MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Deccan Education Society’s

Navinchandra Mehta Institute of

Technology and Development

C E R T I F I C A T E

This is to certify that Mr. Siddhesh Santosh Mhatre of M.C. A.

Semester I with Roll No.C-23071 has completed All practicals of

Advance Database Management System under my supervision in this

college during the year 2023 -2024.

R1 R2 R3 R4

CO (Journal) (Performance (Implementation (Mock Attendance

during lab using different Viva)

session) problem solving

techniques)

CO1

CO2

CO3

CO4

Practical-in-charge Head of Department

MCA Department

(NMITD)

1

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

MCAL13 ADBMS Lab INDEX

Sr.No Title CO Date Sign

1 Implementation of Partitions: Range, List. CO1

Self-Learning Topics : Hash Partition, 14/09/2023

Composite partition

2 Analytical Queries

Roll\_Up, CUBE, First, Last, Lead, Lag, CO3 25/10/2023

Rank and Dense Rank 09/10/2023

Self-Learning Topics: Cume\_list,

Percent\_rank

3 Implementation of,

• Abstract Data Type CO1 11/10/2023

• Reference 25/10/2023

Self-Learning Topics: Nested ADT,

Inheritance

4 ETL Transformation with Pentaho

1. Copy data from Source & store to

Target CO2 22/11/2023

2. Adding Sequence 22/11/2023

3. Adding Calculator

4. Concatenation of Two Fields

5. Splitting of Two Fields

6. Number Range

7. String Operations

8. Sorting Data

9. Implement the Merge Join

10. Implement data validations on

table data

11. Replace Strings

12. Splitting Fields to Rows

5 Introduction to R,

Install packages CO4

Loading packages

Data types, checking variable 11/12/2023

type,printing variable and objects 12/12/2023

(Vector, Matrix, List, Factor, Data frame,

Table)

c-binding and rbinding

Reading and Writing data:

Setw(), getw(), data(),rm()

Attaching and Detaching data

Reading data from the console

Loading data from different data

sources(CSV,Excel)

2

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

6 Data preprocessing techniques in R CO4

Naming and Renaming variables 15/12/2023

Adding a new variables

Dealing with missing value

Dealing with categorical data

Data reduction using subsetting

7 Implementation and analysis of Linear CO4

regression through graphical methods. 18/12/2023

8 Implementation and Analysis

Classification algorithms like Naïve CO4 08/12/2023

Bayesiam, K-Nearest Neighbour, ID3,

C4.5

9 Implementation and analysis of Apriori CO4

Algorithm using Market Basket Analysis 20/12/2023

10 Implementation and analysis of clustering CO4

algorithms like K-means, Agglomarative 20/12/2023

3

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Practical No.: 01

Aim: Implementation of different types of Partitions

• RANGE PARTITIONING

SQL> CREATE TABLE sales\_range01

(salesman\_id NUMBER(5),

salesman\_name VARCHAR2(30),

sales\_amount NUMBER(10),

sales\_date DATE)

PARTITION BY RANGE(sales\_date)

(

PARTITION sales\_jan2000 VALUES LESS

THAN(TO\_DATE('01/02/2000','DD/MM/YYYY')),

PARTITION sales\_feb2000 VALUES LESS

THAN(TO\_DATE('01/03/2000','DD/MM/YYYY')),

PARTITION sales\_mar2000 VALUES LESS

THAN(TO\_DATE('01/04/2000','DD/MM/YYYY')),

PARTITION sales\_apr2000 VALUES LESS

THAN(TO\_DATE('01/05/2000','DD/MM/YYYY'))

)

;

Table created.

SQL> SELECT TABLE\_NAME, PARTITION\_NAME FROM USER\_TAB\_PARTITIONS

WHERE TABLESPACE\_NAME='USERS';

TABLE\_NAME PARTITION\_NAME

------------------------------ ------------------------------

SALES\_RANGE01 SALES\_JAN2000

SALES\_RANGE01 SALES\_FEB2000

SALES\_RANGE01 SALES\_MAR2000

SALES\_RANGE01 SALES\_APR2000

4

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

SQL> insert into sales\_range01 values(1,'John

Smith',5000,TO\_DATE('23/02/2000','DD/MM/YYYY'));

1 row created.

SQL> SELECT\*FROM SALES\_RANGE01;

SALESMAN\_ID SALESMAN\_NAME SALES\_AMOUNT SALES\_DAT

----------- ------------------------------ ------------ ---------

1 John Smith 5000 23-FEB-00

SQL> insert into sales\_range01 values(2,'Tejas

Patil',8000,TO\_DATE('23/02/2000','DD/MM/YYYY'));

1 row created.

SQL> SELECT\*FROM SALES\_RANGE01;

SALESMAN\_ID SALESMAN\_NAME SALES\_AMOUNT SALES\_DAT

----------- ------------------------------ ------------ ---------

1 John Smith 5000 23-FEB-00

2 Tejas Patil 8000 23-FEB-00

SQL> insert into sales\_range01 values(3,'Yogesh

Rahate',4265,TO\_DATE('18/01/2000','DD/MM/YYYY'));

1 row created.

SQL> insert into sales\_range01 values(4,'Shahu

Patil',8500,TO\_DATE('09/01/2000','DD/MM/YYYY'));

1 row created.

5

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

SQL> SELECT\*FROM SALES\_RANGE01;

SALESMAN\_ID SALESMAN\_NAME SALES\_AMOUNT SALES\_DAT

----------- ------------------------------ ------------ ---------

3 Yogesh Rahate 4265 18-JAN-00

4 Shahu Patil 8500 09-JAN-00

1 John Smith 5000 23-FEB-00

2 Tejas Patil 8000 23-FEB-00

SQL> insert into sales\_range01 values(5,'Snehal

Borji',10000,TO\_DATE('04/03/2000','DD/MM/YYYY'));

1 row created.

SQL> insert into sales\_range01 values(6,'Gaurav

Patil',6500,TO\_DATE('14/03/2000','DD/MM/YYYY'));

1 row created.

SQL> SELECT\*FROM SALES\_RANGE01;

SALESMAN\_ID SALESMAN\_NAME SALES\_AMOUNT SALES\_DAT

----------- ------------------------------ ------------ ---------

3 Yogesh Rahate 4265 18-JAN-00

4 Shahu Patil 8500 09-JAN-00

1 John Smith 5000 23-FEB-00

2 Tejas Patil 8000 23-FEB-00

5 Snehal Borji 10000 04-MAR-00

6 Gaurav Patil 6500 14-MAR-00

6 rows selected.

SQL> insert into sales\_range01 values(7,'Sahil

Jadhav',13000,TO\_DATE('08/04/2000','DD/MM/YYYY'));

6

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

1 row created.

SQL> insert into sales\_range01 values(8,'Shruti

Bhuvad',9000,TO\_DATE('16/04/2000','DD/MM/YYYY'));

1 row created.

SQL> SELECT\*FROM SALES\_RANGE01;

SALESMAN\_ID SALESMAN\_NAME SALES\_AMOUNT SALES\_DAT

----------- ------------------------------ ------------ ---------

3 Yogesh Rahate 4265 18-JAN-00

4 Shahu Patil 8500 09-JAN-00

1 John Smith 5000 23-FEB-00

2 Tejas Patil 8000 23-FEB-00

5 Snehal Borji 10000 04-MAR-00

6 Gaurav Patil 6500 14-MAR-00

7 Sahil Jadhav 13000 08-APR-00

8 Shruti Bhuvad 9000 16-APR-00

8 rows selected.

SQL> SELECT\*FROM SALES\_RANGE01 PARTITION(sales\_feb2000);

SALESMAN\_ID SALESMAN\_NAME SALES\_AMOUNT SALES\_DAT

----------- ------------------------------ ------------ ---------

1 John Smith 5000 23-FEB-00

2 Tejas Patil 8000 23-FEB-00

SQL> SELECT\*FROM SALES\_RANGE01 PARTITION(sales\_jan2000);

SALESMAN\_ID SALESMAN\_NAME SALES\_AMOUNT SALES\_DAT

----------- ------------------------------ ------------ ---------

3 Yogesh Rahate 4265 18-JAN-00

4 Shahu Patil 8500 09-JAN-00

7

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

SQL> SELECT\*FROM SALES\_RANGE01 PARTITION(sales\_mar2000);

SALESMAN\_ID SALESMAN\_NAME SALES\_AMOUNT SALES\_DAT

----------- ------------------------------ ------------ ---------

5 Snehal Borji 10000 04-MAR-00

6 Gaurav Patil 6500 14-MAR-00

SQL> SELECT\*FROM SALES\_RANGE01 PARTITION(sales\_apr2000);

SALESMAN\_ID SALESMAN\_NAME SALES\_AMOUNT SALES\_DAT

----------- ------------------------------ ------------ ---------

7 Sahil Jadhav 13000 08-APR-00

8 Shruti Bhuvad 9000 16-APR-00

8

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

• LIST PARTITIONING

CREATE TABLE sales\_list

(salesman\_id NUMBER(5),

salesman\_name VARCHAR2(30),

sales\_state VARCHAR2(20),

sales\_amount NUMBER(10),

sales\_date DATE)

PARTITION BY LIST(sales\_state)

(

PARTITION sales\_west VALUES('Andheri','Boriwali'),

PARTITION sales\_Harbur VALUES('Vashi','Chembur','Panvel'),

PARTITION sales\_central VALUES('Thane','Kalyan'),

PARTITION sales\_other VALUES(DEFAULT)

)

enable row movement

;

Table created.

SQL> SELECT TABLE\_NAME,PARTITION\_NAME FROM USER\_TAB\_PARTITIONS

WHERE TABLESPACE\_NAME='USERS';

TABLE\_NAME PARTITION\_NAME

------------------------------ ------------------------------

SALES\_LIST SALES\_WEST

SALES\_LIST SALES\_HARBUR

SALES\_LIST SALES\_CENTRAL

SALES\_LIST SALES\_OTHER

SQL> insert into sales\_list values(1,'John

Smith','Andheri',2300,TO\_DATE('23/02/2000','DD/MM/YYYY')

2 );

9

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

1 row created.

SQL> insert into sales\_list

values(2,'Alis','Panvel',2300,TO\_DATE('20/04/2004','DD/MM/YYYY'));

1 row created.

SQL> insert into sales\_list

values(3,'Sahil','Thane',4000,TO\_DATE('09/11/2001','DD/MM/YYYY'));

1 row created.

SQL> insert into sales\_list

values(4,'Tejas','Thane',2000,TO\_DATE('05/09/2000','DD/MM/YYYY'));

1 row created.

SQL> insert into sales\_list

values(5,'Shahu','Chembur',1000,TO\_DATE('16/12/2002','DD/MM/YYYY'));

1 row created.

SQL> insert into sales\_list

values(6,'Sakshi','Boriwali',5600,TO\_DATE('04/10/2001','DD/MM/YYYY'));

1 row created.

SQL> insert into sales\_list

values(7,'Snehal','Kalyan',2980,TO\_DATE('30/05/2003','DD/MM/YYYY'));

1 row created.

SQL> insert into sales\_list

10

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

values(8,'Mitesh','Vashi',3000,TO\_DATE('14/06/2001','DD/MM/YYYY'));

1 row created.

SQL> SELECT\*FROM sales\_list;

SALESMAN\_ID SALESMAN\_NAME SALES\_STATE SALES\_AMOUNT SALES\_DAT

----------- ------------------------------ -------------------- ------------ ---------

1 John Smith Andheri 2300 23-FEB-00

6 Sakshi Boriwali 5600 04-OCT-01

2 Alis Panvel 2300 20-APR-04

5 Shahu Chembur 1000 16-DEC-02

8 Shahu Vashi 3000 14-JUN-01

3 Sahil Thane 4000 09-NOV-01

4 Tejas Thane 2000 05-SEP-00

7 Snehal Kalyan 2980 30-MAY-03

8 rows selected.

SELECT\*FROM sales\_list partition(sales\_west);

SALESMAN\_ID SALESMAN\_NAME SALES\_STATE SALES\_AMOUNT SALES\_DAT

----------- ------------------------------ -------------------- ------------ ---------

1 John Smith Andheri 2300 23-FEB-00

6 Sakshi Boriwali 5600 04-OCT-01

SELECT\*FROM sales\_list partition(sales\_harbur);

SALESMAN\_ID SALESMAN\_NAME SALES\_STATE SALES\_AMOUNT SALES\_DAT

----------- ------------------------------ -------------------- ------------ ---------

2 Alis Panvel 2300 20-APR-04

5 Shahu Chembur 1000 16-DEC-02

8 Shahu Vashi 3000 14-JUN-01

11

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

SQL> SELECT\*FROM sales\_list partition(sales\_central);

SALESMAN\_ID SALESMAN\_NAME SALES\_STATE SALES\_AMOUNT SALES\_DAT

----------- ------------------------------ -------------------- ------------ ---------

3 Sahil Thane 4000 09-NOV-01

4 Tejas Thane 2000 05-SEP-00

7 Snehal Kalyan 2980 30-MAY-03

12

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

• HASH PARTITIONING

SQL> CREATE TABLE sales\_hash

2 (salesman\_id NUMBER(5),

3 salesman\_name VARCHAR2(30),

4 sales\_amount NUMBER(10),

5 week\_no NUMBER(2))

6 PARTITION BY HASH(salesman\_id)

7 PARTITIONS 4;

Table created.

SELECT TABLE\_NAME,PARTITION\_NAME FROM USER\_TAB\_PARTITIONS

WHERE TABLESPACE\_NAME='USERS';

TABLE\_NAME PARTITION\_NAME

------------------------------ ------------------------------

SALES\_LIST SALES\_WEST

SALES\_LIST SALES\_HARBUR

SALES\_LIST SALES\_CENTRAL

SALES\_LIST SALES\_OTHER

SALES\_HASH SYS\_P21

SALES\_HASH SYS\_P22

SALES\_HASH SYS\_P23

SALES\_HASH SYS\_P24

8 rows selected.

insert into sales\_hash

2 values(101,'john',44000,12);

1 row created.

SQL> insert into sales\_hash

2 values(102,'alis',54000,18);

1 row created.

13

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

SQL> insert into sales\_hash

2 values(103,'Tejas',15000,6);

1 row created.

SQL> insert into sales\_hash

2 values(104,'shahu',3000,22);

1 row created.

SQL> insert into sales\_hash

2 values(105,'sakshi',4522,32);

1 row created.

SQL> insert into sales\_hash

2 values(106,'sahil',9000,15);

1 row created.

SQL> insert into sales\_hash

2 values(107,'shruti',6500,13);

1 row created.

SQL> insert into sales\_hash

2 values(108,'mitesh',13000,28);

1 row created.

SQL> Select\*from sales\_hash;

14

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

SALESMAN\_ID SALESMAN\_NAME SALES\_AMOUNT WEEK\_NO

----------- ------------------------------ ------------ ----------

108 Shahu 13000 28

104 shahu 3000 22

102 alis 54000 18

103 Tejas 15000 6

105 sakshi 4522 32

107 shruti 6500 13

101 john 44000 12

106 sahil 9000 15

8 rows selected.

SQL> select\*from sales\_hash partition(sys\_p21);

SALESMAN\_ID SALESMAN\_NAME SALES\_AMOUNT WEEK\_NO

----------- ------------------------------ ------------ ----------

108 Shahu 13000 28

SQL> select\*from sales\_hash partition(sys\_p22);

SALESMAN\_ID SALESMAN\_NAME SALES\_AMOUNT WEEK\_NO

----------- ------------------------------ ------------ ----------

104 shahu 3000 22

SQL> select\*from sales\_hash partition(sys\_p23);

SALESMAN\_ID SALESMAN\_NAME SALES\_AMOUNT WEEK\_NO

----------- ------------------------------ ------------ ----------

102 alis 54000 18

103 Tejas 15000 6

105 sakshi 4522 32

107 shruti 6500 13

15

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

SQL> select\*from sales\_hash partition(sys\_p24);

SALESMAN\_ID SALESMAN\_NAME SALES\_AMOUNT WEEK\_NO

----------- ------------------------------ ------------ ----------

101 john 44000 12

106 sahil 9000 15

SQL> commit;

Commit complete.

16

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

COMPOSITE PARTITIONING: (Range-Hash)

SQL> CREATE TABLE orders1(

2 ord NUMBER,

3 orderdate DATE,

4 prod NUMBER,

5 quantity NUMBER)

6 PARTITION BY RANGE(orderdate)

7 SUBPARTITION BY HASH(prod)SUBPARTITIONS 4

8 (PARTITION q1 VALUES LESS THAN('01-APR-2009'),

9 PARTITION q2 VALUES LESS THAN('01-JUL-2009'),

10 PARTITION q3 VALUES LESS THAN('01-OCT-2009'),

11 PARTITION q4 VALUES LESS THAN(MAXVALUE)

12 );

Table created.

SQL> SELECT TABLE\_NAME,PARTITION\_NAME FROM USER\_TAB\_PARTITIONS

WHERE TABLESPACE\_NAME='USERS';

TABLE\_NAME PARTITION\_NAME

------------------------------ ------------------------------

ORDERS1 Q1

ORDERS1 Q2

ORDERS1 Q3

ORDERS1 Q4

4 rows selected.

SQL> insert into orders1 values(1,'23-mar-2009',101,200);

1 row created.

SQL> insert into orders1 values(3,'12-may-2009',101,200);

1 row created.

17

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

SQL> insert into orders1 values(3,'18-aug-2009',101,200);

1 row created.

SQL> insert into orders1 values(3,'18-aug-2010',111,400);

1 row created.

SQL> insert into orders1 values(3,'18-sep-2009',121,300);

1 row created.

SQL> select\*from orders1;

ORD ORDERDATE PROD QUANTITY

---------- --------- ---------- ----------

1 23-MAR-09 101 200

3 12-MAY-09 101 200

3 18-AUG-09 101 200

3 18-SEP-09 121 300

3 18-AUG-10 111 400

SQL> select\*from orders1 PARTITION(q1);

ORD ORDERDATE PROD QUANTITY

---------- --------- ---------- ----------

1 23-MAR-09 101 200

SQL> select\*from orders1 PARTITION(q2);

ORD ORDERDATE PROD QUANTITY

---------- --------- ---------- ----------

3 12-MAY-09 101 200

18

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

SQL> select\*from orders1 PARTITION(q3);

ORD ORDERDATE PROD QUANTITY

---------- --------- ---------- ----------

3 18-AUG-09 101 200

3 18-SEP-09 121 300

SQL> select\*from orders1 PARTITION(q4);

ORD ORDERDATE PROD QUANTITY

---------- --------- ---------- ----------

3 18-AUG-10 111 400

19

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Practical No.: 02

Aim: Implementation of Analytical Queries

SQL> CREATE TABLE employee(

2 emp\_no integer primary key,

3 dep\_no integer,

4 bdate date,

5 sal integer,

6 comm integer,

7 job varchar2(10)

8 );

Table created.

SQL> insert all

2 into employee values (101,10,'12-JAN-82',22000,1000,'clerk')

3 into employee values (102,10,'13-FEB-83',33000,2000,'clerk')

4 into employee values (103,10,'14-MAR-84',44000,200,'clerk')

5 into employee values (104,20,'15-APR-87',55000,3000,'manager')

6 into employee values (105,20,'14-JUN-82',38000,4500,'manager')

7 into employee values (106,20,'15-AUG-88',44000,500,'manager')

8 into employee values (107,10,'31-DEC-81',58000,8000,'manager')

9 into employee values (108,20,'25-JUL-88',49000,700,'clerk')

10 SELECT\*FROM DUAL;

8 rows created.

SQL> SELECT\*FROM EMPLOYEE;

EMP\_NO DEP\_NO BDATE SAL COMM JOB

---------- ---------- --------- ---------- ---------- ----------

101 10 12-JAN-82 22000 1000 clerk

102 10 13-FEB-83 33000 2000 clerk

103 10 14-MAR-84 44000 200 clerk

104 20 15-APR-87 55000 3000 manager

20

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

105 20 14-JUN-82 38000 4500 manager

106 20 15-AUG-88 44000 500 manager

107 10 31-DEC-81 58000 8000 manager

108 20 25-JUL-88 49000 700 clerk

8 rows selected.

Rollup

SQL> SELECT dep\_no,job,count(\*),sum(sal)

2 from employee

3 group by rollup(dep\_no,job);

DEP\_NO JOB COUNT(\*) SUM(SAL)

---------- ---------- ---------- ----------

10 clerk 3 99000

20 manager 3 137000

10 manager 1 58000

20 clerk 1 49000

10 4 157000

20 4 186000

8 343000

7 rows selected.

SQL> select\*from employee;

EMP\_NO DEP\_NO BDATE SAL COMM JOB

---------- ---------- --------- ---------- ---------- ----------

101 10 12-JAN-82 22000 1000 clerk

102 10 13-FEB-83 33000 2000 clerk

103 10 14-MAR-84 44000 200 clerk

104 20 15-APR-87 55000 3000 manager

105 20 14-JUN-82 38000 4500 manager

106 20 15-AUG-88 44000 500 manager

107 10 31-DEC-81 58000 8000 manager

21

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

108 20 25-JUL-88 49000 700 clerk

8 rows selected.

SQL> select dep\_no,job,sum(sal)

2 from employee

3 where dep\_no in(10,20)

4 group by dep\_no ,rollup(job);

DEP\_NO JOB SUM(SAL)

---------- ---------- ----------

10 clerk 99000

10 manager 58000

10 157000

20 clerk 49000

20 manager 137000

20 186000

6 rows selected.

SQL> SELECT dep\_no,job,count(\*),sum(sal)

2 from employee

3 group by job,rollup(dep\_no);

DEP\_NO JOB COUNT(\*) SUM(SAL)

---------- ---------- ---------- ----------

10 clerk 3 99000

20 clerk 1 49000

clerk 4 148000

10 manager 1 58000

20 manager 3 137000

manager 4 195000

6 rows selected.

22

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

CUBE

SQL> SELECT dep\_no,job,count(\*),sum(sal)

2 from employee

3 group by cube(dep\_no,job);

DEP\_NO JOB COUNT(\*) SUM(SAL)

---------- ---------- ---------- ----------

8 343000

clerk 4 148000

manager 4 195000

10 4 157000

10 clerk 3 99000

10 manager 1 58000

20 4 186000

20 clerk 1 49000

20 manager 3 137000

9 rows selected.

