NUMERIC FIELD */
/* 3. BY STATEMENT - TELLS SAS TO TRANSPOSE THE DATASET BY A PARTICULAR COLUMN */

/* EXAMPLE */

DATA PROD_YEAR_SALES;
INFILE CARDS DSD DLM='09'X;
INPUT PROD $ YEAR BANGALORE DELHI MUMBAI CHENNAI BHUBANESWAR KOLKATA;
CARDS;
APPLE   2019    43400   15600   48100   57900   10100   35500
DELL    2019    54000   79500   26000   31700   34200   16200
ACER    2019    15300   23500   60400   58200   22700   87300
ASUS    2019    15500   24300   65900   57800   11100   74700
LENOVO  2019    21000   85000   22500   51500   19300   64100
SAMSUNG 2019    72000   12300   36400   21900   22700   79600
APPLE   2020    19500   32100   17500   53300   41300   38900
DELL    2020    66600   26200   48300   31300   20800   69800
ACER    2020    23900   89700   17100   88400   47200   11300
ASUS    2020    25600   10000   74700   42200   81900   15400
LENOVO  2020    26800   62500   44500   86300   65700   53700
SAMSUNG 2020    19400   38900   40200   65700   54600   84600
;
RUN;

PROC PRINT DATA=PROD_YEAR_SALES;
RUN;
```

```
/* TRANSPOSE THIS DATASET FROM COLUMN TO ROW BY PROD AND YEAR */

PROC SORT DATA=PROD_YEAR_SALES;
BY PROD YEAR;
RUN;

PROC TRANSPOSE DATA=PROD_YEAR_SALES OUT=PROD_YEAR_SALES_V1 (RENAME=(_NAME_=CITY COL1=SALES));
VAR BANGALORE DELHI MUMBAI CHENNAI BHUBANESWAR KOLKATA;
BY PROD YEAR;
RUN;

PROC PRINT DATA=PROD_YEAR_SALES_V1;
RUN;

/* TRANSPOSE FROM ROW TO COLUMN */

PROC PRINT DATA=PROD_YEAR_SALES_V1;
RUN;

PROC SORT DATA=PROD_YEAR_SALES_V1;
BY PROD YEAR;
RUN;

PROC TRANSPOSE DATA=PROD_YEAR_SALES_V1 OUT=PROD_YEAR_SALES_V2 (DROP=_NAME_);
ID CITY;
VAR SALES;
BY PROD YEAR;
RUN;

PROC PRINT DATA=PROD_YEAR_SALES_V2;
RUN;

/* EXAMPLE */

DATA MED_SELECT;
SET SSN5.MED_2023;
KEEP CUSTOMER_ID COMPANY GENDER AGE STATE_CODE NO_OF_TRIPS SPENT_AMOUNT;
RUN;

PROC SQL;
CREATE TABLE MED_SUMMARY AS
SELECT STATE_CODE, COMPANY, GENDER,
COUNT(CUSTOMER_ID) AS SUBS,
SUM(NO_OF_TRIPS) AS VISITS,
SUM(SPENT_AMOUNT) AS SPENT
FROM MED_SELECT
GROUP BY STATE_CODE, COMPANY, GENDER
ORDER BY 1,2,3;
QUIT;

PROC PRINT DATA=MED_SUMMARY;
RUN;

PROC SORT DATA=MED_SUMMARY;
BY STATE_CODE GENDER;
RUN;

PROC TRANSPOSE DATA=MED_SUMMARY OUT=MED_SUMMARY_V1 (RENAME=(_NAME_ = PARAMETER));
ID COMPANY;
VAR SUBS VISITS SPENT;
BY STATE_CODE GENDER;
RUN;

PROC PRINT DATA=MED_SUMMARY_V1;
```

```
RUN;

/* WHAT IS RETAIN STATEMENT */
=========================================

DATA STU_SCORE;
INFILE CARDS DSD DLM='09'X;
INPUT STU_NAME $ SAS SQL EXCEL PYTHON VBA R QLIK ML AI TABLEAU POWER_BI;
CARDS;
BHAWNA  74  50  71  87  46  94  82  50  88  82  70
SUBHASHREE  35  41  95  81  71  63  98  96  96  41  93
KALYAN  95  83  49  52  43  38  61  59  53  37  82
RAJIV   56  85  72  60  36  70  35  86  59  93  32
RENUKA  90  57  37  37  44  85  47  80  66  52  71
MANJULA 95  30  88  76  77  100 92  58  84  87  94
MAHESH  66  66  41  81  62  78  91  49  96  89  83
PARTHO  61  62  68  41  56  55  57  96  94  83  50
SUSHANK 84  60  55  57  52  50  93  45  74  85  70
HARSHA  75  45  46  49  48  68  30  71  73  59  30
ROHAN   67  93  93  62  32  48  87  63  66  36  39
KARTHIK 68  35  36  30  40  72  74  58  37  71  40
VINAY   66  39  85  30  74  49  88  98  44  69  64
MOHINI  66  82  69  89  75  42  88  65  56  42  99
;
RUN;

PROC PRINT DATA=STU_SCORE;
RUN;

/* FROM THE ABOVE DATASET, CAN YOU CHANGE THE ORDER OF COLUMNS */

/* STU_NAME EXCEL VBA SQL SAS PYTHON R TABLEAU POWER_BI ML AI */

DATA STU_SCORE_V1;
SET STU_SCORE;
KEEP STU_NAME EXCEL VBA SQL SAS PYTHON R TABLEAU POWER_BI ML AI;
RUN;

PROC PRINT;
RUN;

DATA STU_SCORE_V1;
RETAIN STU_NAME EXCEL VBA SQL SAS PYTHON R TABLEAU POWER_BI ML AI;
SET STU_SCORE;
RUN;

PROC PRINT;
RUN;

/* WE USE RETAIN STATEMENT TO CHANGE THE ORDER OF VARIABLES IN SAS */

/* WHAT IS THE DIFFERENCE BETWEEN SUM AND ADDITION */
================================================================

DATA STU_SCORE;
INFILE CARDS DSD DLM='09'X MISSOVER;
INPUT STU_NAME : $20. SAS R PYTHON SQL EXCEL TABLEAU POWER_BI;
CARDS;
BHAWNA  38  54  97  37  31  21  98
SUBHASHREE  94  21  93  73  69  53  71
KALYAN  49  46      86  34  51  81
RAJIV   74  92  16  42          12
RENUKA  55          79  44  77  33
MANJULA 48  98  45  87  82  43  18
;
```

```
RUN;

PROC PRINT DATA=STU_SCORE;
RUN;

DATA STU_SCORE;
SET STU_SCORE;
TOTAL_SCORE_1 = SAS + R + PYTHON + SQL + EXCEL + TABLEAU + POWER_BI;
TOTAL_SCORE_2 = SUM(SAS,R,PYTHON,SQL,EXCEL,TABLEAU,POWER_BI);
RUN;

PROC PRINT DATA=STU_SCORE;
RUN;

/* IN ADDITION, IF WE HAVE BLANK VALUES, THEN THE EXPRESSION DOES NOT WORK */
/* SUM WILL CORRECTLY SHOW THE TOTAL EVEN IF BLANKS ARE PRESENT */

/* WHAT IS SAS COMPARE PROCEDURE */
================================================

DATA PROD_SALES1;
INPUT PROD $ SALES;
CARDS;
APPLE 800
DELL 500
HP 300
;
RUN;

DATA PROD_SALES2;
INPUT PROD $ SALES;
CARDS;
APPLE 1000
DELL 500
ACER 300
;
RUN;

PROC PRINT DATA=PROD_SALES1;
RUN;

PROC PRINT DATA=PROD_SALES2;
RUN;

/* COMPARISION OF DATASET AND IDENTIFYING THE DIFFERENCES */

PROC COMPARE BASE=PROD_SALES1 COMPARE=PROD_SALES2;
RUN;

PROC COMPARE BASE=PROD_SALES1 COMPARE=PROD_SALES2;
VAR SALES;
RUN;

/* WHAT IS SAS UPDATE STATEMENT */
=============================================

DATA PROD_PRICE_202001;
INFILE CARDS DSD DLM=',' MISSOVER;
INPUT PRODUCT $ PRICE MONTH $;
CARDS;
APPLE,89000,202001
DELL,67000,202001
ACER,76000,202001
ASUS,80000,202001
LENOVO,65000,202001
```

```
SAMSUNG,89000,202001
;
RUN;