SQL> select\*from employee;

EMP\_NO DEP\_NO BDATE SAL COMM JOB

---------- ---------- --------- ---------- ---------- ----------

101 10 12-JAN-82 22000 1000 clerk

102 10 13-FEB-83 33000 2000 clerk

103 10 14-MAR-84 44000 200 clerk

104 20 15-APR-87 55000 3000 manager

105 20 14-JUN-82 38000 4500 manager

106 20 15-AUG-88 44000 500 manager

107 10 31-DEC-81 58000 8000 manager

108 20 25-JUL-88 49000 700 clerk

8 rows selected.

23

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

RANK

SQL> select emp\_no,dep\_no,sal,comm,

2 rank()over(partition by dep\_no order by sal)as Rank

3 from employee;

EMP\_NO DEP\_NO SAL COMM RANK

---------- ---------- ---------- ---------- ----------

101 10 22000 1000 1

102 10 33000 2000 2

103 10 44000 200 3

107 10 58000 8000 4

105 20 38000 4500 1

106 20 44000 500 2

108 20 49000 700 3

104 20 55000 3000 4

8 rows selected.

DENSE RANK

SQL> select emp\_no,dep\_no,sal,comm,

2 dense\_rank() over(partition by dep\_no order by sal)as Rank

3 from employee;

EMP\_NO DEP\_NO SAL COMM RANK

---------- ---------- ---------- ---------- ----------

101 10 22000 1000 1

102 10 33000 2000 2

103 10 44000 200 3

107 10 58000 8000 4

105 20 38000 4500 1

106 20 44000 500 2

108 20 49000 700 3

104 20 55000 3000 4

24

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

8 rows selected.

LEAD

SQL> select emp\_no,bdate,

2 lead(bdate,1)over(order by bdate) as "next"

3 from employee;

EMP\_NO BDATE next

---------- --------- ---------

107 31-DEC-81 12-JAN-82

101 12-JAN-82 14-JUN-82

105 14-JUN-82 13-FEB-83

102 13-FEB-83 14-MAR-84

103 14-MAR-84 15-APR-87

104 15-APR-87 25-JUL-88

108 25-JUL-88 15-AUG-88

106 15-AUG-88

8 rows selected.

LAG

SQL> select emp\_no,bdate,

2 lag(bdate,1)over(order by bdate) as "Previous"

3 from employee;

EMP\_NO BDATE Previous

---------- --------- ---------

107 31-DEC-81

101 12-JAN-82 31-DEC-81

105 14-JUN-82 12-JAN-82

102 13-FEB-83 14-JUN-82

103 14-MAR-84 13-FEB-83

104 15-APR-87 14-MAR-84

25

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

108 25-JUL-88 15-APR-87

106 15-AUG-88 25-JUL-88

8 rows selected.

SQL> select\*from employee;

EMP\_NO DEP\_NO BDATE SAL COMM JOB

---------- ---------- --------- ---------- ---------- ----------

101 10 12-JAN-82 22000 1000 clerk

102 10 13-FEB-83 33000 2000 clerk

103 10 14-MAR-84 44000 200 clerk

104 20 15-APR-87 55000 3000 manager

105 20 14-JUN-82 38000 4500 manager

106 20 15-AUG-88 44000 500 manager

107 10 31-DEC-81 58000 8000 manager

108 20 25-JUL-88 49000 700 clerk

8 rows selected.

FIRST

SQL> select dep\_no,sal,

2 max(sal)keep(DENSE\_RANK FIRST ORDER BY sal desc)

3 over(PARTITION BY dep\_no)"max"

4 from employee;

DEP\_NO SAL max

---------- ---------- ----------

10 58000 58000

10 44000 58000

10 33000 58000

10 22000 58000

20 44000 55000

26

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

20 38000 55000

20 49000 55000

20 55000 55000

8 rows selected.

LAST

SQL> select dep\_no,sal,

2 min(sal)keep(DENSE\_RANK LAST ORDER BY sal desc)

3 over(PARTITION BY dep\_no)"min"

4 from employee;

DEP\_NO SAL min

---------- ---------- ----------

10 58000 22000

10 44000 22000

10 33000 22000

10 22000 22000

20 44000 38000

20 38000 38000

20 49000 38000

20 55000 38000

8 rows selected.

SQL> select dep\_no,sal,

2 max(sal)keep(DENSE\_RANK FIRST ORDER BY sal desc)

3 over(PARTITION BY dep\_no)"max",

4 max(sal)keep(DENSE\_RANK LAST ORDER BY sal desc)

5 over(PARTITION BY dep\_no)"min"

6 from employee;

DEP\_NO SAL max min

---------- ---------- ---------- ----------

10 58000 58000 22000

27

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

10 44000 58000 22000

10 33000 58000 22000

10 22000 58000 22000

20 44000 55000 38000

20 38000 55000 38000

20 49000 55000 38000

20 55000 55000 38000

8 rows selected.

28

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Practical No.: 03

Aim: Implementation of ORDBMS concepts

SQL> create type type\_name As object

2 (

3 fname varchar(20),

4 mname varchar(20),

5 lname varchar(20)

6 );

7 /

Type created.

SQL> create type type\_address As object

2 (

3 street varchar(20),

4 city varchar(20),

5 pincode number(10)

6 );

7 /

Type created.

SQL> create table customer1

2 (

3 c\_id number(5) primary key,

4 c\_name type\_name,

5 c\_add type\_address,

6 c\_phno number(10)

7 );

Table created.

SQL> insert into customer1

2 values(1,type\_name('varsha','s','atul'),

29

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

3 type\_address('Sainagar','Mumbai',400042),123456789);

1 row created.

SQL> insert into customer1

2 values(2,type\_name('Tejas','ganesh','agre'),

3 type\_address('ganesh gawade','Mumbai',400080),9867820182);

1 row created.

SQL> insert into customer1

2 values(3,type\_name('shruti','sunil','bhuvad'),

3 type\_address('tilak','Pune',400078),9965824563);

1 row created.

SQL> insert into customer1

2 values(4,type\_name('mitesh','rajendra','roge'),

3 type\_address('nehru','Pune',400085),9945238522);

1 row created.

SQL> insert into customer1

2 values(5,type\_name('kanase','dagdu','sakshi'),

3 type\_address('vishnu','satara',400012),8856489621);

1 row created.

SQL> insert into customer1

2 values(6,type\_name('sahil','jagannath','mahale'),

3 type\_address('savarkar','mumbai',400079),8452036008);

1 row created.

30

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

SQL> insert into customer1

2 values(7,type\_name('Ankit','ganesh','agre'),

3 type\_address('gandhinagar','mumbai',400016),8184569254);

SQL> select\*from customer1;

C\_ID

----------

C\_NAME(FNAME, MNAME, LNAME)

--------------------------------------------------------------------------------

C\_ADD(STREET, CITY, PINCODE)

--------------------------------------------------------------------------------

C\_PHNO

----------

1

TYPE\_NAME('varsha', 's', 'atul')

TYPE\_ADDRESS('Sainagar', 'Mumbai', 400042)

123456789

C\_ID

----------

C\_NAME(FNAME, MNAME, LNAME)

--------------------------------------------------------------------------------

C\_ADD(STREET, CITY, PINCODE)

--------------------------------------------------------------------------------

C\_PHNO

----------

2

TYPE\_NAME('Tejas', 'ganesh', 'agre')

TYPE\_ADDRESS('ganesh gawade', 'Mumbai', 400080)

9867820182

C\_ID

31

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

----------

C\_NAME(FNAME, MNAME, LNAME)

--------------------------------------------------------------------------------

C\_ADD(STREET, CITY, PINCODE)

--------------------------------------------------------------------------------

C\_PHNO

----------

3

TYPE\_NAME('shruti', 'sunil', 'bhuvad')

TYPE\_ADDRESS('tilak', 'Pune', 400078)

9965824563

C\_ID

----------

C\_NAME(FNAME, MNAME, LNAME)

--------------------------------------------------------------------------------

C\_ADD(STREET, CITY, PINCODE)

--------------------------------------------------------------------------------

C\_PHNO

----------

4

TYPE\_NAME('mitesh', 'rajendra', 'roge')

TYPE\_ADDRESS('nehru', 'Pune', 400085)

9945238522

C\_ID

----------

C\_NAME(FNAME, MNAME, LNAME)

--------------------------------------------------------------------------------

C\_ADD(STREET, CITY, PINCODE)

--------------------------------------------------------------------------------

C\_PHNO

----------

5

32

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

TYPE\_NAME('kanase', 'dagdu', 'sakshi')

TYPE\_ADDRESS('vishnu', 'satara', 400012)

8856489621

C\_ID

----------

C\_NAME(FNAME, MNAME, LNAME)

--------------------------------------------------------------------------------

C\_ADD(STREET, CITY, PINCODE)

--------------------------------------------------------------------------------

C\_PHNO

----------

6

TYPE\_NAME('sahil', 'jagannath', 'mahale')

TYPE\_ADDRESS('savarkar', 'mumbai', 400079)

8452036008

C\_ID

----------

C\_NAME(FNAME, MNAME, LNAME)

--------------------------------------------------------------------------------

C\_ADD(STREET, CITY, PINCODE)

--------------------------------------------------------------------------------

C\_PHNO

----------

7

TYPE\_NAME('Ankit', 'ganesh', 'agre')

TYPE\_ADDRESS('gandhinagar', 'mumbai', 400016)

8184569254

7 rows selected.

33

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

SQL> select\*from customer1;

C\_ID

----------

C\_NAME(FNAME, MNAME, LNAME)

--------------------------------------------------------------------------------

C\_ADD(STREET, CITY, PINCODE)

--------------------------------------------------------------------------------

C\_PHNO

----------

1

TYPE\_NAME('varsha', 's', 'atul')

TYPE\_ADDRESS('Sainagar', 'Mumbai', 400042)

123456789

C\_ID

----------

C\_NAME(FNAME, MNAME, LNAME)

--------------------------------------------------------------------------------

C\_ADD(STREET, CITY, PINCODE)

--------------------------------------------------------------------------------

C\_PHNO

----------

2

TYPE\_NAME('Tejas', 'ganesh', 'agre')

TYPE\_ADDRESS('ganesh gawade', 'Mumbai', 400080)

9867820182

C\_ID

----------

C\_NAME(FNAME, MNAME, LNAME)

--------------------------------------------------------------------------------

34

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

C\_ADD(STREET, CITY, PINCODE)

--------------------------------------------------------------------------------

C\_PHNO

----------

3

TYPE\_NAME('shruti', 'sunil', 'bhuvad')

TYPE\_ADDRESS('tilak', 'Pune', 400078)

9965824563

C\_ID

----------

C\_NAME(FNAME, MNAME, LNAME)

--------------------------------------------------------------------------------

C\_ADD(STREET, CITY, PINCODE)

--------------------------------------------------------------------------------

C\_PHNO

----------

4

TYPE\_NAME('mitesh', 'rajendra', 'roge')

TYPE\_ADDRESS('nehru', 'Pune', 400085)

9945238522

C\_ID

----------

C\_NAME(FNAME, MNAME, LNAME)

--------------------------------------------------------------------------------

C\_ADD(STREET, CITY, PINCODE)

--------------------------------------------------------------------------------

C\_PHNO

----------

5

TYPE\_NAME('kanase', 'dagdu', 'sakshi')

35

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

TYPE\_ADDRESS('vishnu', 'satara', 400012)

8856489621

C\_ID

----------

C\_NAME(FNAME, MNAME, LNAME)

--------------------------------------------------------------------------------

C\_ADD(STREET, CITY, PINCODE)

--------------------------------------------------------------------------------

C\_PHNO

----------

6

TYPE\_NAME('sahil', 'jagannath', 'mahale')

TYPE\_ADDRESS('savarkar', 'mumbai', 400079)

8452036008

C\_ID

----------

C\_NAME(FNAME, MNAME, LNAME)

--------------------------------------------------------------------------------

C\_ADD(STREET, CITY, PINCODE)

--------------------------------------------------------------------------------

C\_PHNO

----------

7

TYPE\_NAME('Ankit', 'ganesh', 'agre')

TYPE\_ADDRESS('gandhinagar', 'mumbai', 400016)

8184569254

7 rows selected.

SQL> desc customer1;

36

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Name Null? Type

----------------------------------------- -------- ----------------------------

C\_ID NOT NULL NUMBER(5)

C\_NAME TYPE\_NAME

C\_ADD TYPE\_ADDRESS

C\_PHNO NUMBER(10)

SQL> set describe depth 2;

SQL> desc customer1;

Name Null? Type

---------------------------------- -------- ----------------------------

C\_ID NOT NULL NUMBER(5)

C\_NAME TYPE\_NAME

FNAME VARCHAR2(20)

MNAME VARCHAR2(20)

LNAME VARCHAR2(20)

C\_ADD TYPE\_ADDRESS

STREET VARCHAR2(20)

CITY VARCHAR2(20)

PINCODE NUMBER(10)

C\_PHNO NUMBER(10)

SQL> select c.c\_add.street from customer1 c where c\_id=1;

C\_ADD.STREET

--------------------

Sainagar

SQL> select c.c\_name.fname from customer1 c where c\_id=1;

C\_NAME.FNAME

--------------------

varsha

SQL> select c\_name from customer1;

37

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

C\_NAME(FNAME, MNAME, LNAME)

--------------------------------------------------------------------------------

TYPE\_NAME('varsha', 's', 'atul')

TYPE\_NAME('Tejas', 'ganesh', 'agre')

TYPE\_NAME('shruti', 'sunil', 'bhuvad')

TYPE\_NAME('mitesh', 'rajendra', 'roge')

TYPE\_NAME('kanase', 'dagdu', 'sakshi')

TYPE\_NAME('sahil', 'jagannath', 'mahale')

TYPE\_NAME('Ankit', 'ganesh', 'agre')

7 rows selected.

SQL> select c\_id,c.c\_name.lname from customer1 c;

C\_ID C\_NAME.LNAME

---------- --------------------

1 atul

2 agre

3 bhuvad

4 roge

5 sakshi

6 mahale

7 agre

7 rows selected.

SQL> select c.c\_name.fname||' '||c.c\_name.mname||' '||c.c\_name.lname

2 from customer1 c;

C.C\_NAME.FNAME||''||C.C\_NAME.MNAME||''||C.C\_NAME.LNAME

--------------------------------------------------------------

varsha s atul

Tejas ganesh agre

38

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

shruti sunil bhuvad

Shahurajendra roge

kanase dagdu sakshi

sahil jagannath mahale

Ankit ganesh agre

7 rows selected.

Nesting and Altering existing type

SQL> alter type type\_address

2 add attribute(name type\_name)cascade;

Type altered.

SQL> create table cust2

2 (cust\_no integer,

3 add1 type\_address);

Table created.

SQL> insert into cust2

2 values(101,

3 type\_address('prabhadevi','mumbai',400012,

4 type\_name('yashu','gaurang','sarvesh')));

1 row created.

SQL> insert into cust2

2 values(102,

3 type\_address('dadar','mumbai',400027,

4 type\_name('yashu','gaurang','sarvesh')));

1 row created.

SQL> insert into cust2

2 values(103,

39

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

3 type\_address('thane','mumbai',400018,

4 type\_name('Tejas','ganesh','agre')));

1 row created.

SQL> insert into cust2

2 values(104,

3 type\_address('mulund','mumbai',400080,

4 type\_name('shruti','sunil','bhuvad')));

1 row created.

SQL> insert into cust2

2 values(105,

3 type\_address('bhandup','mumbai',400078,

4 type\_name('sahil','jagannath','mahale')));

1 row created.

SQL> drop table cust2;

Table dropped.

SQL> create table cust2

2 (cust\_no integer,

3 add1 type\_address);

Table created.

SQL> insert into cust2

2 values(101,

3 type\_address('prabhadevi','mumbai',400012,

4 type\_name('yashu','gaurang','sarvesh')));

40

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

1 row created.

SQL> insert into cust2

2 values(102,

3 type\_address('dadar','mumbai',400027,

4 type\_name('mitesh','rajendra','roge')));

1 row created.

SQL> insert into cust2

2 values(103,

3 type\_address('thane','mumbai',400018,

4 type\_name('Tejas','ganesh','agre')));

1 row created.

SQL> insert into cust2

2 values(104,

3 type\_address('mulund','mumbai',400080,

4 type\_name('shruti','sunil','bhuvad')));

1 row created.

SQL> insert into cust2

2 values(105,

3 type\_address('bhandup','mumbai',400078,

4 type\_name('sahil','jagannath','mahale')));

1 row created.

SQL> desc cust2

Name Null? Type

----------------------------------------- -------- ----------------------------

CUST\_NO NUMBER(38)

ADD1 TYPE\_ADDRESS

41

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

SQL> select cust\_no from cust2;

CUST\_NO

----------

101

102

103

104

105

SQL> select cust\_no,c.add1.street,c.add1.name.fname from cust2 c;

CUST\_NO ADD1.STREET ADD1.NAME.FNAME

---------- -------------------- --------------------

101 prabhadevi yashu

102 dadar mitesh

103 thane Tejas

104 mulund shruti

105 bhandup sahil

OBJECT TABLE:

SQL> create or replace type stud\_type as object

2 (roll\_no number(5),

3 name varchar2(30)

4 );

5 /

SQL> create table students of stud\_type;

Table created.

SQL> insert into students values(stud\_type(1,'Tejas'));

42

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

1 row created.

SQL> insert into students values(stud\_type(2,'sakshi'));

1 row created.

SQL> insert into students values(stud\_type(3,'shahu'));

1 row created.

SQL> insert into students values(stud\_type(4,'shruti'));

1 row created.

SQL> insert into students values(stud\_type(5,'snehal'));

1 row created.

SQL> select\*from students;

ROLL\_NO NAME

---------- ------------------------------

1 Tejas

2 sakshi

3 shahu

4 shruti

5 snehal

SQL> select roll\_no from students;

ROLL\_NO

----------

1

2

3

43

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

4

5

SQL> select s.roll\_no from students s;

ROLL\_NO

----------

1

2

3

4

5

44

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

REF & DREF Function

Creating Object tables

SQL> create or replace type ANIMAL\_TY as object

2 (Breed varchar2(25),

3 Name varchar2(25),

4 BirthDate DATE);

5 /

Type created.

SQL> create table ANIMAL of ANIMAL\_TY;

Table created.

SQL> insert into ANIMAL values(

2 ANIMAL\_TY('MULE','FRANCES','01-APR-02'));

1 row created.

SQL> insert into ANIMAL values(

2 ANIMAL\_TY('DOG','BENJI','03-SEP-01'));

1 row created.

SQL> insert into ANIMAL values(

2 ANIMAL\_TY('HUSKY','BRUNO','16-oct-02'));

1 row created.

The REF Function

SQL> select REF(A) from ANIMAL A;

REF(A)

--------------------------------------------------------------------------------

0000280209B7B787E9D66947D288EFFF32FB969ECD55911AAC613543BCB6EFFAB11

45

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

6D17540010006

ED0000

0000280209837FC3B240784C6E85724E1BC2424DB255911AAC613543BCB6EFFAB116

D17540010006

ED0001

0000280209A8EC9E5BC4C34CFC9CA4EA509FE091D655911AAC613543BCB6EFFAB1

16D17540010006

ED0002

Using the DREF Function

SQL> create table KEEPER

2 (KeeperName varchar2(25),

3 AnimalKept REF ANIMAL\_TY);

Table created.

SQL> set describe depth 2;

SQL> describe KEEPER

Name Null? Type

----------------------------------------- -------- ----------------------------

KEEPERNAME VARCHAR2(25)

ANIMALKEPT REF OF ANIMAL\_TY

BREED VARCHAR2(25)

NAME VARCHAR2(25)

BIRTHDATE DATE

SQL> insert into KEEPER

2 select 'CATHERINE',

3 REF(A)

4 from ANIMAL A

5 where Name='BENJI';

46

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

1 row created.

SQL> insert into KEEPER

2 select 'ELENA',

3 REF(A)

4 from ANIMAL A

5 where NAME='BRUNO';

1 row created.

SQL> select\*from KEEPER;

KEEPERNAME

-------------------------

ANIMALKEPT

--------------------------------------------------------------------------------

CATHERINE

0000220208837FC3B240784C6E85724E1BC2424DB255911AAC613543BCB6EFFAB116

D17540

ELENA

0000220208A8EC9E5BC4C34CFC9CA4EA509FE091D655911AAC613543BCB6EFFAB1

16D17540

SQL> select KeeperName,DEREF(K.AnimalKept)

2 from KEEPER K;

KEEPERNAME

-------------------------

DEREF(K.ANIMALKEPT)(BREED, NAME, BIRTHDATE)

--------------------------------------------------------------------------------

CATHERINE

ANIMAL\_TY('DOG', 'BENJI', '03-SEP-01')

ELENA

ANIMAL\_TY('HUSKY', 'BRUNO', '16-OCT-02')

47

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Practical No.: 04

Aim: Implementation of ETL transformation with Pentaho

CREATE TABLE:

SQL> create table emp01

2 (

3 emp\_no numeric(5),

4 fname varchar2(10),

5 lname varchar2(10),

6 salary numeric(5),

7 comm numeric(5)

8 );

Table created.

SQL> insert into emp01 values(01,'Rasika','Kerkar',50000,500);

1 row created.

SQL> insert into emp01 values(02,'Tejas','Agre',45000,200);

1 row created.

SQL> insert into emp01 values(03,'Sakshi','Kanase',55000,700);

1 row created.

SQL> insert into emp01 values(04,'Shahu','Kalange',40000,600);

1 row created.

SQL> insert into emp01 values(05,'Mitesh','Roge',35000,300);

1 row created.

SQL> insert into emp01 values(06,'Riddhik','Jamdade',75000,1200);

1 row created.

SQL> insert into emp01 values(07,'Sahil','Mahale',60000,200);

1 row created.

SQL> insert into emp01 values(08,'Shruti','Bhuvad',45000,400);

1 row created.

SQL> insert into emp01 values(09,'Snehal','Borji',72000,1500);

1 row created.

SQL> insert into emp01 values(10,'Kaustubh','Naik',82000,750);

1 row created.