DATA PROD_PRICE_202002;
INFILE CARDS DSD DLM=',' MISSOVER;
INPUT PRODUCT $ PRICE MONTH $;
CARDS;
APPLE,89000,202002
DELL,80000,202002
ACER,76000,202002
ASUS,70000,202002
TOSHIBA,54000,202002
HCL,34000,202002
;
RUN;

PROC PRINT DATA=PROD_PRICE_202001;
RUN;

PROC PRINT DATA=PROD_PRICE_202002;
RUN;

/* GIVE ME A DATASET WITH THE LATEST PRICE INFORMATION */

PROC SORT DATA=PROD_PRICE_202001;
BY PRODUCT;
RUN;

PROC SORT DATA=PROD_PRICE_202002;
BY PRODUCT;
RUN;

DATA PROD_PRICE_UPDATED;
UPDATE PROD_PRICE_202001 PROD_PRICE_202002;
BY PRODUCT;
RUN;

/* HOW TO CREATE A ROW NUMBER IN SAS */

DATA MED_SELECT;
SET SSN5.MED_2023;
KEEP CUSTOMER_ID COMPANY GENDER AGE STATE_CODE SPENT_AMOUNT ROW_NUMBER;
ROW_NUMBER = _N_;
RUN;

/* HOW TO GET 100 RANDOM OBS */

PROC SURVEYSELECT DATA=MED_SELECT METHOD=SRS N=100 OUT=MED_SAMPLE;
RUN;

/* HOW TO SELECT EVERY ALTERNATE RECORDS */

DATA MED_SELECT;
SET SSN5.MED_2023;
KEEP CUSTOMER_ID COMPANY GENDER AGE STATE_CODE SPENT_AMOUNT ROW_NUMBER;
ROW_NUMBER = _N_;
IF MOD(ROW_NUMBER, 2) = 0;
RUN;

DATA MED_SELECT;
SET SSN5.MED_2023;
KEEP CUSTOMER_ID COMPANY GENDER AGE STATE_CODE SPENT_AMOUNT ROW_NUMBER;
ROW_NUMBER = _N_;
IF MOD(ROW_NUMBER, 2) = 1;
```

```
RUN;

DATA MED_SELECT;
SET SSN5.MED_2023;
KEEP CUSTOMER_ID COMPANY GENDER AGE STATE_CODE SPENT_AMOUNT ROW_NUMBER;
ROW_NUMBER = _N_;
IF MOD(ROW_NUMBER, 5) = 0;
RUN;

/* HOW TO CREATE A DATASET BY TAKING ONLY 500 OBS */

DATA MED_SELECT;
SET SSN5.MED_2023 (OBS=500);
RUN;

/* SAS FIRST. AND LAST. */
===================================

DATA PROD_SALES;
INFILE CARDS DSD DLM='09'X;
INPUT PROD $ CITY : $15. SALES;
CARDS;
APPLE   BANGALORE   63000
APPLE   CHENNAI 27000
APPLE   DELHI   55600
APPLE   KOLKATA 60700
APPLE   MUMBAI  31100
DELL    BANGALORE   70200
DELL    CHENNAI 30600
DELL    DELHI   63600
DELL    KOLKATA 61000
DELL    MUMBAI  42900
LENOVO  BANGALORE   15800
LENOVO  CHENNAI 37600
LENOVO  DELHI   60600
LENOVO  KOLKATA 46000
LENOVO  MUMBAI  87500
;
RUN;

PROC PRINT DATA=PROD_SALES;
RUN;

/* SORT THE DATASET BEFORE APPLYING FIRST. AND LAST. */

PROC SORT DATA = PROD_SALES;
BY PROD;
RUN;

DATA PROD_SALES_V1;
SET PROD_SALES;
BY PROD;

IF FIRST.PROD THEN FIRST=1;
ELSE FIRST=0;

IF LAST.PROD THEN LAST=1;
ELSE LAST=0;

RUN;

PROC PRINT;
RUN;

DATA PROD_SALES_V2;
```

```
SET PROD_SALES;
BY PROD;

IF FIRST.PROD THEN FIRST=1;
ELSE FIRST=0;

IF LAST.PROD THEN LAST=1;
ELSE LAST=0;