SQL> select\*from emp01;

EMP\_NO FNAME LNAME SALARY COMM

---------- ---------- ---------- ---------- ----------

1 Rasika Kerkar 50000 500

2 Tejas Agre 45000 200

3 Sakshi Kanase 55000 700

4 Shahu Kalange 40000 600

5 Shahu Roge 35000 300

6 Riddhik Jamdade 75000 1200

7 Sahil Mahale 60000 200

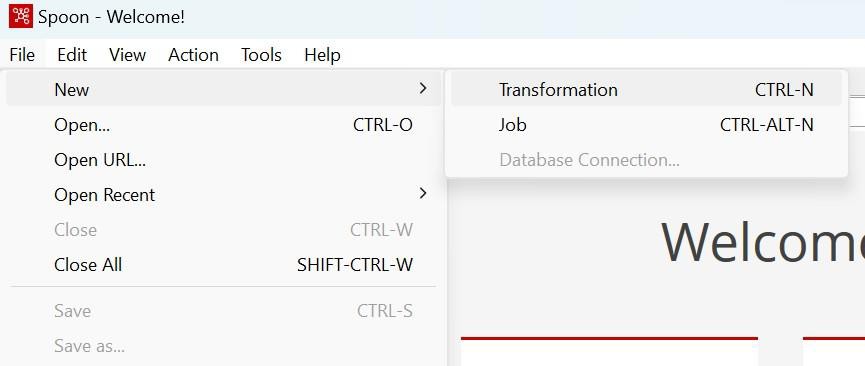
8 Shruti Bhuvad 45000 400

9 Snehal Borji 72000 1500

10 Kaustubh Naik 82000 750

10 rows selected.

48



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

TRANSFORMATION 1:

Step 1: In the data integration folder open“Spoon (Windows Batch File)”.

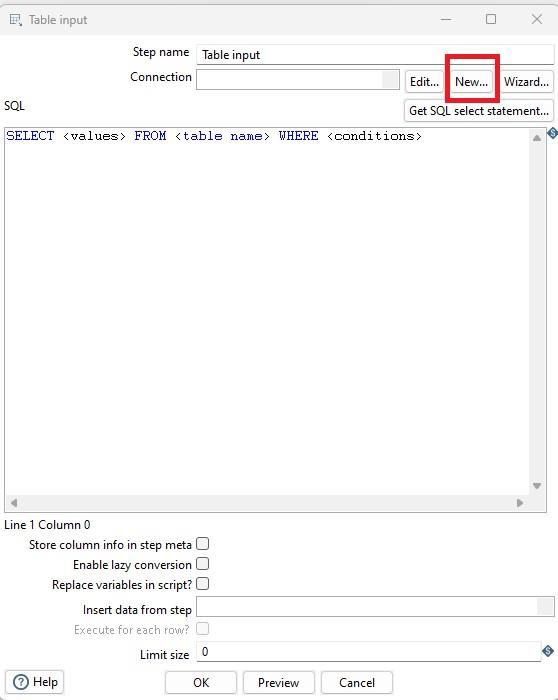
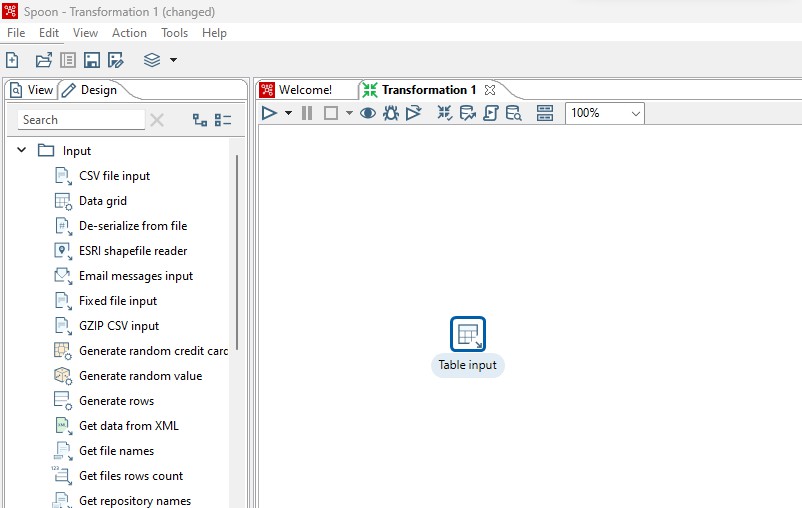
Step 2:Go to File→New→Transformation.

Step 3:Import SQL table to pentaho-

Design tab→Input folder→Drag and dropTable input

Double click on table input

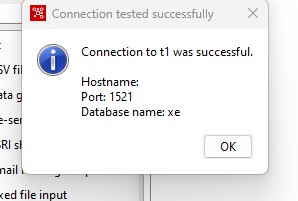
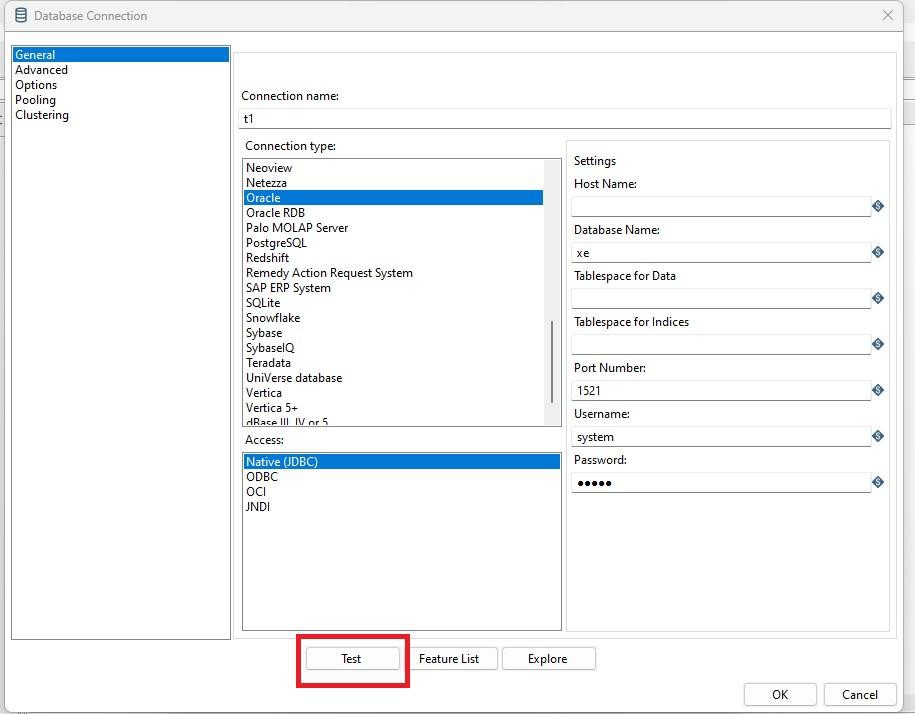
49



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Click on New

50



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Connect to the Database:

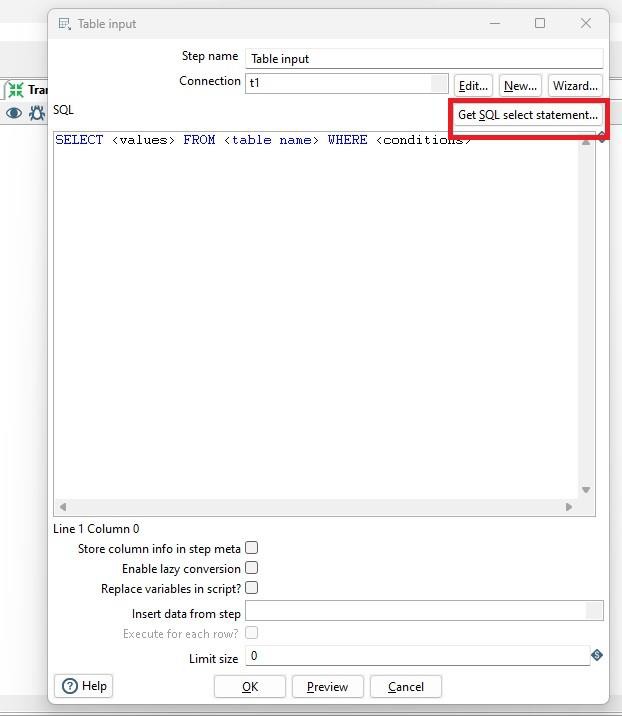
Fill in the details as below. Here enter User Name and Password same as your database

username and password. Then click on Test.

Click OK→

again Ok

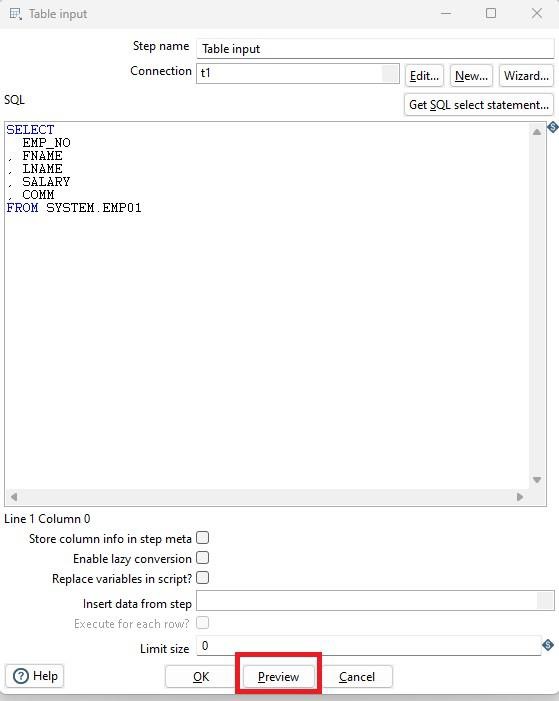
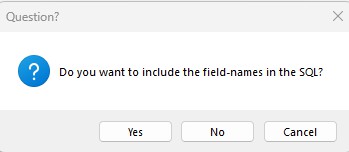
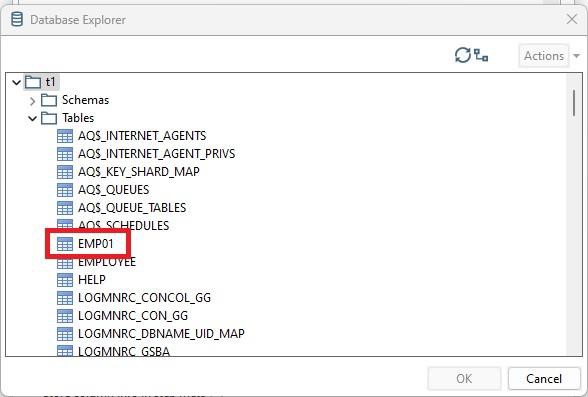
51



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Get SQL select statement… in table input window.

52



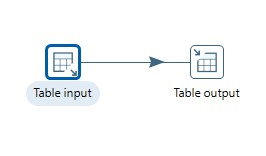
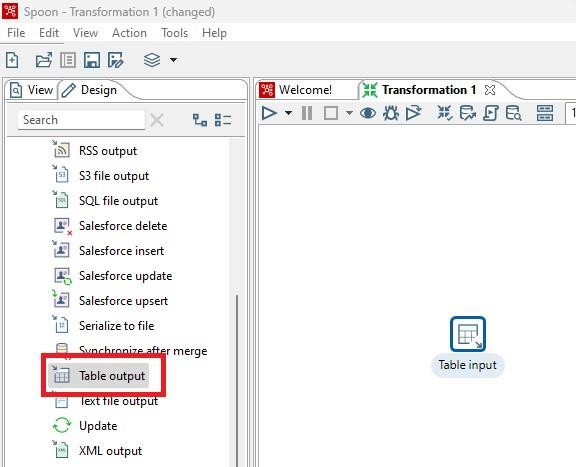
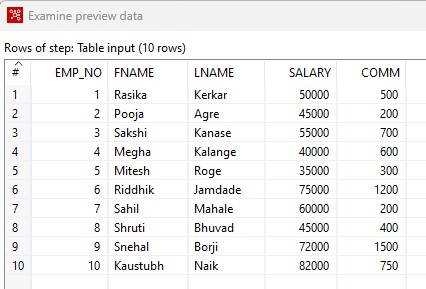
MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Import table:

In t1, under tables, select the required table (In this case emp01).

Click on OK→Click on Yes

53



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Preview inTable input window →OK

Click Close →OK

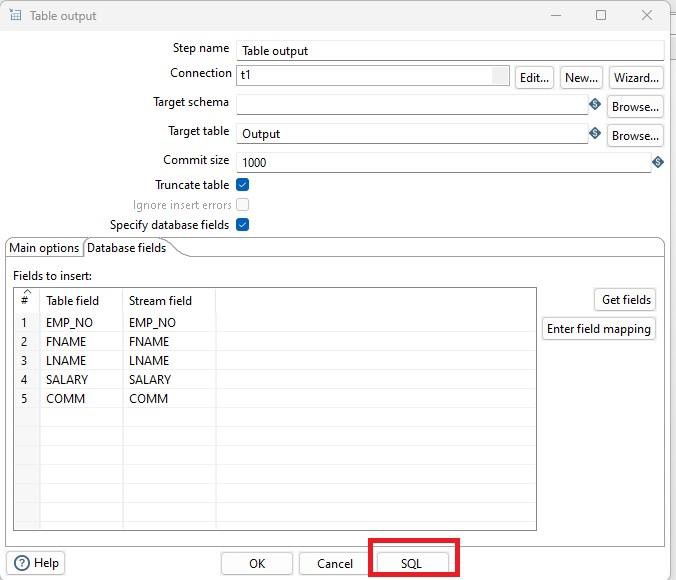
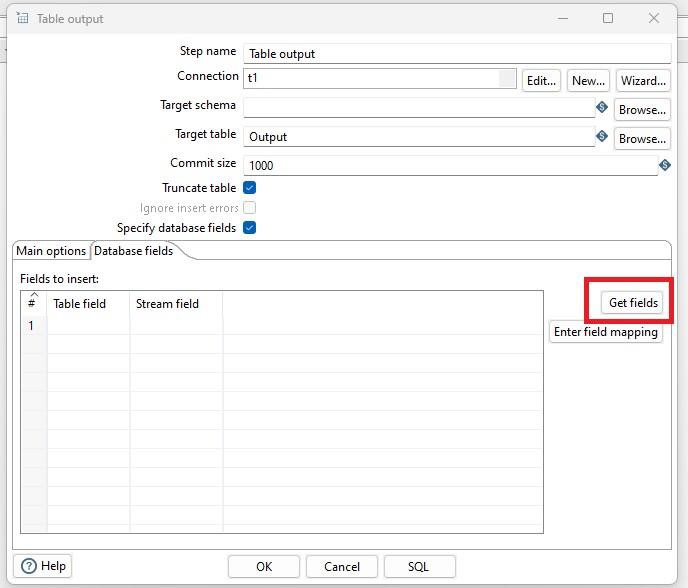
Step 4:Show output:

Drag and DropTable Output

Hold the mouse pointer on Table input and select and drag the output

connector to theTableoutput.

54



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Double Click on Table Output.

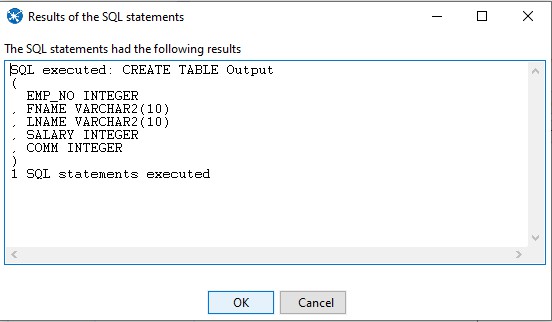
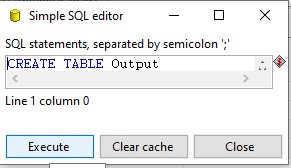
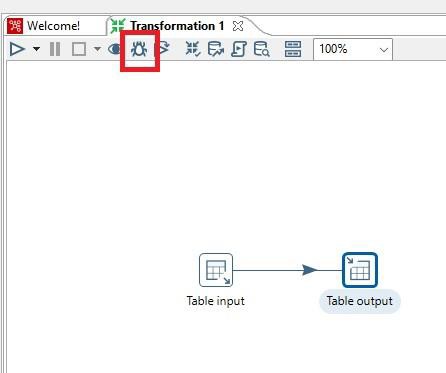
In the Table Output Window, give name to the Target table, check the check

boxes and clickon Get fields.

Click on SQL.

Click on Execute→OK

55



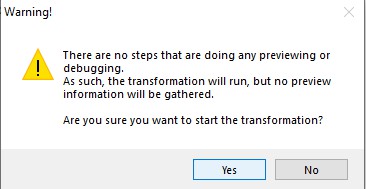
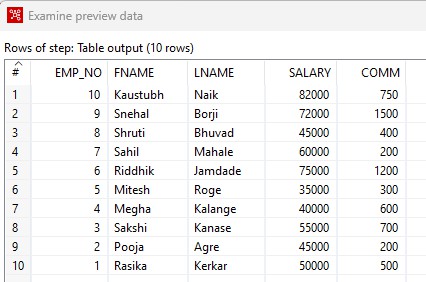
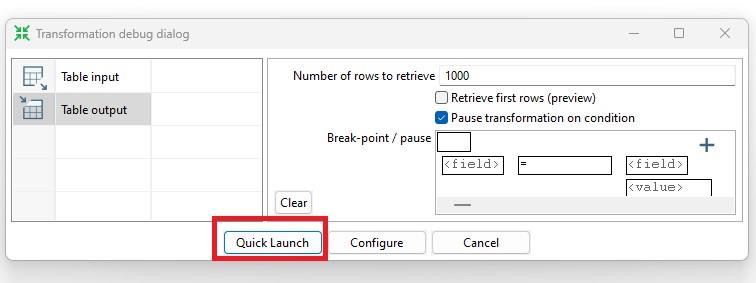
MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Click on Close→OK

Then click on Debug this transformation.

Click on Quick Launch.

56

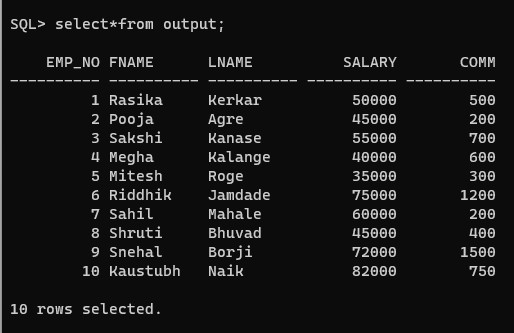
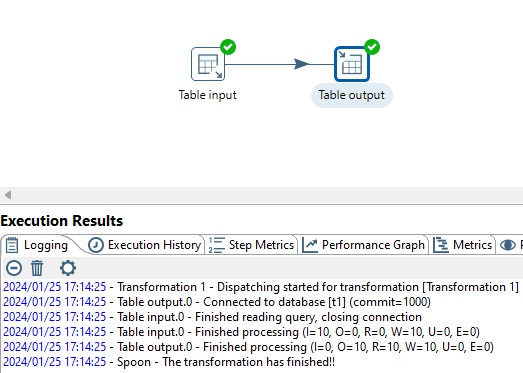


MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Click on Yes.

If the Transformation is successful, you will see green ticks.

57



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

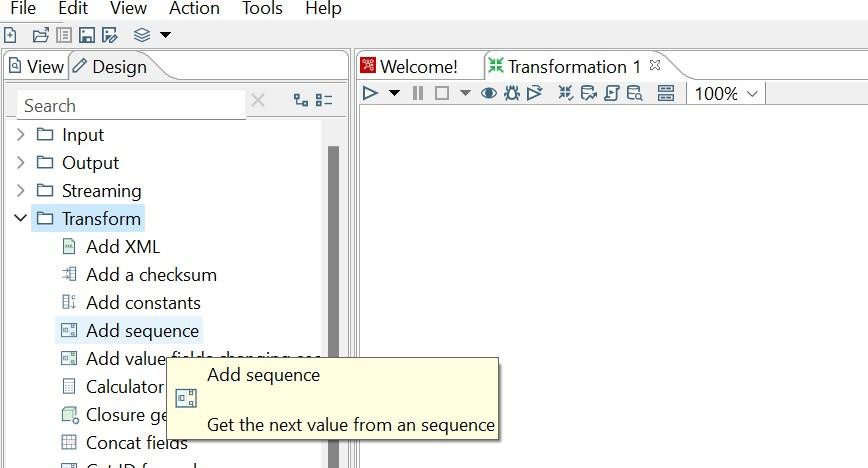
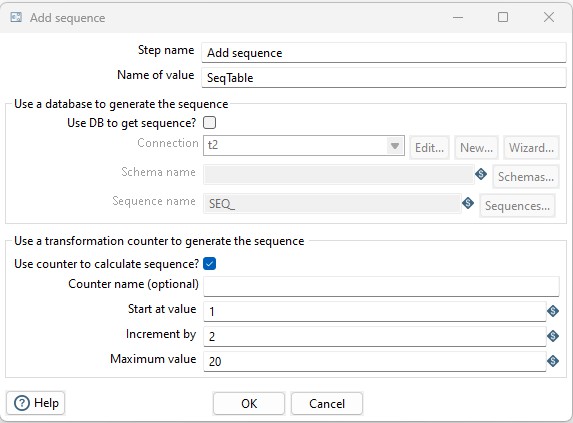
Close→close

You will also see the table created in the database with the name same as the

target table tablename.

Step 5:Run query in SQL Plus.

58



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

TRAMSFORMATION 2: Add sequence.

Step 1:Repeat Steps 2 and 3 from TRANSFORMATION 1.

Step 2:Perform transformation (Add sequence).

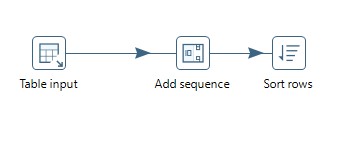
Drag and dropAdd Sequencefrom the transform folder under the Design tab.

Hold the mouse pointer on Table input and select and drag the output connector to the Add

sequence.

Double click on Add sequence and fill in the details as shown below→Click on OK.