IF FIRST.PROD THEN DO;
COUNT = 0;
CUMULATIVE_SUM = 0;
END;
COUNT+1;
CUMULATIVE_SUM + SALES;
RUN;

PROC PRINT;
RUN;

/* GET THE TOTAL NUMBER OF ROWS FOR EACH PRODUCT AND TOTAL SALES FOR EACH PRODUCT */

DATA PROD_SALES_V2;
SET PROD_SALES;
BY PROD;

IF FIRST.PROD THEN FIRST=1;
ELSE FIRST=0;

IF LAST.PROD THEN LAST=1;
ELSE LAST=0;

IF FIRST.PROD THEN DO;
COUNT = 0;
CUMULATIVE_SUM = 0;
END;
COUNT+1;
CUMULATIVE_SUM + SALES;

IF LAST.PROD;
DROP CITY SALES FIRST LAST;
RUN;

PROC PRINT;
RUN;

/* IN SAS SQL */

PROC SQL;
SELECT PROD,
COUNT(PROD) AS COUNT,
SUM(SALES) AS TOTAL_SALES
FROM PROD_SALES
GROUP BY PROD;
QUIT;

/* EXAMPLE */

DATA STU_SCORE;
INFILE CARDS DSD DLM='09'X;
INPUT STU_NAME $ SUBJECT  $ SCORE;
CARDS;
ALAM    EXCEL   81
ALAM    SQL 42
ALAM    PYTHON  84
ALAM    SAS 19
ALAM    TABLEAU 75
ALAM    POWER BI    47
KIRAN   EXCEL   13
KIRAN   SQL 29
KIRAN   PYTHON  69
KIRAN   SAS 38
KIRAN   TABLEAU 72
KIRAN   POWER BI    73
JINUS   EXCEL   19
JINUS   SQL 13
JINUS   PYTHON  71
JINUS   SAS 45
JINUS   TABLEAU 95
JINUS   POWER BI    41
;
RUN;

PROC PRINT DATA=STU_SCORE;
RUN;

PROC SORT DATA=STU_SCORE;
BY STU_NAME;
RUN;

/* GIVE ME STU_NAME, TOTAL SUBJECTS AND THE SUM OF MARKS SECURED */

DATA STU_SCORE_V1;
SET STU_SCORE;
BY STU_NAME;

IF FIRST.STU_NAME THEN DO;
COUNT=0;
CUMULATIVE_MARKS=0;
END;
COUNT+1;
CUMULATIVE_MARKS + SCORE;

IF LAST.STU_NAME;
DROP SUBJECT SCORE;
RUN;


PROC PRINT DATA=STU_SCORE_V1;
RUN;

/* EXAMPLE-  */


DATA PROD_SALES_ALL;
INFILE CARDS DSD DLM='09'X;
INPUT PROD $ CITY : $15. YEAR UNITS PRICE SALES;
CARDS;
APPLE   BANGALORE   2015    278 31284   8696952
DELL    BANGALORE   2015    321 43176   13859496
LENOVO  BANGALORE   2015    722 38210   27587620
ACER    BANGALORE   2015    430 43232   18589760
HP  BANGALORE   2015    856 37819   32373064
SAMSUNG BANGALORE   2015    761 36887   28071007
TOSHIBA BANGALORE   2015    766 38176   29242816
APPLE   BANGALORE   2016    484 31284   15141456
DELL    BANGALORE   2016    590 43176   25473840
LENOVO  BANGALORE   2016    812 38210   31026520
ACER    BANGALORE   2016    528 43232   22826496
HP  BANGALORE   2016    581 37819   21972839
```

```
SAMSUNG BANGALORE   2016    863 36887   31833481
TOSHIBA BANGALORE   2016    320 38176   12216320
APPLE   BANGALORE   2017    222 31284   6945048
DELL    BANGALORE   2017    209 43176   9023784
LENOVO  BANGALORE   2017    399 38210   15245790
ACER    BANGALORE   2017    188 43232   8127616
HP  BANGALORE   2017    524 37819   19817156
SAMSUNG BANGALORE   2017    556 36887   20509172
TOSHIBA BANGALORE   2017    398 38176   15194048
APPLE   BANGALORE   2018    521 31284   16298964
DELL    BANGALORE   2018    783 43176   33806808
LENOVO  BANGALORE   2018    553 38210   21130130
ACER    BANGALORE   2018    635 43232   27452320
HP  BANGALORE   2018    148 37819   5597212
SAMSUNG BANGALORE   2018    652 36887   24050324
TOSHIBA BANGALORE   2018    414 38176   15804864
APPLE   CHENNAI 2015    719 32353   23261807
DELL    CHENNAI 2015    577 41936   24197072
LENOVO  CHENNAI 2015    846 40836   34547256
ACER    CHENNAI 2015    436 42648   18594528
HP  CHENNAI 2015    410 48388   19839080
SAMSUNG CHENNAI 2015    770 38833   29901410
TOSHIBA CHENNAI 2015    222 32866   7296252
APPLE   CHENNAI 2016    387 32353   12520611
DELL    CHENNAI 2016    576 41936   24155136
LENOVO  CHENNAI 2016    472 40836   19274592
ACER    CHENNAI 2016    386 42648   16462128
HP  CHENNAI 2016    501 48388   24242388
SAMSUNG CHENNAI 2016    782 38833   30367406
TOSHIBA CHENNAI 2016    261 32866   8578026
APPLE   CHENNAI 2017    699 32353   22614747
DELL    CHENNAI 2017    253 41936   10609808
LENOVO  CHENNAI 2017    309 40836   12618324
ACER    CHENNAI 2017    877 42648   37402296
HP  CHENNAI 2017    391 48388   18919708
SAMSUNG CHENNAI 2017    128 38833   4970624
TOSHIBA CHENNAI 2017    528 32866   17353248
APPLE   CHENNAI 2018    245 32353   7926485
DELL    CHENNAI 2018    642 41936   26922912
LENOVO  CHENNAI 2018    248 40836   10127328
ACER    CHENNAI 2018    468 42648   19959264
HP  CHENNAI 2018    408 48388   19742304
SAMSUNG CHENNAI 2018    103 38833   3999799
TOSHIBA CHENNAI 2018    408 32866   13409328
APPLE   MUMBAI  2015    508 43413   22053804
DELL    MUMBAI  2015    218 38892   8478456
LENOVO  MUMBAI  2015    439 34544   15164816
ACER    MUMBAI  2015    119 46415   5523385
HP  MUMBAI  2015    795 35322   28080990
SAMSUNG MUMBAI  2015    696 30241   21047736
TOSHIBA MUMBAI  2015    238 33291   7923258
APPLE   MUMBAI  2016    371 43413   16106223
DELL    MUMBAI  2016    258 38892   10034136
LENOVO  MUMBAI  2016    815 34544   28153360
ACER    MUMBAI  2016    897 46415   41634255
HP  MUMBAI  2016    176 35322   6216672
SAMSUNG MUMBAI  2016    817 30241   24706897
TOSHIBA MUMBAI  2016    374 33291   12450834
APPLE   MUMBAI  2017    864 43413   37508832
DELL    MUMBAI  2017    415 38892   16140180
LENOVO  MUMBAI  2017    457 34544   15786608
ACER    MUMBAI  2017    898 46415   41680670
HP  MUMBAI  2017    429 35322   15153138
SAMSUNG MUMBAI  2017    766 30241   23164606
TOSHIBA MUMBAI  2017    465 33291   15480315
```