59



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Step 3:Perform transformation (Sort rows)

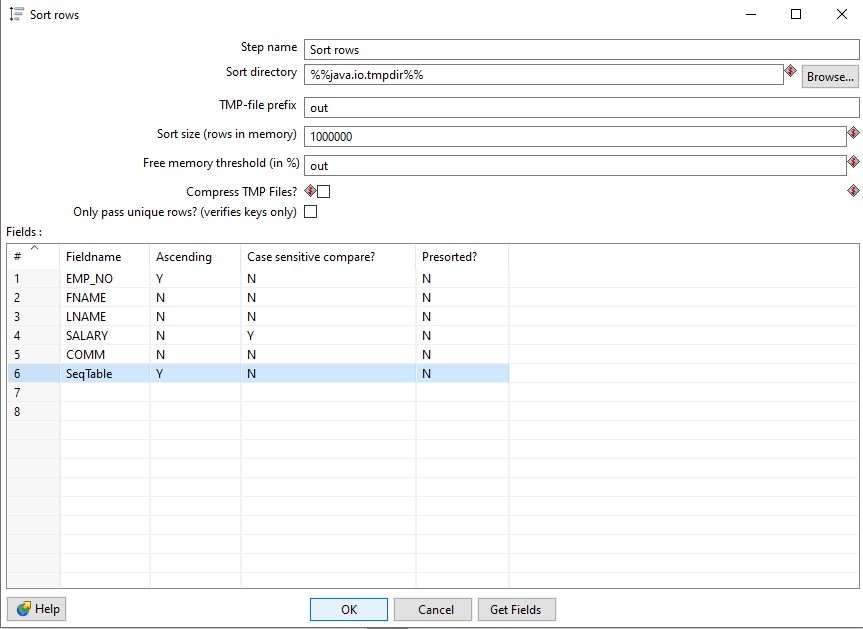
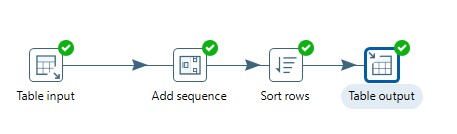
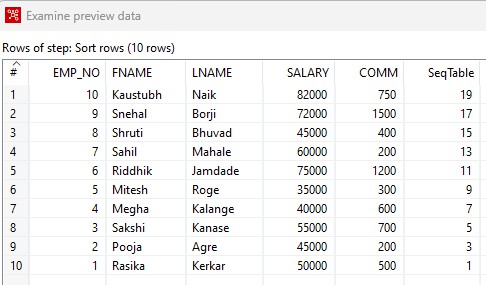
Drag and dropSort rows from the transform folder under theDesign tab.

Hold the mouse pointer on Add sequenceand select and drag the output connector to the Sort

rows.

Double click on Sort rows and fill in the details as shown below→Click on OK.

60

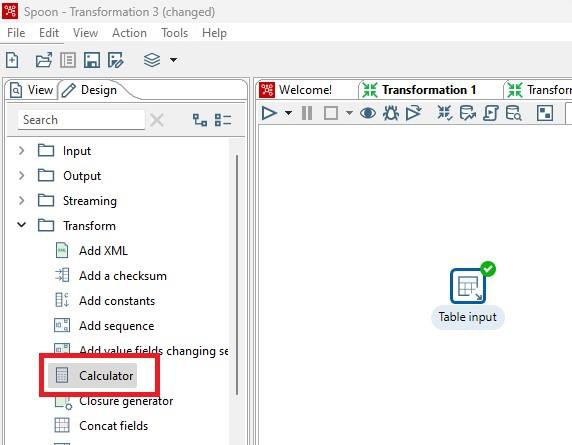
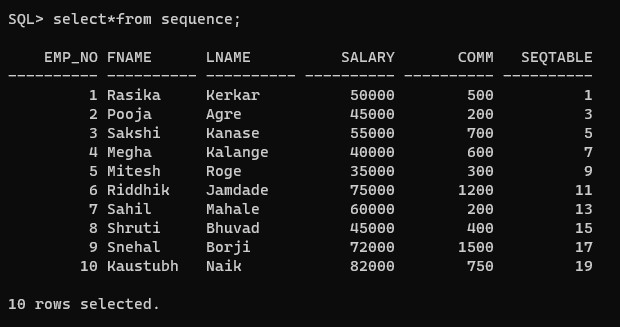


MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Step 4:Repeat Step 4 from TRANSFORMATION 1.

If the Transformation is successful, you will see green ticks.

61



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Step 5:Run query in SQL.

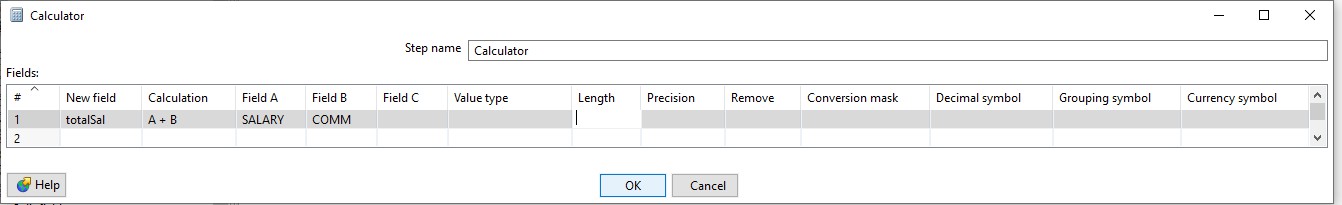
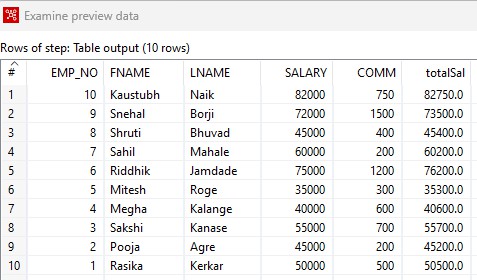
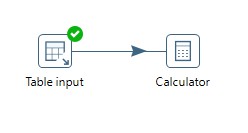
TRAMSFORMATION 3: Calculator

Repeat Steps 1 to 3 from TRANSFORMATION 1.

Step 4:Perform Transformation

Drag and dropCalculator from Transform folder underDesign tab

62



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Hold the mouse Pointer on Table input and select and drag the output connector to the

Caculator.

Double Click on Calculator and fill in the details as shown below.

This will add the values in SALARY column and COMM column as result will be stored in

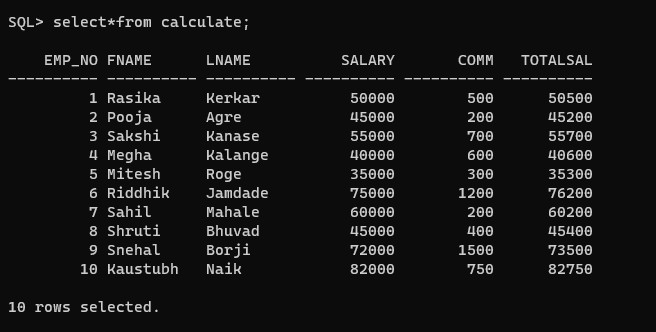
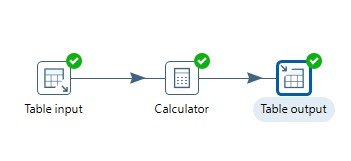
TOTALSAL column.

Click on OK.

Step 5: Repeat Step 4 from TRANSFORMATION 1.

If the Transformation is successful, you will see green ticks.

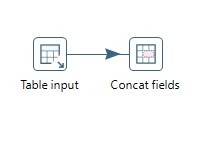
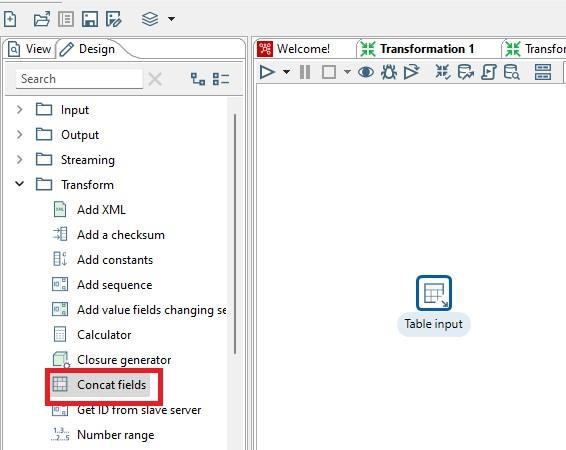
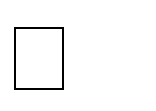
63



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Step 6:Run SQL query.

64



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

TRANSFORMATION 4: Concat Fields

Step 1: Repeat Steps 2 and 3 from TRANSFORMATION 1.

Step 2: Perform Transformation.

Drag and dropConcat Fields from Transform folder under Design tab

Hold the mouse Pointer on Table input and select and drag the output connector to the Concat

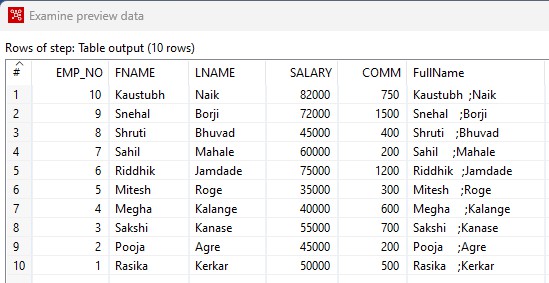
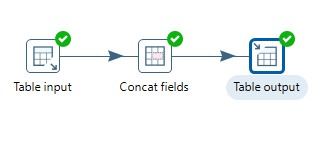
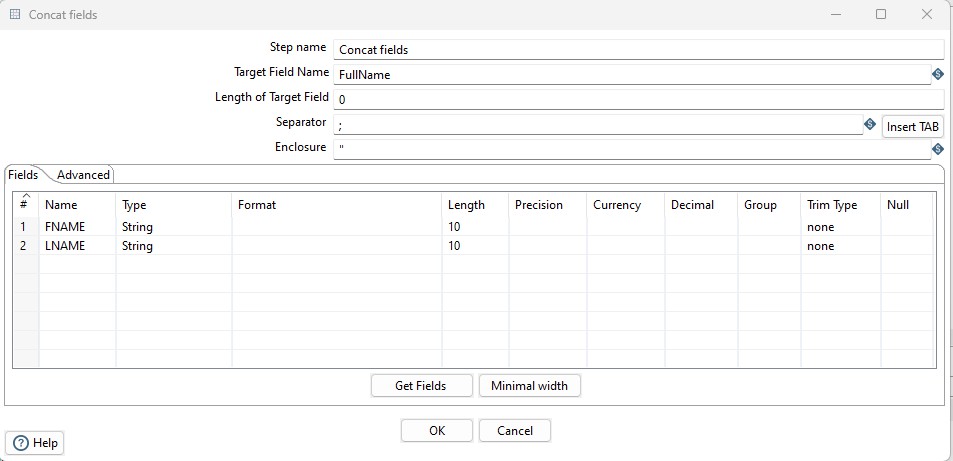
Fields.

Double Click on Concat Fields and fill in the details as shown below Click on OK.

Step 3:Repeat Step 4 from TRANSFORMATION 1.

If the Transformation is successful, you will see green ticks.

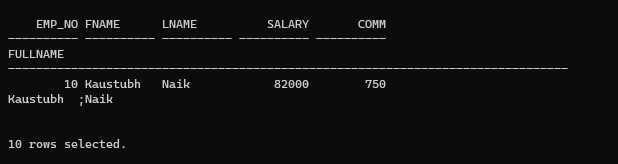
65



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

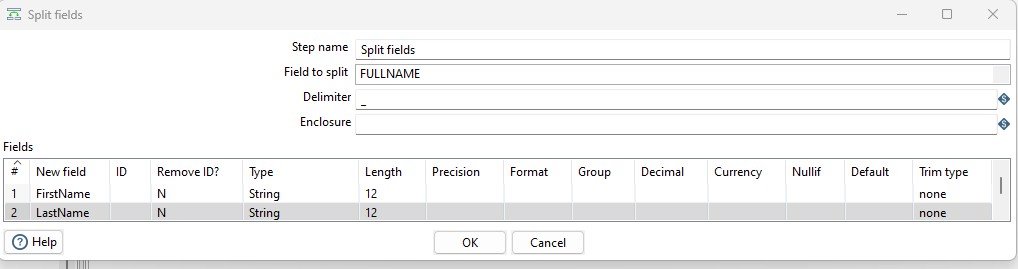
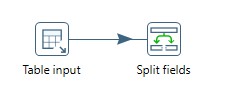
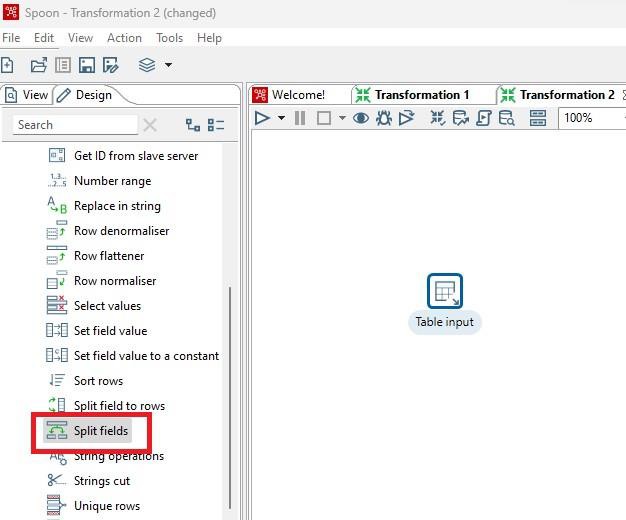
Step 4:Run SQL query.

66



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

67



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

TRANSFORMATION 5: Split Fields.

Step 1:Repeat Steps 2 and 3 from TRANSFORMATION 1 (Import output

table of concatfields transformation as Table input).

Step 2:Perform Transformation.

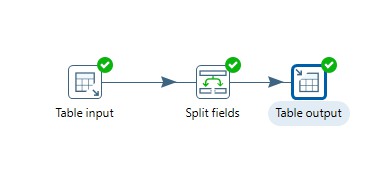
Drag and dropConcat Fields from Transform folder under Design tab.

Hold the mouse Pointer on Table input and select and drag the output connector to the Split

Fields.

Double Click on Split Fields and fill in the details as shown below→Click on OK

68



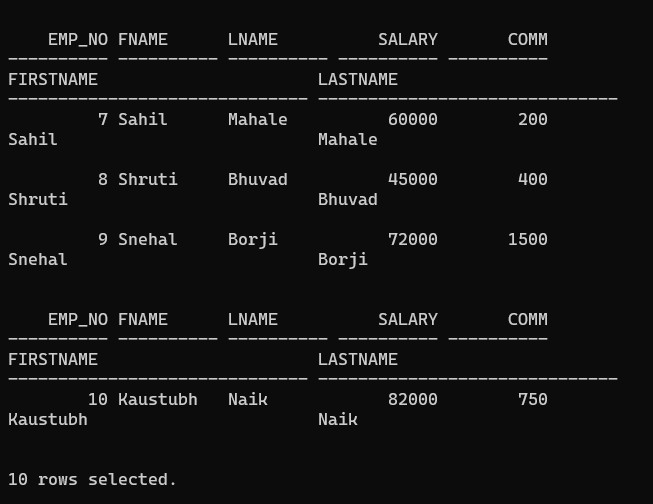
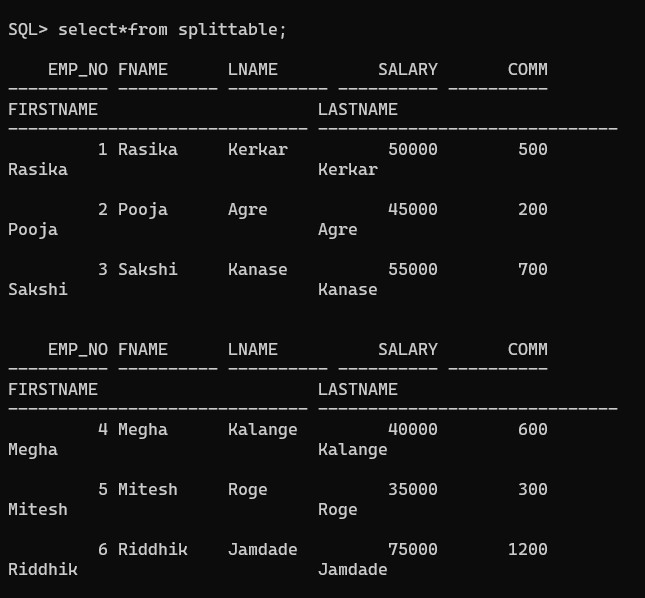
MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Step 3:Repeat Step 4 from TRANSFORMATION 1.

If the Transformation is successful, you will see green ticks

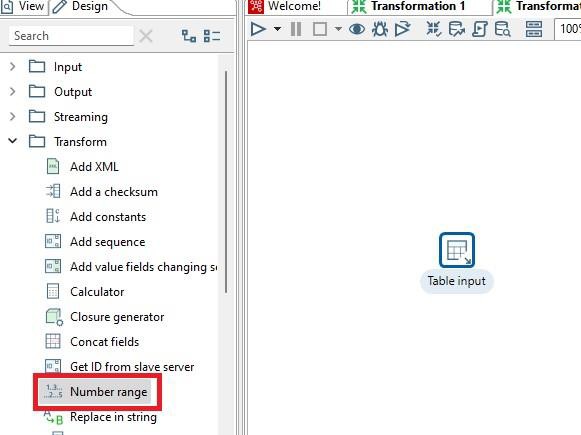
Step 4: Run SQL query.

69



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

70



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

TRANSFORMATION 6: Number Range

SQL> select\*from percent01;

ROLL\_NO NAME PERCENTAGE

---------- ---------- ----------

1 Tejas 90

2 Tejas 85

3 Vedika 68

4 Ruhi 79

5 Raj 95

6 Swara 33

7 Reva 56

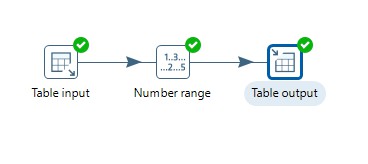
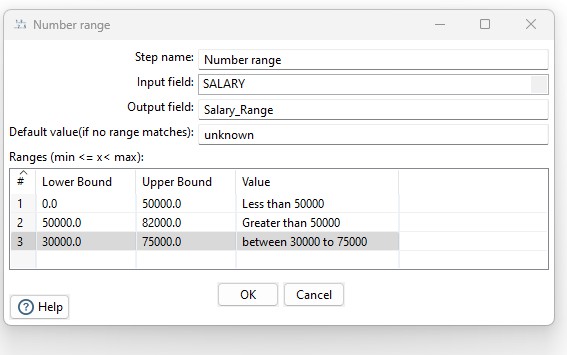
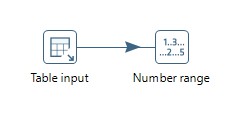
7 rows selected.

Step 1:Repeat Steps 2 and 3 from TRANSFORMATION 1.

Step 2:Perform Transformation.

Drag and dropNumber Range from Transform folder under Design tab.

71



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Hold the mouse Pointer on Table input and select and drag the output connector to the

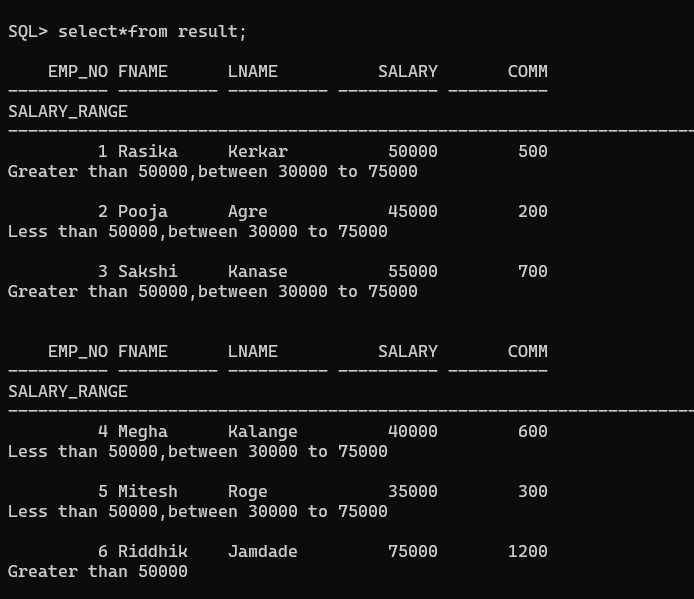
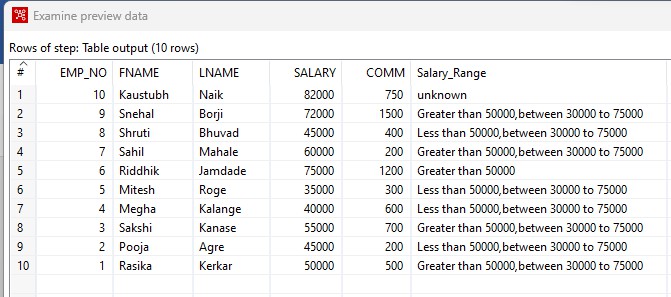
Number range.

Double Click on Number range and fill in the details as shown below→Click on OK

Step 3:Repeat Step 4 from TRANSFORMATION 1.

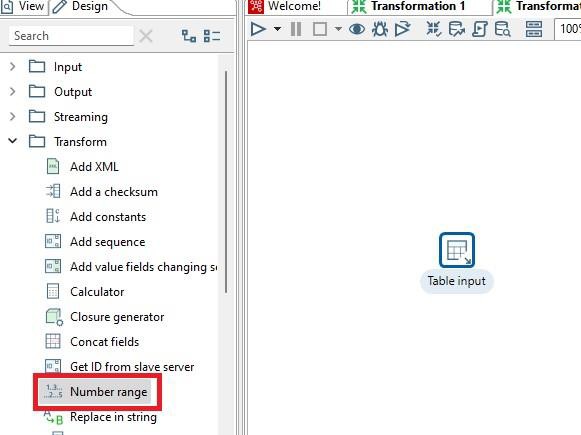
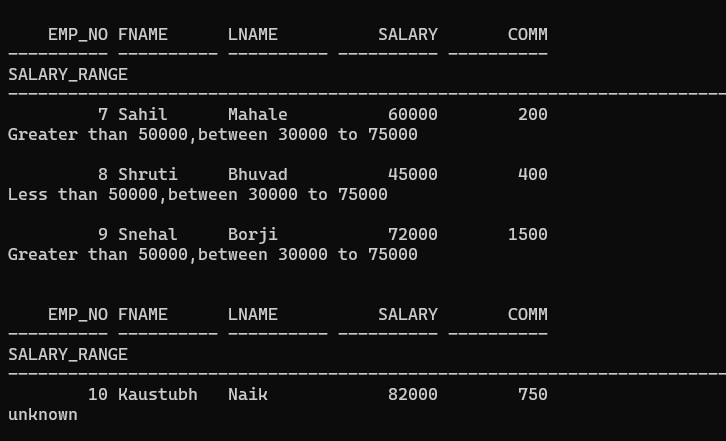
If the Transformation is successful, you will see green ticks.