```
APPLE   MUMBAI  2018    140 43413   6077820
DELL    MUMBAI  2018    895 38892   34808340
LENOVO  MUMBAI  2018    867 34544   29949648
ACER    MUMBAI  2018    497 46415   23068255
HP  MUMBAI  2018    834 35322   29458548
SAMSUNG MUMBAI  2018    852 30241   25765332
TOSHIBA MUMBAI  2018    297 33291   9887427
APPLE   BHUBANESWAR 2015    119 49835   5930365
DELL    BHUBANESWAR 2015    325 35567   11559275
LENOVO  BHUBANESWAR 2015    635 36115   22933025
ACER    BHUBANESWAR 2015    107 44708   4783756
HP  BHUBANESWAR 2015    878 34491   30283098
SAMSUNG BHUBANESWAR 2015    405 30496   12350880
TOSHIBA BHUBANESWAR 2015    471 45199   21288729
APPLE   BHUBANESWAR 2016    482 49835   24020470
DELL    BHUBANESWAR 2016    472 35567   16787624
LENOVO  BHUBANESWAR 2016    854 36115   30842210
ACER    BHUBANESWAR 2016    598 44708   26735384
HP  BHUBANESWAR 2016    499 34491   17211009
SAMSUNG BHUBANESWAR 2016    826 30496   25189696
TOSHIBA BHUBANESWAR 2016    561 45199   25356639
APPLE   BHUBANESWAR 2017    556 49835   27708260
DELL    BHUBANESWAR 2017    421 35567   14973707
LENOVO  BHUBANESWAR 2017    897 36115   32395155
ACER    BHUBANESWAR 2017    178 44708   7958024
HP  BHUBANESWAR 2017    718 34491   24764538
SAMSUNG BHUBANESWAR 2017    758 30496   23115968
TOSHIBA BHUBANESWAR 2017    752 45199   33989648
APPLE   BHUBANESWAR 2018    104 49835   5182840
DELL    BHUBANESWAR 2018    390 35567   13871130
LENOVO  BHUBANESWAR 2018    428 36115   15457220
ACER    BHUBANESWAR 2018    121 44708   5409668
HP  BHUBANESWAR 2018    894 34491   30834954
SAMSUNG BHUBANESWAR 2018    439 30496   13387744
TOSHIBA BHUBANESWAR 2018    830 45199   37515170
;
RUN;
```

```
PROC PRINT DATA=PROD_SALES_ALL;
RUN;

/* SUMMARIZE THE DATA BY PRODUCT AND CITY-WISE SALES AMOUNT */
```

```
PROC SQL;
SELECT PROD, CITY,
SUM(UNITS) AS TOTAL_UNITS_SOLD,
SUM(SALES) AS TOTAL_SALES
FROM PROD_SALES_ALL
GROUP BY 1,2;
QUIT;

/* SAS FIRST. AND LAST.  */
```

```
PROC SORT DATA=PROD_SALES_ALL;
BY PROD CITY;
RUN;
```

```
DATA PROD_SALES_V1;
SET PROD_SALES_ALL;
BY PROD CITY;

IF FIRST.CITY THEN DO;
TOTAL_UNITS_SOLD =0;
TOTAL_SALES=0;
END;
```

```
TOTAL_UNITS_SOLD + UNITS;
TOTAL_SALES + SALES;

IF LAST.CITY;
DROP YEAR UNITS PRICE SALES;
RUN;

PROC PRINT;

/* CASE STUDY */

/* ASSUME THAT WE HAVE A STU_PERF DATASET HAVING 5 VARIABLES */

/* 1. STU_NAME */
/* 2. SUBJECT */
/* 3. COMPANY */
/* 4. YEAR */
/* 5. SCORE */

/* Q. GET STUDENT, COMPANY, YEAR-WISE SCORE TOTAL */

/* IN BASE SAS */

PROC SORT DATA=STU_PERF;
BY STU_NAME COMPANY YEAR;
RUN;

DATA STU_PERF_V1;
SET STU_PERF;
BY STU_NAME COMPANY YEAR;

IF FIRST.YEAR THEN DO;
TOTAL_SCORE=0;
END;

TOTAL_SCORE + SCORE;

IF LAST.YEAR;
DROP SUBJECT SCORE;
RUN;

PROC PRINT;
RUN;

/* PROC RANK */
==========================

DATA STU_SUB_SCORE;
INFILE CARDS DSD DLM='09'X;
INPUT STU_NAME : $15. SUBJECT : $15. SCORE;
CARDS;
BHAWNA  EXCEL   86
BHAWNA  VBA 41
BHAWNA  SQL 29
BHAWNA  SAS 61
BHAWNA  PYTHON  92
BHAWNA  ALTERYX 43
BHAWNA  TABLEAU 43
BHAWNA  POWER_BI    40
BHAWNA  ML  10
BHAWNA  ADV_ANALYTICS   31
DIBYA   EXCEL   60
DIBYA   VBA 25
DIBYA   SQL 36
DIBYA   SAS 97
DIBYA   PYTHON  44
DIBYA   ALTERYX 43
DIBYA   TABLEAU 17
DIBYA   POWER_BI    36
DIBYA   ML  99
DIBYA   ADV_ANALYTICS   82
AMOL    EXCEL   22
AMOL    VBA 18
AMOL    SQL 72
AMOL    SAS 68
AMOL    PYTHON  65
AMOL    ALTERYX 14
AMOL    TABLEAU 32
AMOL    POWER_BI    11
AMOL    ML  26
AMOL    ADV_ANALYTICS   37
;
RUN;