72



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

73



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

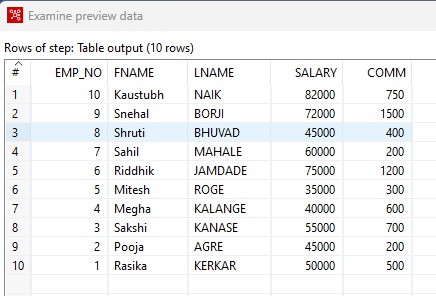
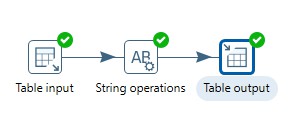
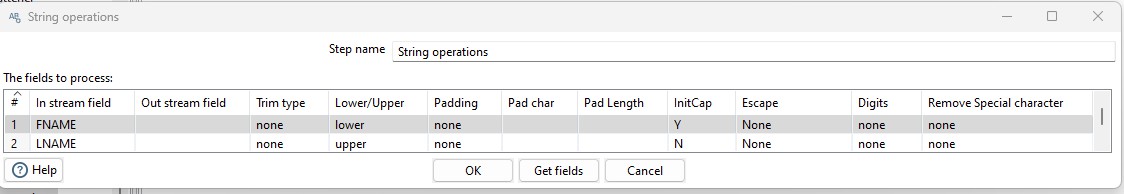
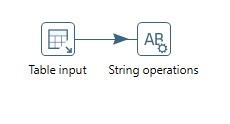
TRANSFORMATION 7: String Operations

Step 1:Repeat Steps 2 and 3 from TRANSFORMATION 1.

Step 2:Perform Transformation.

Drag and dropString Operations from Transform folder underDesign tab.

74



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Hold the mouse Pointer on Table input and select and drag the output connector to the String

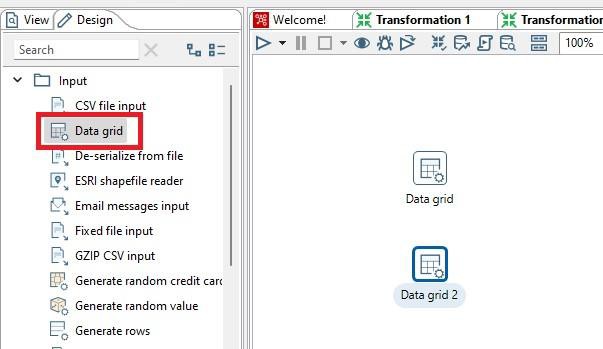
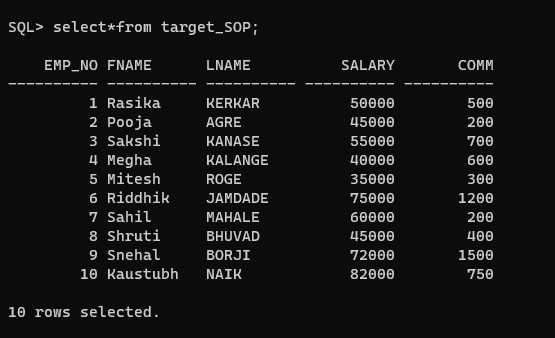
operations

Double Click on String operations and fill in the details as shown below→Click on OK.

Step 3:Repeat Step 4 from TRANSFORMATION 1.

If the Transformation is successful, you will see green ticks.

75



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Step 4:Run SQL query

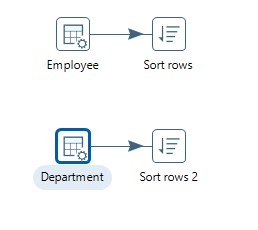
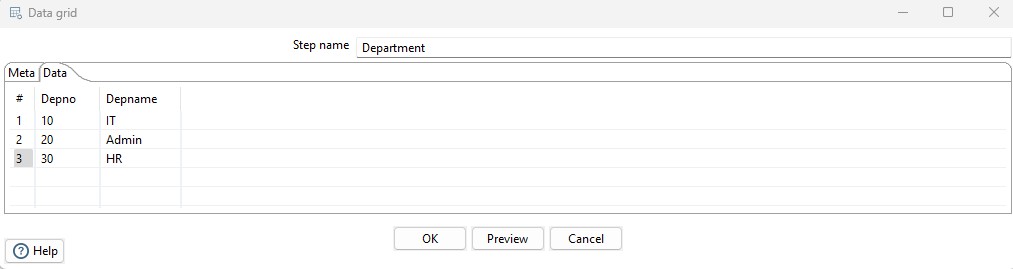
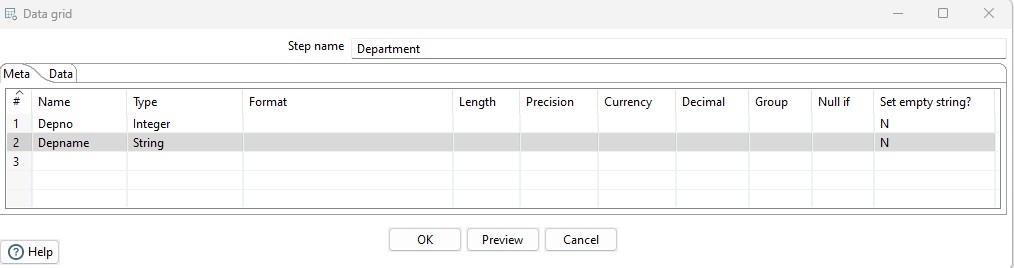
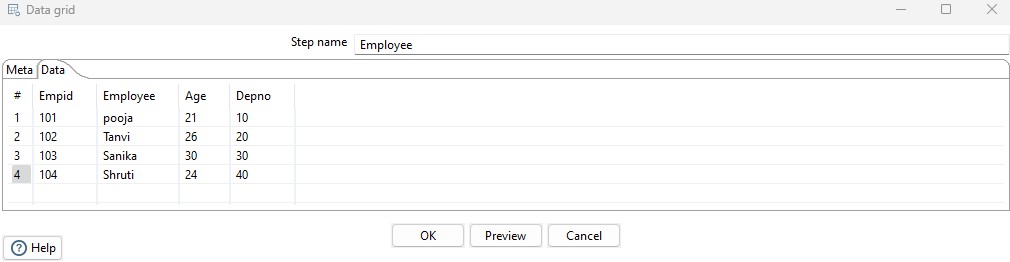
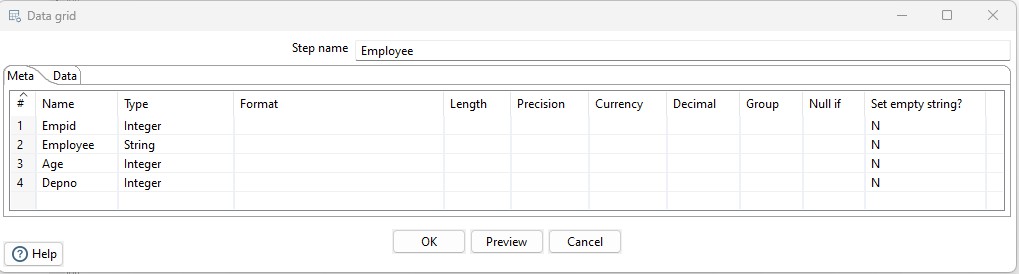
TRANSFORMATION 8: Merge Join

Step 1:Drag and drop 2 Data Grid from Input folder underDesign tab.

Rename them as Employee and Department.

Step 2:Double click on them and insert records into respective grids→Click on OK.

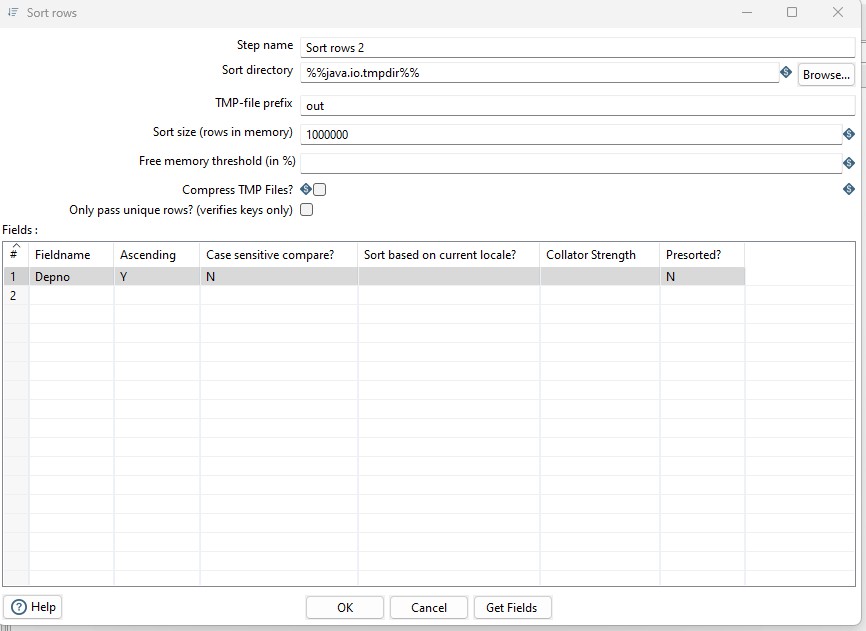
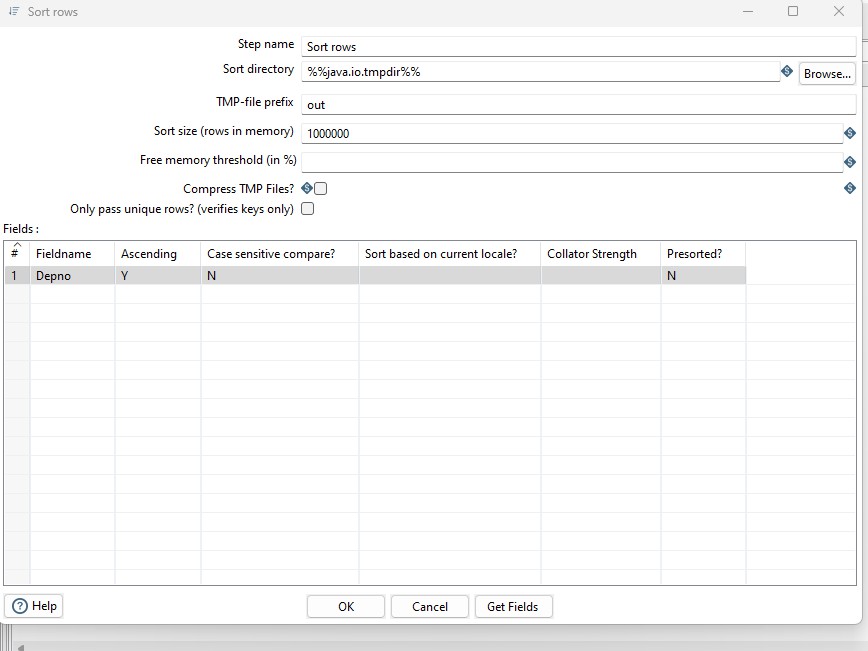
76



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

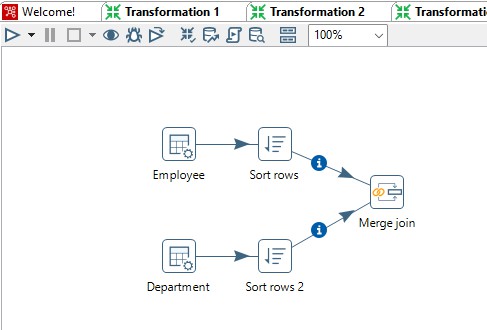
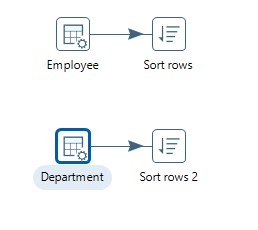
Step3: Insert sort rows transformation(on Empid) for both Employee and Department.

77



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

78



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

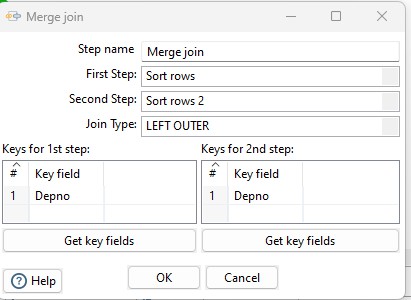
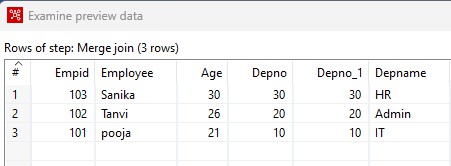
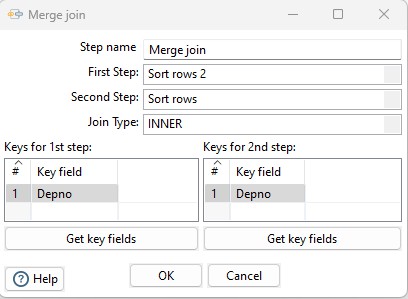
Step4: Add Merge join From Joins and connect it with sort rows.

Step5: Double click on Merge join and made necessary settings First select First\_step, then

Second\_step, select Join type INNER, and from get fields select Depno only, after that quick

launch the transformation.

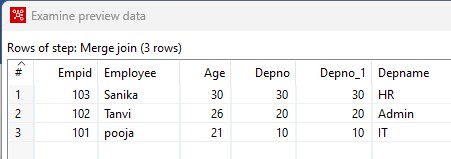
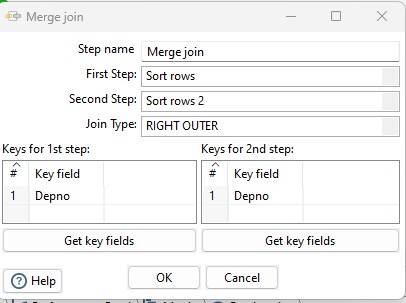
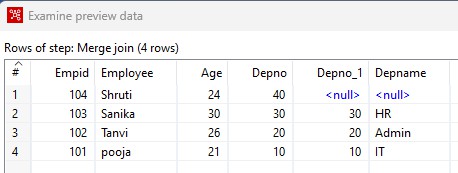
79



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

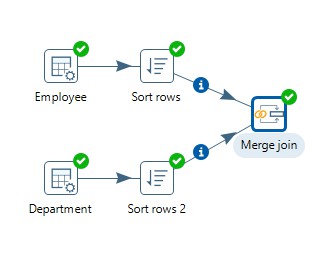
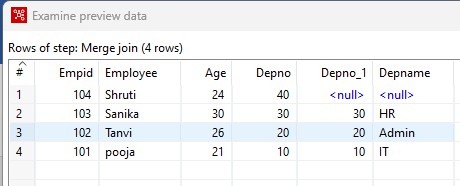
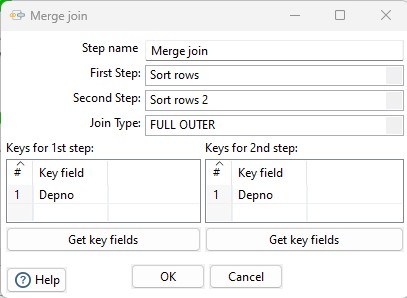
Step 6: Select Join Type Left Outer, then Right Outer and finally Full Outer join

80



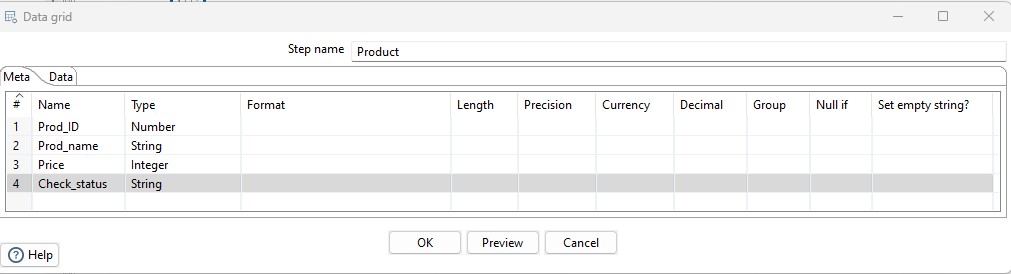
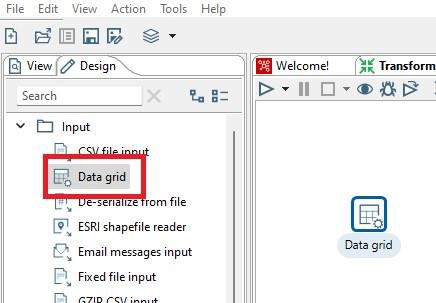
MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

81



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

82



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

TRANSFORMATION 9: Data validations

Step 1:Drag and dropData Grid from Input folder under Design tab

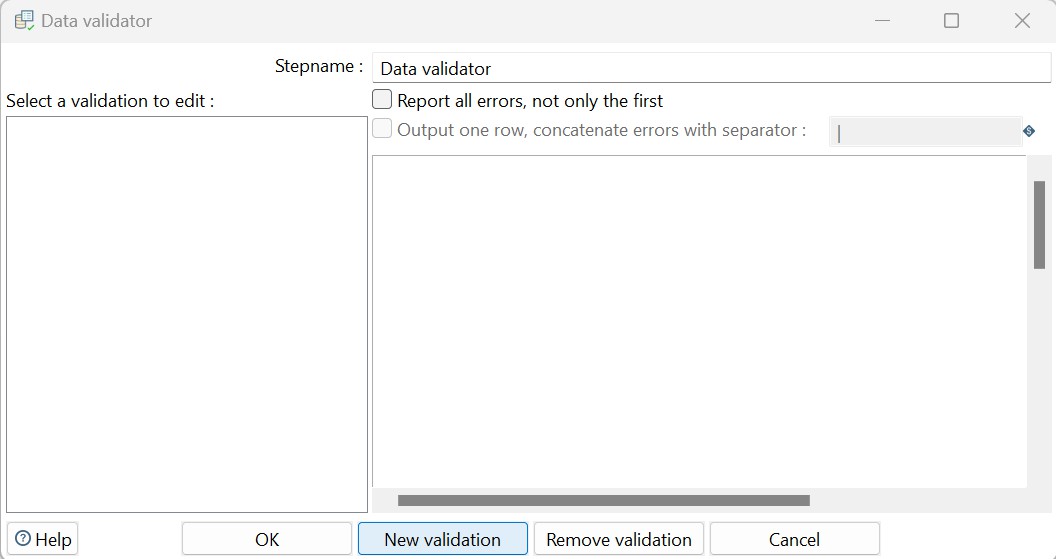
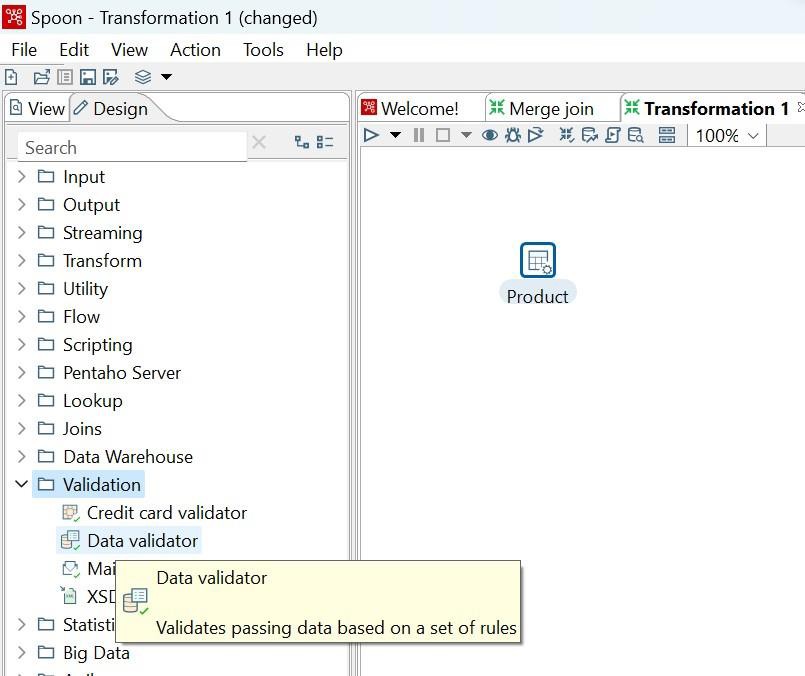
Rename it as Product.

Step 2:Double click on Product data grid and insert records as shown below→Click on

OK.

Step 3:Drag and drop Data validator from Validation folder under Design tab.

83



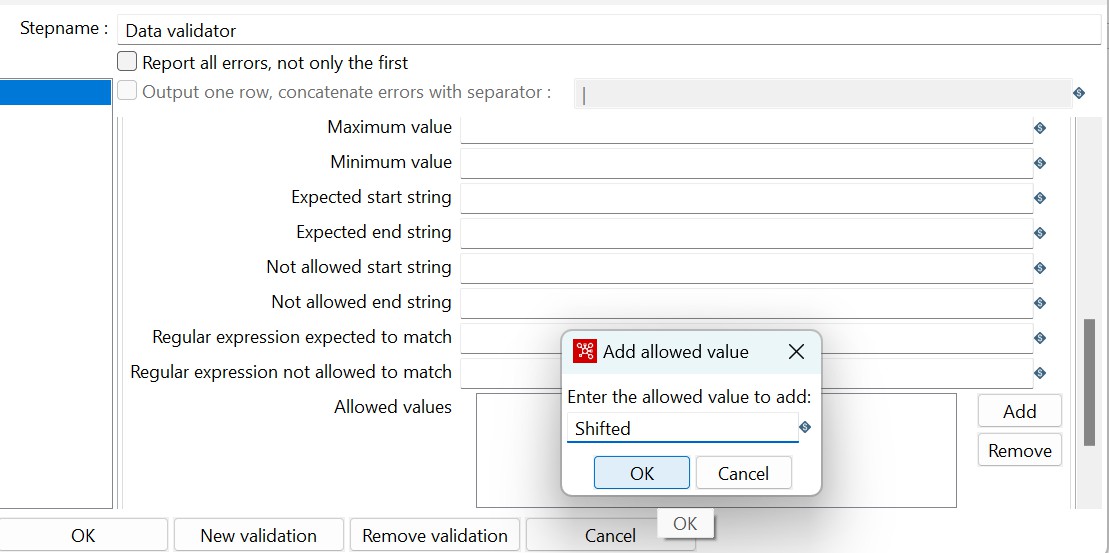
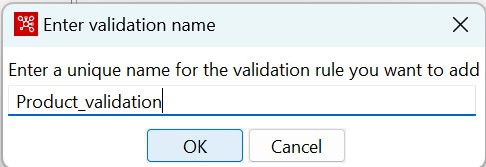
MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Hold the mouse pointer on Product data grid and select and drag the output connector to the

Data validator.

Double clickc on Data validator→New Validattion

84



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Give Valiadtion Name and click OK.

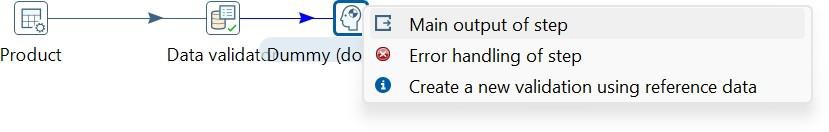
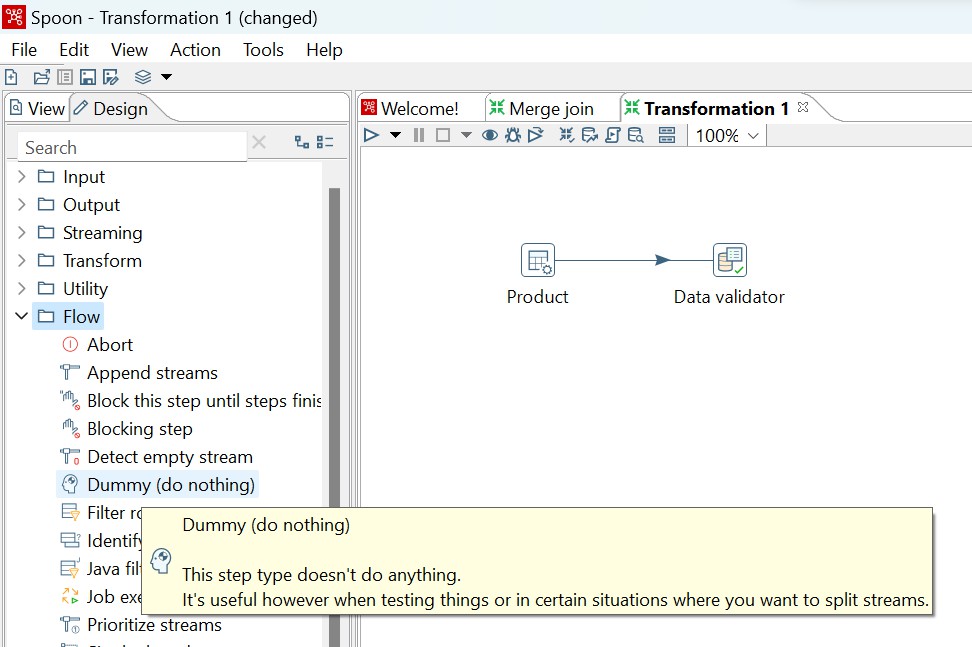
Select the validation to edit and fill in the details as shown below.

Click on add to set validation, set it 65 to Shifted and press Enter and click on ok.

Click on OK.

Step 4: Drag and drop Dummy from Flow folder under the Design tab

85



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

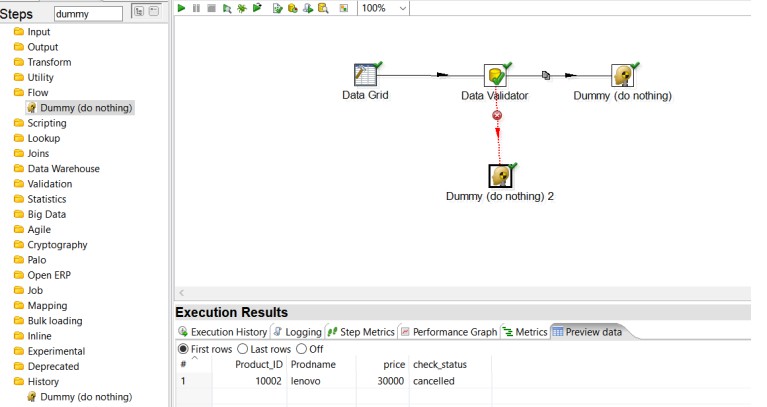
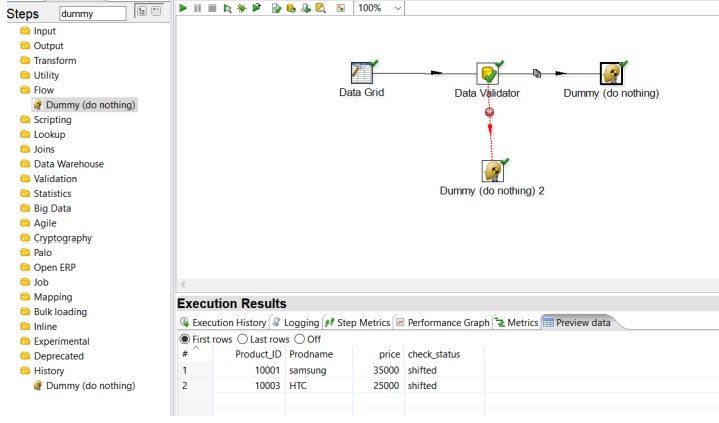
Hold the mouse pointer on Data validator and select and drag the output connector to the

Dummy. Select Main output of step.

Step 5:Drag and drop another Dummy from Flow folder under the Design tab and connect it

to the data validator. Select Error handling of step. In the next window click in Copy.

86



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Step 6:Quick Launch the transformation selecting one dummy file each

87

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Practical No.:05

Introduction to R programming and Data acquisition

Data types, checking type of variable, printing variable and objects.

> x=1

> print(x)

[1] 1

> class(x)

[1] "numeric"

> y="c"

> print(y)

[1] "c"

> is.character(y)

[1] TRUE

> is.integer(y)

[1] FALSE

Vector operations

> x=c(11.3,27.5,33.8)

> y=c(4,5,6)

> x\*y

[1] 45.2 137.5 202.8

> x-y

[1] 7.3 22.5 27.8

> x+y

[1] 15.3 32.5 39.8

# cbind(column bind) and rbind(row bind)

> x = c(1,2,3)

> y = c(11,12,13)

> cbind(x,y)

x y

[1,] 1 11

[2,] 2 12

[3,] 3 13

> rbind(x,y)

[,1] [,2] [,3]

x 1 2 3

y 11 12 13

88

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Matrix operations

# using matrix() function

> m = matrix(c(11,12,13,55,60,65,66,72,78),nrow=3,ncol=3)

> m

[,1] [,2] [,3]

[1,] 11 55 66

[2,] 12 60 72

[3,] 13 65 78

> dim(m)

[1] 3 3

> m = matrix(c(11,12,13,55,60,65,66,72,78),nrow=3,ncol=3,byrow = TRUE)

> m

[,1] [,2] [,3]

[1,] 11 12 13

[2,] 55 60 65

[3,] 66 72 78

# matrix multiplication

> p=3\*m

> p

[,1] [,2] [,3]

[1,] 33 36 39

[2,] 165 180 195

[3,] 198 216 234

> n = matrix(c(4,5,6,13,14,15,24,25,26),nrow = 3,ncol = 3)

> q=m+n

> q

[,1] [,2] [,3]

[1,] 15 25 37

[2,] 60 74 90

[3,] 72 87 104

>

> mdash=t(m)

> mdash

[,1] [,2] [,3]

[1,] 11 55 66

89

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

[2,] 12 60 72

[3,] 13 65 78

Determinant

> s = matrix(c(2,3,4,14,15,16,21,22,23),nrow=3,ncol=3,byrow = TRUE)

> s\_det = det(s)

> s\_det

[1] 2.109424e-15

>

> r\_det = det(m)

> r\_det

[1] 0

Dataframe

> student\_id = c(1,2,3)

> student\_names = c("Tejas","Sakshi","shahu")

> position = c("First","Second","Third")

> data = data.frame(student\_id,student\_names,position)

> data

student\_id student\_names position

1 1 Tejas First

2 2 Sakshi Second

3 3 shahu Third

>

> data$student\_id

[1] 1 2 3

> names(data)

[1] "student\_id" "student\_names" "position"

2-Dimensional table in R

> smoke = matrix(c(51,43,22,92,28,21,68,22,9),ncol=3,byrow=TRUE)

> colnames(smoke) = c("High","Low","Middle")

> rownames(smoke) = c("current","former","never")

> smoke = as.table(smoke)

> smoke

High Low Middle

current 51 43 22

former 92 28 21

90

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

never 68 22 9

Reading and Writing data From csv

> dataT = read.table("C:/Users/Tejas/OneDrive/Documents/Emp.csv",sep = ",", header = T)

> dataT

Emp\_no Emp\_name Salary

1 101 Tejas 25000

2 102 Sakshi 30000

3 103 Shahu 45000

4 104 Uttu 40000

5 105 Shruti 35000

> dim(dataT)

[1] 5 3

> head(dataT,2)

Emp\_no Emp\_name Salary

1 101 Tejas 25000

2 102 Sakshi 30000

> tail(dataT,2)

Emp\_no Emp\_name Salary

4 104 Uttu 40000

5 105 Shruti 35000

Reading and Writing data from Excel using XLConnect

Installing the packages

install.packages("XLConnect")

library(XLConnect)

Read file from directory

> data\_E=read.table("C:/Users/Tejas/OneDrive/Documents/Example1.csv",sep=",",header=T)

> print(data\_E)

Emp\_no Emp\_name Salary

1 1 ram 25000

2 2 sham 45000

3 3 siya 18000

4 4 riya 45000

5 5 tina 49000

> dim(data\_E)

[1] 5 3

> head(data\_E,3)

91

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Emp\_no Emp\_name Salary

1 1 ram 25000

2 2 sham 45000

3 3 siya 18000

> tail(data\_E,2)

Emp\_no Emp\_name Salary

4 4 riya 45000

5 5 tina 49000

> z=data.frame(a=12,b=45,pi=3.142)

> print(z)

a b pi

1 12 45 3.142

>

> write.csv(z,file="data.csv")

Using ConnectXL

> dataX = XLConnect::readWorksheetFromFile("C:/Users/Tejas/OneDrive/Documents/Stude

nt.xlsx",sheet=1)

> dataX

Roll\_No Name Marks

1 1 Tejas 90

2 2 Sakshi 80

3 3 Shahu 85

4 4 Uttu 95

5 5 Shreya 75

92

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Practical No.: 06

Aim: Implementation of Data preprocessing techniques

Data Preprocessing Techniques

1. Naming and Renaming variables, adding a new variable.

2. Dealing with missing data.

3. Dealing with categorical data.

4. Data reduction using subsetting

Naming and Renaming variables:

> setwd("C:\\Users\\ranea\\OneDrive\\Desktop\\ADBMSLab-SEM1\\R")

> getwd()

[1] "C:/Users/ranea/OneDrive/Desktop/ADBMSLab-SEM1/R"

#Open mtcars file

> my\_data<-mtcars

> head(my\_data,5)

mpg cyl disp hp drat wt qsec vs am gear carb

Mazda RX4 21.0 6 160 110 3.90 2.620 16.46 0 1 4 4

Mazda RX4 Wag 21.0 6 160 110 3.90 2.875 17.02 0 1 4 4

Datsun 710 22.8 4 108 93 3.85 2.320 18.61 1 1 4 1

Hornet 4 Drive 21.4 6 258 110 3.08 3.215 19.44 1 0 3 1

Hornet Sportabout 18.7 8 360 175 3.15 3.440 17.02 0 0 3 2

# Renaming columns with dplyr::rename()

> require(dplyr)> my\_data1 = rename(my\_data,horse\_power = hp)

> my\_data1

mpg cyl disp horse\_power drat wt qsec vs am gear carb

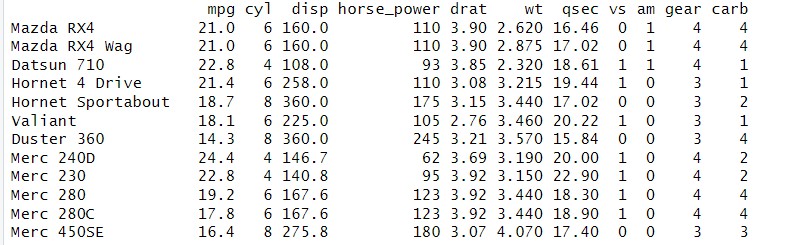
Mazda RX4 21.0 6 160.0 110 3.90 2.620 16.46 0 1 4 4

Mazda RX4 Wag 21.0 6 160.0 110 3.90 2.875 17.02 0 1 4 4

Datsun 710 22.8 4 108.0 93 3.85 2.320 18.61 1 1 4 1

Hornet 4 Drive 21.4 6 258.0 110 3.08 3.215 19.44 1 0 3 1

93



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Hornet Sportabout 18.7 8 360.0 175 3.15 3.440 17.02 0 0 3 2

Valiant 18.1 6 225.0 105 2.76 3.460 20.22 1 0 3 1

Adding a new variable:

> setwd("C:\\Users\\ranea\\OneDrive\\Desktop\\ADBMSLab-SEM1\\R")

> getwd()

[2] "C:/Users/ranea/OneDrive/Desktop/ADBMSLab-SEM1/R"

> my\_data<-mtcars

> head(my\_data,5)

mpg cyl disp hp drat wt qsec vs am gear carb

Mazda RX4 21.0 6 160 110 3.90 2.620 16.46 0 1 4 4

Mazda RX4 Wag 21.0 6 160 110 3.90 2.875 17.02 0 1 4 4

Datsun 710 22.8 4 108 93 3.85 2.320 18.61 1 1 4 1

Hornet 4 Drive 21.4 6 258 110 3.08 3.215 19.44 1 0 3 1

Hornet Sportabout 18.7 8 360 175 3.15 3.440 17.02 0 0 3 2

> require(dplyr)> my\_data1 = rename(my\_data,horse\_power = hp)

> my\_data1

mpg cyl disp horse\_power drat wt qsec vs am gear carb

Mazda RX4 21.0 6 160.0 110 3.90 2.620 16.46 0 1 4 4

Mazda RX4 Wag 21.0 6 160.0 110 3.90 2.875 17.02 0 1 4 4

Datsun 710 22.8 4 108.0 93 3.85 2.320 18.61 1 1 4 1

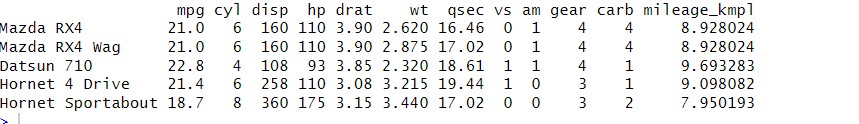
Hornet 4 Drive 21.4 6 258.0 110 3.08 3.215 19.44 1 0 3 1

Hornet Sportabout 18.7 8 360.0 175 3.15 3.440 17.02 0 0 3 2

Valiant 18.1 6 225.0 105 2.76 3.460 20.22 1 0 3 1

# Add a new variable mileage\_kmpl

94



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

> my\_data$mileage\_kmpl <- my\_data$mpg \* 0.425144>

> head(my\_data, 5)

mpg cyl disp hp drat wt qsec vs am gear carb mileage\_kmpl

Mazda RX4 21.0 6 160 110 3.90 2.620 16.46 0 1 4 4 8.928024

Mazda RX4 Wag 21.0 6 160 110 3.90 2.875 17.02 0 1 4 4 8.928024

Datsun 710 22.8 4 108 93 3.85 2.320 18.61 1 1 4 1 9.693283

Hornet 4 Drive 21.4 6 258 110 3.08 3.215 19.44 1 0 3 1 9.098082

Hornet Sportabout 18.7 8 360 175 3.15 3.440 17.02 0 0 3 2 7.950193

Dealing with missing value:

Imputation

The process of estimating or deriving

missing valuesThere are various

methods for imputation

– Imputation of the mean

– Imputation of the median

– Imputation using linear regression models

> setwd("C:\\Users\\ranea\\OneDrive\\Documents")

# Read "emp.csv" with missing values

> data2 <- read.csv(file = "emp.csv", col.names = c("Sno", "NAME", "AGE", "SALARY"))

# Display the original dataset

> print("Original dataset:")

[1] "Original dataset:"

> print(data2)

Sno NAME AGE SALARY

1 101 Anish Rane 20 50000

2 102 Ashutosh NA 62000

95

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

3 103 Sankalp Shukla 22 70000

4 104 Aman Mishra NA NA

5 105 Rutuja 26 54000

#Impute missing values (replace with mean or median)

> data2\_imputed <- transform(data2, AGE = ifelse(is.na(AGE), mean(AGE, na.rm = TRUE),

AGE),+ SALARY = ifelse(is.na(SALARY), median(SALARY, na.rm = TR

UE), SALARY))

# Round the imputed ages to whole numbers

> data2\_imputed$AGE <- round(data2\_imputed$AGE)

# Display the updated dataset

> print("Updated dataset with rounded ages:")

[1] "Updated dataset with rounded ages:"

> print(data2\_imputed)

Sno NAME AGE SALARY

1 101 Anish Rane 20 50000

2 102 Ashutosh 23 62000

3 103 Sankalp Shukla 22 70000

4 104 Aman Mishra 23 58000

5 105 Rutuja 26 54000

Dealing with Categorical data:

The function factor is used to encode a vector as a factor (the terms ‘category’ and

‘enumerated type’ are also used for factors). If argument ordered is TRUE, the factor levels

are assumed to be ordered.

Factors are variables in R which take on a limited number of different values;

Convert Character into Factor(categorical data)

96

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

> setwd("C:\\Users\\ranea\\OneDrive\\Documents")

# Read "emp.csv"

> data2 <- read.csv(file = "emp.csv", col.names = c("Sno", "NAME", "AGE", "GENDER", "

SALARY"))>

# Display the original dataset

> print("Original dataset:")

[1] "Original dataset:"

> print(data2)

Sno NAME AGE GENDER SALARY

1 101 Anish Rane 20 Male 50000

2 102 Ashutosh 23 Male 62000

3 103 Sanika 22 Female 70000

4 104 Neha 21 Female 65000

5 105 Rutuja 26 Male 54000

# Convert 'GENDER' column to a factor

> data2$GENDER <- as.factor(data2$GENDER)

# Display the updated dataset

> print("Updated dataset with categorical 'GENDER':")

[1] "Updated dataset with categorical 'GENDER':

> print(data2)

Sno NAME AGE GENDER SALARY

1 101 Anish Rane 20 Male 50000

2 102 Ashutosh 23 Male 62000

3 103 Sanika 22 Female 70000

4 104 Neha 21 Female 65000

5 105 Rutuja 26 Male 54000

97

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Data reduction using subsetting;

Data reduction using subsetting involves selecting a subset of the original dataset based on

certain criteria

Data reduction using subsetting on the mtcars dataset in R:

# Original dataset

> print("Original dataset:")

[1] "Original dataset:"

> print(mtcars)

mpg cyl disp hp drat wt qsec vs am gear carb

Mazda RX4 21.0 6 160.0 110 3.90 2.620 16.46 0 1 4 4

Mazda RX4 Wag 21.0 6 160.0 110 3.90 2.875 17.02 0 1 4 4

Datsun 710 22.8 4 108.0 93 3.85 2.320 18.61 1 1 4 1

Hornet 4 Drive 21.4 6 258.0 110 3.08 3.215 19.44 1 0 3 1

Hornet Sportabout 18.7 8 360.0 175 3.15 3.440 17.02 0 0 3 2

Valiant 18.1 6 225.0 105 2.76 3.460 20.22 1 0 3 1

Duster 360 14.3 8 360.0 245 3.21 3.570 15.84 0 0 3 4

Merc 240D 24.4 4 146.7 62 3.69 3.190 20.00 1 0 4 2

Merc 230 22.8 4 140.8 95 3.92 3.150 22.90 1 0 4 2

Merc 280 19.2 6 167.6 123 3.92 3.440 18.30 1 0 4 4

Merc 280C 17.8 6 167.6 123 3.92 3.440 18.90 1 0 4 4

Merc 450SE 16.4 8 275.8 180 3.07 4.070 17.40 0 0 3 3

Merc 450SL 17.3 8 275.8 180 3.07 3.730 17.60 0 0 3 3

Merc 450SLC 15.2 8 275.8 180 3.07 3.780 18.00 0 0 3 3

Cadillac Fleetwood 10.4 8 472.0 205 2.93 5.250 17.98 0 0 3 4

Lincoln Continental 10.4 8 460.0 215 3.00 5.424 17.82 0 0 3 4

Chrysler Imperial 14.7 8 440.0 230 3.23 5.345 17.42 0 0 3 4

Fiat 128 32.4 4 78.7 66 4.08 2.200 19.47 1 1 4 1

Honda Civic 30.4 4 75.7 52 4.93 1.615 18.52 1 1 4 2

98

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Toyota Corolla 33.9 4 71.1 65 4.22 1.835 19.90 1 1 4 1

Toyota Corona 21.5 4 120.1 97 3.70 2.465 20.01 1 0 3 1

Dodge Challenger 15.5 8 318.0 150 2.76 3.520 16.87 0 0 3 2

AMC Javelin 15.2 8 304.0 150 3.15 3.435 17.30 0 0 3 2

Camaro Z28 13.3 8 350.0 245 3.73 3.840 15.41 0 0 3 4

Pontiac Firebird 19.2 8 400.0 175 3.08 3.845 17.05 0 0 3 2

Fiat X1-9 27.3 4 79.0 66 4.08 1.935 18.90 1 1 4 1

Porsche 914-2 26.0 4 120.3 91 4.43 2.140 16.70 0 1 5 2

Lotus Europa 30.4 4 95.1 113 3.77 1.513 16.90 1 1 5 2

Ford Pantera L 15.8 8 351.0 264 4.22 3.170 14.50 0 1 5 4

Ferrari Dino 19.7 6 145.0 175 3.62 2.770 15.50 0 1 5 6

Maserati Bora 15.0 8 301.0 335 3.54 3.570 14.60 0 1 5 8

Volvo 142E 21.4 4 121.0 109 4.11 2.780 18.60 1 1 4 2

# Data reduction: Selecting cars with mpg greater than 20

> reduced\_data <- mtcars[mtcars$mpg > 20, ]

# Display the reduced dataset

> print("Reduced dataset (cars with mpg > 20):")

[1] "Reduced dataset (cars with mpg > 20):"

> print(reduced\_data)

mpg cyl disp hp drat wt qsec vs am gear carb

Mazda RX4 21.0 6 160.0 110 3.90 2.620 16.46 0 1 4 4

Mazda RX4 Wag 21.0 6 160.0 110 3.90 2.875 17.02 0 1 4 4

Datsun 710 22.8 4 108.0 93 3.85 2.320 18.61 1 1 4 1

Hornet 4 Drive 21.4 6 258.0 110 3.08 3.215 19.44 1 0 3 1

Merc 240D 24.4 4 146.7 62 3.69 3.190 20.00 1 0 4 2

Merc 230 22.8 4 140.8 95 3.92 3.150 22.90 1 0 4 2

Fiat 128 32.4 4 78.7 66 4.08 2.200 19.47 1 1 4 1

Honda Civic 30.4 4 75.7 52 4.93 1.615 18.52 1 1 4 2

Toyota Corolla 33.9 4 71.1 65 4.22 1.835 19.90 1 1 4 1

Toyota Corona 21.5 4 120.1 97 3.70 2.465 20.01 1 0 3 1

Fiat X1-9 27.3 4 79.0 66 4.08 1.935 18.90 1 1 4 1

99

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Porsche 914-2 26.0 4 120.3 91 4.43 2.140 16.70 0 1 5 2

Lotus Europa 30.4 4 95.1 113 3.77 1.513 16.90 1 1 5 2

Volvo 142E 21.4 4 121.0 109 4.11 2.780 18.60 1 1 4 2

Data reduction using subsetting on the emp.csv dataset in R:

> setwd("C:\\Users\\ranea\\OneDrive\\Documents")

# Read "emp.csv" with missing values

> data\_emp <- read.csv(file = "emp.csv", header = TRUE )

> print("Original dataset:")

[1] "Original dataset:"

> print(data\_emp)

sr Name Age Gender Salary

1 101 Anish Rane 20 Male 50000

2 102 Ashutosh 23 Male 62000

3 103 Sanika 22 Female 70000

4 104 Neha 21 Female 65000

5 105 Rutuja 26 Male 54000

# Data reduction: Selecting employees with age greater than 22

> reduced\_data\_emp <- data\_emp[data\_emp$Age > 22, ]

# Display the reduced dataset

> print("Reduced dataset (employees with age > 22):")

[1] "Reduced dataset (employees with age > 22):"

> print(reduced\_data\_emp)

sr Name Age Gender Salary

2 102 Ashutosh 23 Male 62000

5 105 Rutuja26 Male 54000

100

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Practical No.: 07

Aim: Implementation and analysis of Linear regression through

graphical methods.