PROC PRINT DATA=STU_SUB_SCORE;
RUN;

/* SORT THE DATASET BEFORE APPLYING ANY RANKING */

PROC SORT DATA=STU_SUB_SCORE;
BY STU_NAME DESCENDING SCORE;
RUN;

/* USE OF FIRST. FOR RANKING */

DATA STU_SUB_SCORE_V1;
SET STU_SUB_SCORE;
BY STU_NAME;

IF FIRST.STU_NAME THEN DO;
RANK=0;
END;

RANK+1;
RUN;

PROC PRINT;
RUN;

/* USING PROC RANK */

PROC SORT DATA=STU_SUB_SCORE;
BY STU_NAME DESCENDING SCORE;
RUN;

PROC RANK DATA=STU_SUB_SCORE OUT=STU_SUB_SCORE_V2;
VAR SCORE;
RANKS RANKING_SCORE;
BY STU_NAME;
RUN;

PROC PRINT DATA=STU_SUB_SCORE_V2;
RUN;

PROC RANK DATA=STU_SUB_SCORE DESCENDING OUT=STU_SUB_SCORE_V3;
VAR SCORE;
RANKS RANKING_SCORE;
BY STU_NAME;
RUN;
```

```
PROC PRINT DATA=STU_SUB_SCORE_V3;
RUN;

PROC RANK DATA=STU_SUB_SCORE DESCENDING OUT=STU_SUB_SCORE_V3 TIES=DENSE;
VAR SCORE;
RANKS RANKING_SCORE;
BY STU_NAME;
RUN;

PROC PRINT DATA=STU_SUB_SCORE_V3;
RUN;

/* SAS LOOPS */
====================================

/* LOOPS - ITERATIVE STATEMENTS THAT ARE EXECUTED UNTIL ANY GIVEN CONDITION IS MET */

/* PRINT 1 TO 100 */

DATA ALL_NUM;
DO NUM=1 TO 100;
OUTPUT;
END;
RUN;

PROC PRINT;
RUN;

DATA ALL_NUM;
DO NUM=1 TO 100 BY 10;
OUTPUT;
END;
RUN;

PROC PRINT;
RUN;

DATA ALL_NUM;
DO NUM=0 TO 100 BY 10;
OUTPUT;
END;
RUN;

PROC PRINT;
RUN;

DATA ALL_NUM;
DO NUM=100 TO 0 BY -1;
OUTPUT;
END;
RUN;

PROC PRINT;
RUN;

DATA ALL_NUM;
DO NUM=100 TO 0 BY -1;
OUTPUT;
END;
RUN;

PROC PRINT;
RUN;
```

```
/* BACKWARD LOOPING */

/* CHARACTER VALUE LOOP */
```

```
DATA STU_DETAILS;
DO NAME='AYASHKANT', 'SARTHAK', 'SHEKHAR', 'ADITYA';
OUTPUT;
END;
RUN;
```

```
PROC PRINT;
RUN;
```

```
DATA STU_DETAILS1;
DO NAME="AYASHKANT", "SARTHAK", "SHEKHAR", "ADITYA";
OUTPUT;
END;
RUN;
```

```
PROC PRINT;
RUN;

/* MITRABHANU - 12LPA , 80K SALARY PER MONTH */
/* 50K IS THE SAVINGS. INVEST YOUR MONEY IN SOME FINANCIAL INSTRUMENT*/

/* 50*12 = 6L */
```

```
DATA INVESTMENT;
CAPITAL = 600000;
INT= 0.10;
DO YEAR=1 TO 20;
CAPITAL + (CAPITAL*INT);
OUTPUT;
END;
RUN;
```

```
PROC PRINT;
RUN;
```

```
DATA INVESTMENT2;
INT= 0.10;
DO YEAR=1 TO 20;
CAPITAL+600000;
CAPITAL + (CAPITAL*INT);
OUTPUT;
END;
RUN;
```

```
PROC PRINT;
RUN;