Simple linear regression

> ads <- read.csv('C:/Users/Tejas/Downloads/Advertising.csv')

> View(ads)

> nrow(ads)

[1] 200

> ncol(ads)

[1] 4

> colnames(ads)

[1] "TV" "Radio" "Newspaper" "Sales"

> TV <- ads$TV

> Sales <-ads$Sales

> plot(TV,Sales)

> plot (TV, Sales, pch=16, cex=1, col='blue', main='TV vs Sales', xlab='TV', ylab='sales')

> model <- lm(Sales ~ TV)

> summary(model)

Call:

lm(formula = Sales ~ TV)

Residuals:

Min 1Q Median 3Q Max

-8.3860 -1.9545 -0.1913 2.0671 7.2124

Coefficients:

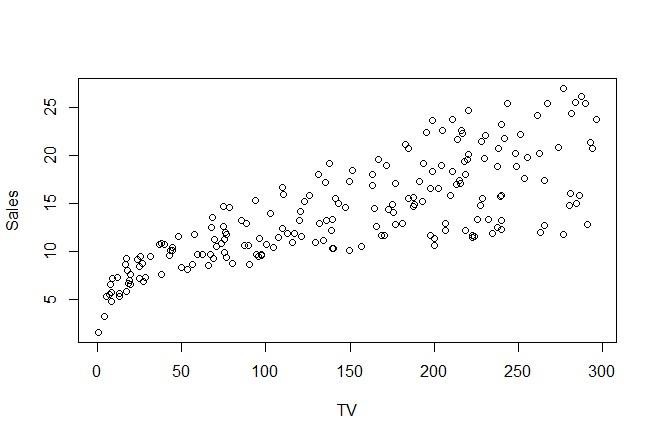
Estimate Std. Error t value Pr(>|t|)

(Intercept) 7.032594 0.457843 15.36 <2e-16 \*\*\*

TV 0.047537 0.002691 17.67 <2e-16 \*\*\*

---

101



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.259 on 198 degrees of freedom

Multiple R-squared: 0.6119, Adjusted R-squared: 0.6099

F-statistic: 312.1 on 1 and 198 DF, p-value: < 2.2e-16

> coefficients(model)

(Intercept) TV

7.03259355 0.04753664

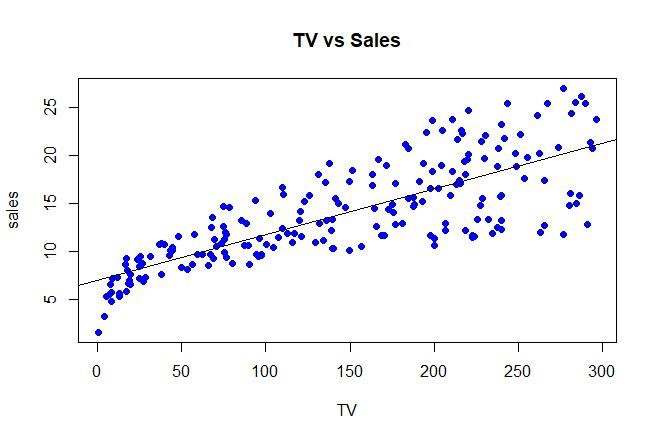
> coef(model)

(Intercept) TV

7.03259355 0.04753664

> abline(model)

102



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Multi linear regression

> data\_in = read.csv('C:/Users/Tejas/Downloads/Advertising.csv')

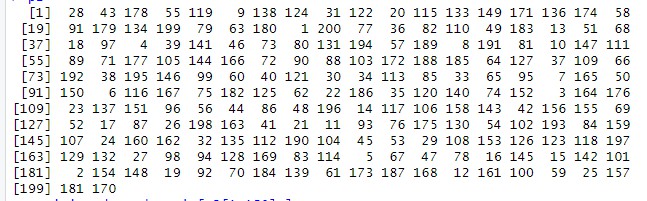
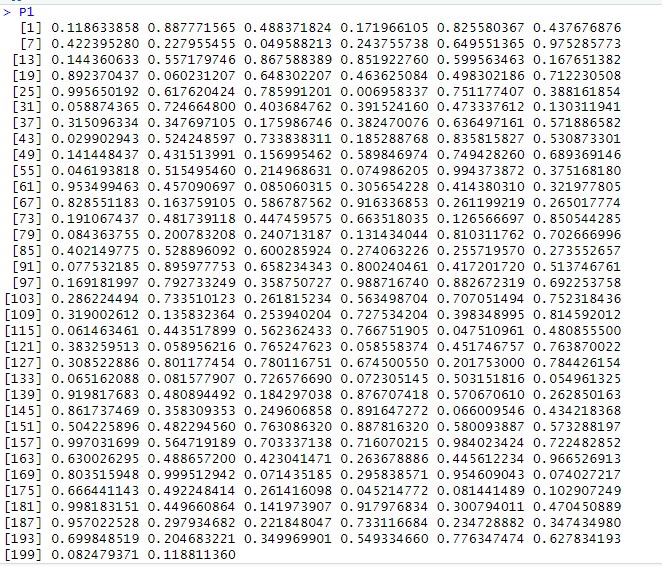
> plot(data\_in$Newspaper,data\_in$Radio)

> #Data Partition

> P1 = runif(200)

> P1

103



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

> p2 = order(P1)

> p2

> training\_ds = data\_in[p2[1:150],]

>

> test\_ds = data\_in[p2[150:200],]

> #model building

> multiple\_line\_regression = lm(Newspaper ~ Radio + TV, data = training\_ds)

104

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

> abline(multiple\_line\_regression)

> summary(multiple\_line\_regression)

Call:

lm(formula = Newspaper ~ Radio + TV, data = training\_ds)

Residuals:

Min 1Q Median 3Q Max

-40.320 -15.887 0.175 12.927 76.551

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) 17.203771 4.486597 3.834 0.000186 \*\*\*

Radio 0.543624 0.120626 4.507 1.33e-05 \*\*\*

TV 0.005146 0.019975 0.258 0.797053

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 21.5 on 147 degrees of freedom

Multiple R-squared: 0.1215, Adjusted R-squared: 0.1095

F-statistic: 10.16 on 2 and 147 DF, p-value: 7.338e-05

> plot(multiple\_line\_regression)

Hit <Return> to see next plot:

Hit <Return> to see next plot:

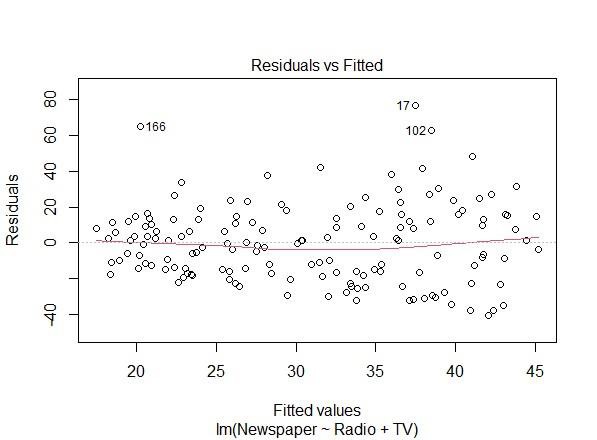
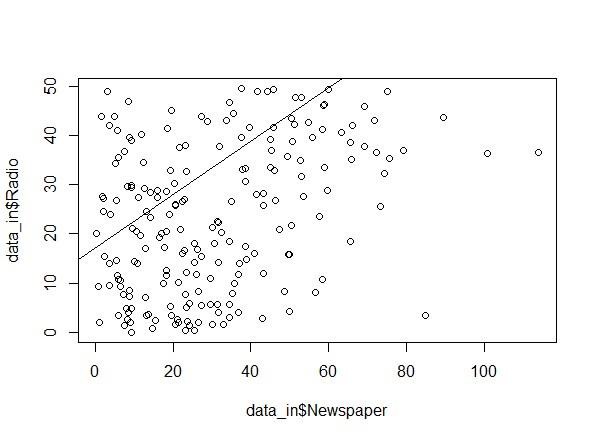
Hit <Return> to see next plot:

Hit <Return> to see next plot:

> pred\_value = predict(multiple\_line\_regression, newdata = test\_ds)

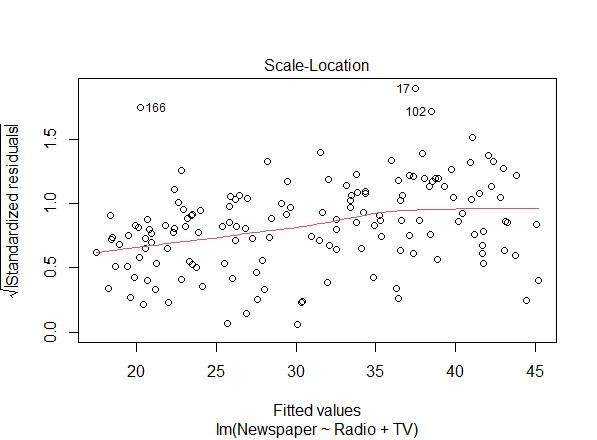
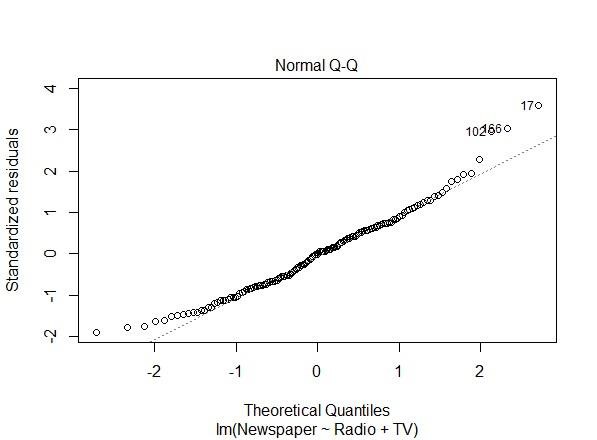
> test\_ds$Predicted\_values = pred\_value

105



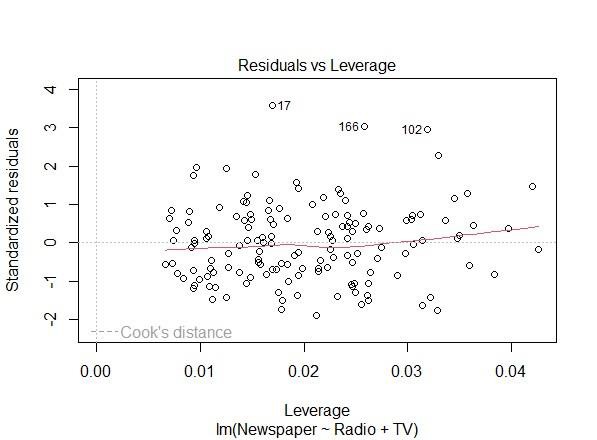
MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

106



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

107



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

108

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Practical No.:08

Aim: Implementation and analysis of Classification algorithms

Decision tree

> install.packages("rpart")

> install.packages("rpart.plot")

> library(rpart)

> library(rpart.plot)

> data\_in=read.csv('C:/Users/acer/Downloads/data1.csv')

> tree<-rpart(Height~Gender+Weight,data\_in)

> a<-data.frame(Gender=c("Male"),Weight=c(85))

> result<-predict(tree,a)

> print(result)

1

180

> tree<-rpart(Gender~Height+Weight,data\_in)

> a<-data.frame(Height=c(190),Weight=c(85))

> result<-predict(tree,a)

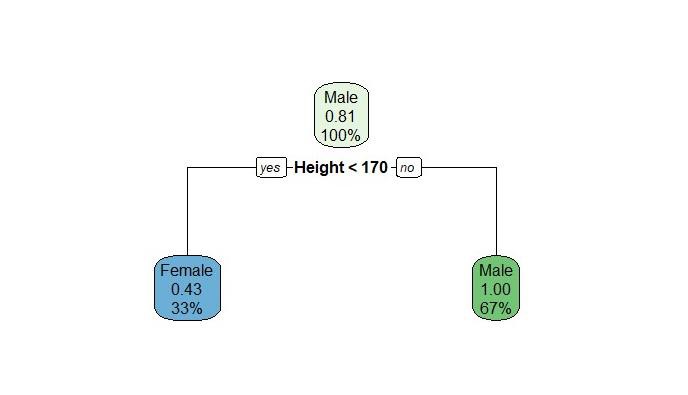
> print(result)

Female Male

1 0 1

> rpart.plot(tree)

109



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Naive bayes

> install.packages("e1071")

> install.packages("klaR")

> install.packages("caret")

> data("iris")

> head(iris)

Sepal.Length Sepal.Width Petal.Length Petal.Width Species

1 5.1 3.5 1.4 0.2 setosa

2 4.9 3.0 1.4 0.2 setosa

3 4.7 3.2 1.3 0.2 setosa

4 4.6 3.1 1.5 0.2 setosa

5 5.0 3.6 1.4 0.2 setosa

6 5.4 3.9 1.7 0.4 setosa

> unique(iris$Species)

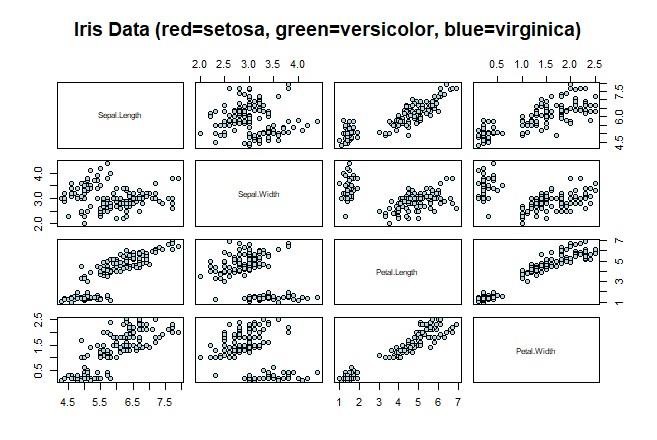
[1] setosa versicolor virginica

Levels: setosa versicolor virginica

> pairs(iris[1:4], main="Iris Data (red=setosa, green=versicolor, blue=virginica)", pch=21, bg

="lightblue")

110



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

> # Load the caret package

> library(caret)

> # Set the seed for reproducibility

> set.seed(123)

>

> # Split the data into training and testing sets

> index <- sample(nrow(iris), floor(nrow(iris) \* 0.7))

> train <- iris[index, ]

> test <- iris[-index, ]

>

> # Create xTrain, yTrain, xTest, yTest

> xTrain <- train[, -5]

> yTrain <- train$Species

>

> xTest <- test[, -5]

> yTest <- test$Species

>

> # Train a Naive Bayes model using caret

111

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

> model <- train(xTrain, yTrain, method = 'nb', trControl = trainControl(method = 'cv', numb

er = 10))

>

> # Print the trained model

> print(model)

Naive Bayes

105 samples

4 predictor

3 classes: 'setosa', 'versicolor', 'virginica'

No pre-processing

Resampling: Cross-Validated (10 fold)

Summary of sample sizes: 94, 95, 95, 95, 95, 94, ...

Resampling results across tuning parameters:

usekernel Accuracy Kappa

FALSE 0.9409091 0.9106552

TRUE 0.9409091 0.9106552

Tuning parameter 'fL' was held constant at a value of 0

Tuning

parameter 'adjust' was held constant at a value of 1

Accuracy was used to select the optimal model using the largest value.

The final values used for the model were fL = 0, usekernel = FALSE and

adjust = 1.

K Nearest Neighbor

> install.packages("class")

> library(class)

> df <- data(iris)

> head(iris)

Sepal.Length Sepal.Width Petal.Length Petal.Width Species

1 5.1 3.5 1.4 0.2 setosa

112

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

2 4.9 3.0 1.4 0.2 setosa

3 4.7 3.2 1.3 0.2 setosa

4 4.6 3.1 1.5 0.2 setosa

5 5.0 3.6 1.4 0.2 setosa

6 5.4 3.9 1.7 0.4 setosa

> ran <- sample(1:nrow(iris), 0.9 \* nrow(iris))

> nor <-function(x) { (x -min(x))/(max(x)-min(x))}

> iris\_norm <- as.data.frame(lapply(iris[,c(1,2,3,4)], nor))

> summary(iris\_norm)

Sepal.Length Sepal.Width Petal.Length Petal.Width

Min. :0.0000 Min. :0.0000 Min. :0.0000 Min. :0.00000

1st Qu.:0.2222 1st Qu.:0.3333 1st Qu.:0.1017 1st Qu.:0.08333

Median :0.4167 Median :0.4167 Median :0.5678 Median :0.50000

Mean :0.4287 Mean :0.4406 Mean :0.4675 Mean :0.45806

3rd Qu.:0.5833 3rd Qu.:0.5417 3rd Qu.:0.6949 3rd Qu.:0.70833

Max. :1.0000 Max. :1.0000 Max. :1.0000 Max. :1.00000

> iris\_train <- iris\_norm[ran,]

> iris\_test <- iris\_norm[-ran,]

> iris\_target\_category <- iris[ran,5]

> iris\_test\_category <- iris[-ran,5]

> library(class)

> pr <- knn(iris\_train,iris\_test,cl=iris\_target\_category,k=13)

> tab <- table(pr,iris\_test\_category)

> accuracy <- function(x){sum(diag(x)/(sum(rowSums(x)))) \* 100}

> accuracy(tab)

[1] 100

ID3

> library(rpart)>

> # Load the iris dataset

> data(iris)

> # Explore the structure and summary of the dataset

> str(iris)

113

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

'data.frame': 150 obs. of 5 variables:

$ Sepal.Length: num 5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...

$ Sepal.Width : num 3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...

$ Petal.Length: num 1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...

$ Petal.Width : num 0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...

$ Species : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1 1 ...

>

> summary(iris)

Sepal.Length Sepal.Width Petal.Length Petal.Width Species

Min. :4.300 Min. :2.000 Min. :1.000 Min. :0.100 setosa :50

1st Qu.:5.100 1st Qu.:2.800 1st Qu.:1.600 1st Qu.:0.300 versicolor:50

Median :5.800 Median :3.000 Median :4.350 Median :1.300 virginica :50

Mean :5.843 Mean :3.057 Mean :3.758 Mean :1.199

3rd Qu.:6.400 3rd Qu.:3.300 3rd Qu.:5.100 3rd Qu.:1.800

Max. :7.900 Max. :4.400 Max. :6.900 Max. :2.500

> # Set a seed for reproducibility

> set.seed(123)

> # Create an index for splitting the dataset

> index <- sample(1:nrow(iris), 0.7 \* nrow(iris))

> # Create training and testing sets

> train\_data <- iris[index, ]

> test\_data <- iris[-index, ]

> # Train the ID3-like decision tree model using rpart

> id3\_model <- rpart(Species ~ ., data = train\_data, method = "class")

> # Make predictions on the test set

> predictions <- predict(id3\_model, newdata = test\_data, type = "class")

>

> # Confusion matrix

> conf\_matrix <- table(predictions, test\_data$Species)

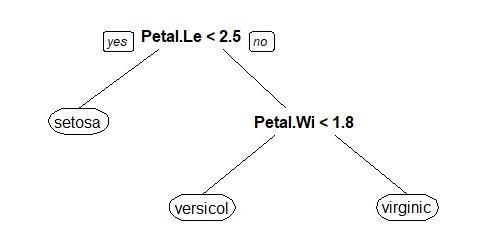
> print(conf\_matrix)

predictions setosa versicolor virginica

setosa 14 0 0

versicolor 0 18 1

114



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

virginica 0 0 12

>

> # Calculate accuracy

> accuracy <- sum(diag(conf\_matrix)) / sum(conf\_matrix)

> print(paste("Accuracy:", accuracy))

[1] "Accuracy: 0.977777777777778"

> # Plot the decision tree (requires the rpart.plot package)

> install.packages("rpart.plot")

> library(rpart.plot)

> # Plot the decision tree

> prp(id3\_model)

C4.5

> library(C50)

> # Load the iris dataset

> data(iris)

> # Explore the structure and summary of the dataset

> str(iris)

'data.frame': 150 obs. of 5 variables:

$ Sepal.Length: num 5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...