/* BREAKING OF LOOP */
```

```
DATA INVESTMENT2;
DO YEAR=1 TO 20 UNTIL (CAPITAL >=30000000);
CAPITAL+600000;
CAPITAL + (CAPITAL*0.1);
OUTPUT;
END;
RUN;
```

```
PROC PRINT;
RUN;
```

```
DATA INVESTMENT3;
DO WHILE (CAPITAL <= 30000000);
YEAR+1;
CAPITAL+600000;
CAPITAL + (CAPITAL*0.10);
OUTPUT;
END;
RUN;
```

```
PROC PRINT;
RUN;

/* SAS ARRAYS */
=====================
```

```
DATA MED_STORE;
INFILE CARDS DSD DLM='09'X;
INPUT CUSTOMER_ID Company $ GENDER $ Age STATE_CODE $ NO_OF_TRIPS Spent_Amount;
CARDS;
1000032547  MED +   Female  29  QLD 4   69.37
1000032548  DR.REDDYS   Female  39  WA  17  498.5
1000032549  RELEGARE    Female  54  NSW 1   1.56
1000032550  GSK Male        VIC     160.98
1000032551  GSK Male        VIC     289.1
1000032552  APPOLO  Female      NSW     690.45
1000032553      Female  26  VIC     160.89
1000032554      Male    26  QLD 8   253.23
1000032555      Male    55  NSW 1   69.77
1000032556  MED +   Female  61      2   232.31
1000032557  GSK Female  28      1   274.66
1000032558  MED +       57      1   42.83
1000032559  GENO        34  VIC 1
1000032560  DR.REDDYS       28  NSW 1
1000032561  RELEGARE        51  QLD 3
1000032562  GUARDIAN    Female  61  VIC 2
1000032563  RELEGARE    Female      QLD 2
1000032564  GUARDIAN    Female      NSW 4   83.87
1000032565  GENO    Female      NSW 2   92.28
1000032566  RANBAXY Male    64  NSW 2   249.46
1000032567  MED +   Male    41  NSW 6   251.45
;
RUN;
```

```
PROC PRINT DATA=MED_STORE;
RUN;

/* REQUIREMENT */
/* 1. REPLACE ALL CHARACTER VARIABLE MISSING FIELDS AS "NA" */
/* 2. REPLACE ALL NUMERIC VARIABLE MISSING FIELDS AS 0 */
```

```
DATA MED_STORE;
SET MED_STORE;

/* ALL NUMERIC */
ARRAY NUM _NUMERIC_;
DO OVER NUM;
IF NUM=. THEN NUM=0;
END;

/* ALL CHARACTERS */
ARRAY CHAR _CHARACTER_;
DO OVER CHAR;
IF CHAR=' ' THEN CHAR='NA';
END;
```

```
RUN;

PROC PRINT; RUN;


/* MATHEMATICAL CALCULATIONS USING AN ARRAY */

DATA PROD_SALES;
INFILE CARDS DSD DLM='09'X;
INPUT PRODUCT $ MONTH1 MONTH2 MONTH3 MONTH4 MONTH5 MONTH6;
CARDS;
APPLE   297 296 201 267 442 462
DELL    303 823 275 645 640 811
HP  207 393 517 236 589 869
LENOVO  233 494 852 675 282 334
ACER    348 690 879 843 233 654
;
RUN;

PROC PRINT DATA=PROD_SALES;
RUN;

DATA PROD_SALES_V1;
SET PROD_SALES;
MONTH1 = MONTH1 + 100;
MONTH2 = MONTH2 + 100;
MONTH3 = MONTH3 + 100;
MONTH4 = MONTH4 + 100;
MONTH5 = MONTH5 + 100;
MONTH6 = MONTH6 + 100;
RUN;

/* USING ARRAY */

DATA PROD_SALES_V2;
SET PROD_SALES;
ARRAY SALES{*} _NUMERIC_;
DO I=1 TO DIM(SALES);
SALES(I) = SALES(I) + 100;
END;
DROP I;
RUN;

PROC PRINT;
RUN;

DATA PROD_SALES_V2;
SET PROD_SALES;
ARRAY SALES{6} _NUMERIC_;
DO I=1 TO 6;
SALES(I) = SALES(I) + 100;
END;
DROP I;
RUN;

PROC PRINT;
RUN;

WE HAVE COMPLETED SESSION-5
```