$ Sepal.Width : num 3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...

$ Petal.Length: num 1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...

$ Petal.Width : num 0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...

$ Species : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1 1 ...

115

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

> summary(iris)

Sepal.Length Sepal.Width Petal.Length Petal.Width Species

Min. :4.300 Min. :2.000 Min. :1.000 Min. :0.100 setosa :50

1st Qu.:5.100 1st Qu.:2.800 1st Qu.:1.600 1st Qu.:0.300 versicolor:50

Median :5.800 Median :3.000 Median :4.350 Median :1.300 virginica :50

Mean :5.843 Mean :3.057 Mean :3.758 Mean :1.199

3rd Qu.:6.400 3rd Qu.:3.300 3rd Qu.:5.100 3rd Qu.:1.800

Max. :7.900 Max. :4.400 Max. :6.900 Max. :2.500

> # Set a seed for reproducibility

> set.seed(123)

> # Create an index for splitting the dataset

> index <- sample(1:nrow(iris), 0.7 \* nrow(iris))

> # Create training and testing sets

> train\_data <- iris[index, ]

> test\_data <- iris[-index, ]

> # Train the C5.0 decision tree model

> c50\_model <- C5.0(Species ~ ., data = train\_data)

> # Make predictions on the test set

> predictions <- predict(c5\_model, newdata = test\_data)

> # Confusion matrix

> conf\_matrix <- table(predictions, test\_data$Species)

> print(conf\_matrix)

predictions setosa versicolor virginica

setosa 14 0 0

versicolor 0 17 0

virginica 0 1 13

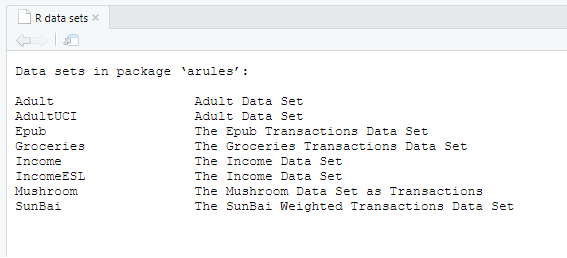
> # Calculate accuracy

> accuracy <- sum(diag(conf\_matrix)) / sum(conf\_matrix)

> print(paste("Accuracy:", accuracy))

[1] "Accuracy: 0.977777777777778"

116



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Practical No.: 09

Aim: Implementation and analysis of Apriori Algorithm using

Market Basket Analysis

Market Basket Analysis

> install.packages("arules")

> library(arules)

> install.packages("arulesViz")

> library(arulesViz)

> install.packages("RColorBrewer")

> library(RColorBrewer)

> data(package="arules")

> data("Groceries")

> rules<-apriori(Groceries,parameter = list(supp=0.01, conf=0.2))

Apriori

Parameter specification:

confidence minval smax arem aval originalSupport maxtime support minlen

0.2 0.1 1 none FALSE TRUE 5 0.01 1

maxlen target ext

10 rules TRUE

Algorithmic control:

filter tree heap memopt load sort verbose

117

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

0.1 TRUE TRUE FALSE TRUE 2 TRUE

Absolute minimum support count: 98

set item appearances ...[0 item(s)] done [0.00s].

set transactions ...[169 item(s), 9835 transaction(s)] done [0.01s].

sorting and recoding items ... [88 item(s)] done [0.00s].

creating transaction tree ... done [0.00s].

checking subsets of size 1 2 3 4 done [0.00s].

writing ... [232 rule(s)] done [0.00s].

creating S4 object ... done [0.00s].

>

> inspect(rules[1:10])

lhs rhs support confidence coverage

[1] {} => {whole milk} 0.25551601 0.2555160 1.00000000

[2] {hard cheese} => {whole milk} 0.01006609 0.4107884 0.02450432

[3] {butter milk} => {other vegetables} 0.01037112 0.3709091 0.02796136

[4] {butter milk} => {whole milk} 0.01159126 0.4145455 0.02796136

[5] {ham} => {whole milk} 0.01148958 0.4414062 0.02602949

[6] {sliced cheese} => {whole milk} 0.01077783 0.4398340 0.02450432

[7] {oil} => {whole milk} 0.01128622 0.4021739 0.02806304

[8] {onions} => {other vegetables} 0.01423488 0.4590164 0.03101169

[9] {onions} => {whole milk} 0.01209964 0.3901639 0.03101169

[10] {berries} => {yogurt} 0.01057448 0.3180428 0.03324860

lift count

[1] 1.000000 2513

[2] 1.607682 99

[3] 1.916916 102

[4] 1.622385 114

[5] 1.727509 113

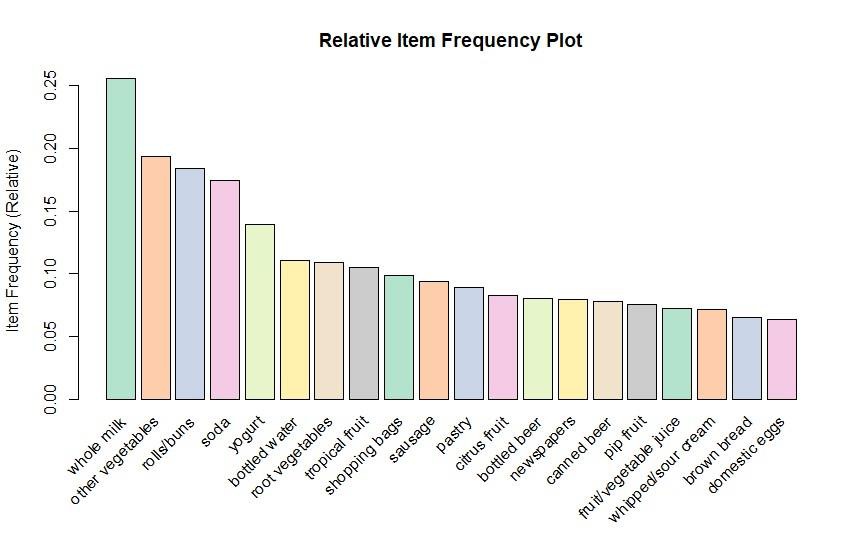
[6] 1.721356 106

[7] 1.573968 111

[8] 2.372268 140

[9] 1.526965 119

118



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

[10] 2.279848 104

> arules::itemFrequencyPlot(Groceries,topN=20,

+ col=brewer.pal(8,'Pastel2'),

+ main = 'Relative Item Frequency Plot',

+ type="relative",

+ ylab="Item Frequency (Relative)")

119

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Practical No.: 10

Aim: Implementation and analysis of clustering algorithms like

K-means, Agglomarative

1. K-Means

> install.packages("cluster")

> install.packages("factoextra")

> install.packages("gridExtra")

> # Loading Libraries

> library(cluster)

> library(factoextra)

> data()

> df<- animals

> #na.omit is used to remove missing values.

> df<- na.omit(df)

> #head print the default no. of rows.

> head(df)

war fly ver end gro hai

ant 1 1 1 1 2 1

bee 1 2 1 1 2 2

cat 2 1 2 1 1 2

cpl 1 1 1 1 1 2

chi 2 1 2 2 2 2

cow 2 1 2 1 2 2

> head(df,n=10)

war fly ver end gro hai

ant 1 1 1 1 2 1

bee 1 2 1 1 2 2

cat 2 1 2 1 1 2

cpl 1 1 1 1 1 2

chi 2 1 2 2 2 2

cow 2 1 2 1 2 2

120

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

duc 2 2 2 1 2 1

eag 2 2 2 2 1 1

ele 2 1 2 2 2 1

fly 1 2 1 1 1 1

> k2<- kmeans(df, centers = 2, nstart = 25)

> str(k2)

List of 9

$ cluster : Named int [1:15] 2 2 1 2 1 1 1 1 1 2 ...

..- attr(\*, "names")= chr [1:15] "ant" "bee" "cat" "cpl" ...

$ centers : num [1:2, 1:6] 2 1 1.22 1.33 2 ...

..- attr(\*, "dimnames")=List of 2

.. ..$ : chr [1:2] "1" "2"

.. ..$ : chr [1:6] "war" "fly" "ver" "end" ...

$ totss : num 19.9

$ withinss : num [1:2] 7.56 5.5

$ tot.withinss: num 13.1

$ betweenss : num 6.81

$ size : int [1:2] 9 6

$ iter : int 1

$ ifault : int 0

- attr(\*, "class")= chr "kmeans"

> k2

K-means clustering with 2 clusters of sizes 9, 6

Cluster means:

war fly ver end gro hai

1 2 1.222222 2.000000 1.555556 1.777778 1.555556

2 1 1.333333 1.333333 1.000000 1.500000 1.333333

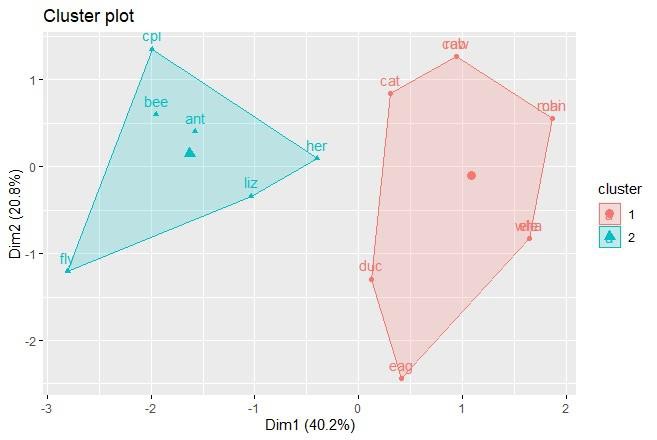
Clustering vector:

ant bee cat cpl chi cow duc eag ele fly her liz man rab wha

2 2 1 2 1 1 1 1 1 2 2 2 1 1 1

Within cluster sum of squares by cluster:

121



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

[1] 7.555556 5.500000

(between\_SS / total\_SS = 34.3 %)

Available components:

[1] "cluster" "centers" "totss" "withinss" "tot.withinss"

[6] "betweenss" "size" "iter" "ifault"

> fviz\_cluster(k2, data=df)

> fviz\_cluster(k2, data=df)

> k3<- kmeans(df, centers = 3, nstart = 25)

> k4<- kmeans(df, centers = 4, nstart = 25)

> k5<- kmeans(df, centers = 5, nstart = 25)

> p1<- fviz\_cluster(k2, geom="point", data = df) + ggtitle("k2")

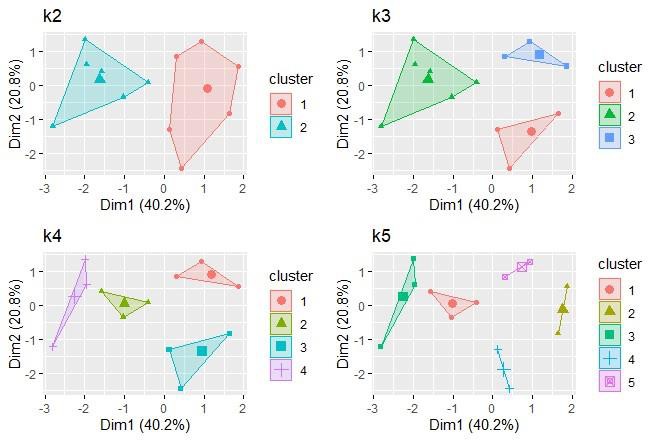
> p2<- fviz\_cluster(k3, geom="point", data = df) + ggtitle("k3")

> p3<- fviz\_cluster(k4, geom="point", data = df) + ggtitle("k4")

> p4<- fviz\_cluster(k5, geom="point", data = df) + ggtitle("k5")

> library(gridExtra)

122



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Warning message:

package ‘gridExtra’ was built under R version 4.2.3

> grid.arrange(p1,p2,p3,p4, nrow=2)

2. Agglomerative Hierarchical Clustering.

A. Maximum or complete linkage clustering

> install.packages("cluster")

> install.packages("factoextra")

> library(tidyverse) # data manipulation

> library(cluster) # clustering algorithms

> library(factoextra) # clustering visualization

> df <- animals

> df <- na.omit(df)

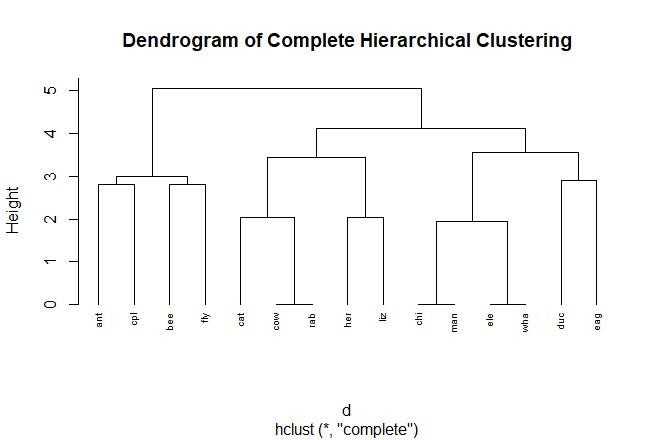
> df <- scale(df)

> head(df)

war fly ver end gro hai

ant -1.1832160 -0.5825753 -1.6020820 -0.6831301 0.6831301 -0.9036961

123



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

bee -1.1832160 1.6020820 -1.6020820 -0.6831301 0.6831301 1.0327956

cat 0.7888106 -0.5825753 0.5825753 -0.6831301 -1.3662601 1.0327956

cpl -1.1832160 -0.5825753 -1.6020820 -0.6831301 -1.3662601 1.0327956

chi 0.7888106 -0.5825753 0.5825753 1.3662601 0.6831301 1.0327956

cow 0.7888106 -0.5825753 0.5825753 -0.6831301 0.6831301 1.0327956

> # Dissimilarity matrix

> d <- dist(df, method = "euclidean")

> # Hierarchical clustering using Complete Linkage

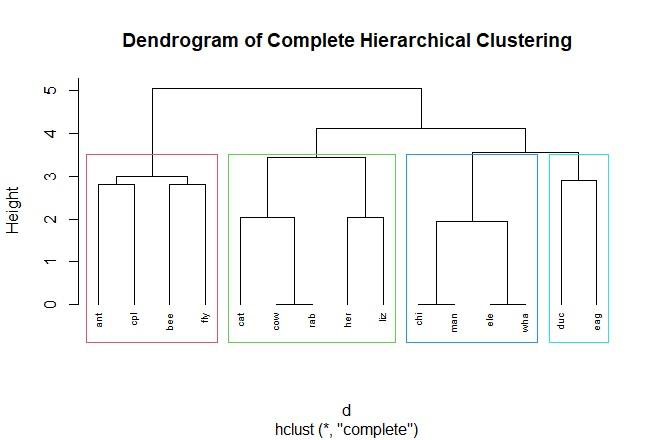
> hc1 <- hclust(d, method = "complete" )

> # Plot the obtained dendrogram

> plot(hc1, cex = 0.6, hang = -1, main= "Dendrogram of Complete Hierarchical Clustering")

> rect.hclust(hc1, k = 4, border = 2:5)

124



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

B. Minimum or single linkage clustering

For Single Hierarchical Clustering change the method name.

> install.packages("cluster")

> install.packages("factoextra")

> library(tidyverse) # data manipulation

> library(cluster) # clustering algorithms

> library(factoextra) # clustering visualization

> df <- animals

> df <- na.omit(df)

> df <- scale(df)

> head(df)

war fly ver end gro hai

ant -1.1832160 -0.5825753 -1.6020820 -0.6831301 0.6831301 -0.9036961

bee -1.1832160 1.6020820 -1.6020820 -0.6831301 0.6831301 1.0327956

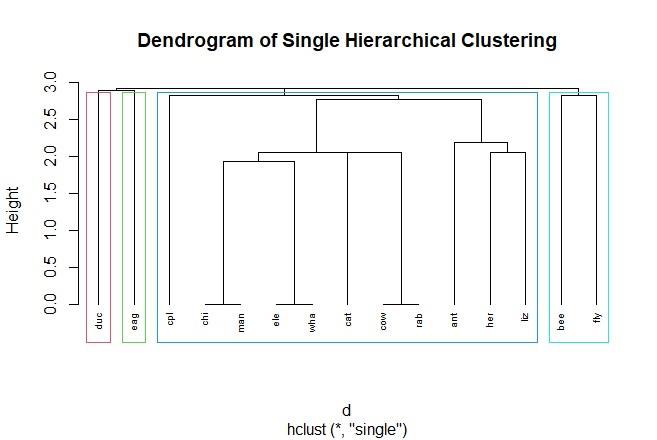
cat 0.7888106 -0.5825753 0.5825753 -0.6831301 -1.3662601 1.0327956

cpl -1.1832160 -0.5825753 -1.6020820 -0.6831301 -1.3662601 1.0327956

chi 0.7888106 -0.5825753 0.5825753 1.3662601 0.6831301 1.0327956

cow 0.7888106 -0.5825753 0.5825753 -0.6831301 0.6831301 1.0327956

125



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

> # Dissimilarity matrix

> d <- dist(df, method = "euclidean")

> # Hierarchical clustering using Single Linkage

> hc1 <- hclust(d, method = "single" )

> # Plot the obtained dendrogram

> plot(hc1, cex = 0.6, hang = -1, main= "Dendrogram of Single Hierarchical Clustering")

> rect.hclust(hc1, k = 4, border = 2:5)

C. Mean or average linkage clustering

For Average Hierarchical Clustering change the method name.

> install.packages("cluster")

> install.packages("factoextra")

> library(tidyverse) # data manipulation

> library(cluster) # clustering algorithms

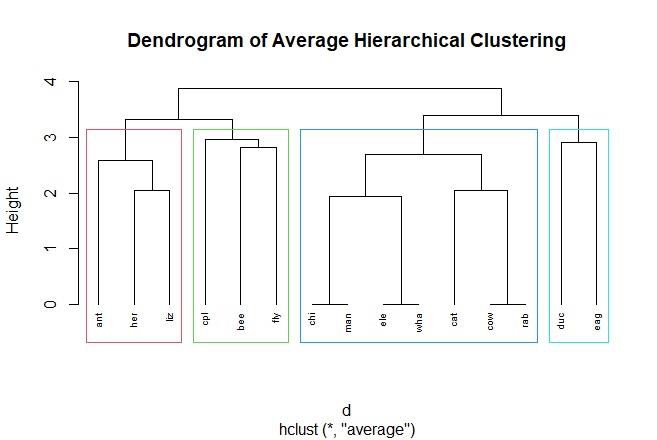
> library(factoextra) # clustering visualization

> df <- animals

> df <- na.omit(df)

> df <- scale(df)

126



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

> head(df)

war fly ver end gro hai

ant -1.1832160 -0.5825753 -1.6020820 -0.6831301 0.6831301 -0.9036961

bee -1.1832160 1.6020820 -1.6020820 -0.6831301 0.6831301 1.0327956

cat 0.7888106 -0.5825753 0.5825753 -0.6831301 -1.3662601 1.0327956

cpl -1.1832160 -0.5825753 -1.6020820 -0.6831301 -1.3662601 1.0327956

chi 0.7888106 -0.5825753 0.5825753 1.3662601 0.6831301 1.0327956

cow 0.7888106 -0.5825753 0.5825753 -0.6831301 0.6831301 1.0327956

> # Dissimilarity matrix

> d <- dist(df, method = "euclidean")

> # Hierarchical clustering using Average Linkage

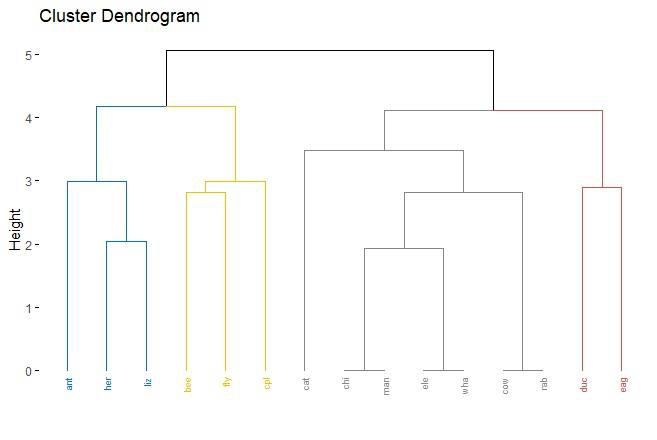
> hc1 <- hclust(d, method = "average" )

> # Plot the obtained dendrogram

> plot(hc1, cex = 0.6, hang = -1, main= "Dendrogram of Average Hierarchical Clustering")

> rect.hclust(hc1, k = 4, border = 2:5)

127



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

3. Divisive Hierarchical Clustering.

> # Compute diana()

> library(cluster)

> df<- animals

> df<-na.omit(df)

> df<- scale(df)

> head(df)

war fly ver end gro hai

ant -1.1832160 -0.5825753 -1.6020820 -0.6831301 0.6831301 -0.9036961

bee -1.1832160 1.6020820 -1.6020820 -0.6831301 0.6831301 1.0327956

cat 0.7888106 -0.5825753 0.5825753 -0.6831301 -1.3662601 1.0327956

cpl -1.1832160 -0.5825753 -1.6020820 -0.6831301 -1.3662601 1.0327956

chi 0.7888106 -0.5825753 0.5825753 1.3662601 0.6831301 1.0327956

cow 0.7888106 -0.5825753 0.5825753 -0.6831301 0.6831301 1.0327956

> dhc <- diana(df)

> # Plot the dendrogram

> library(factoextra)

> # Cut in four groups, Color palette

> fviz\_dend(dhc, cex = 0.5, k = 4, palette = "jco")

128

MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

Agglomerative Clustering

> head(iris)

Sepal.Length Sepal.Width Petal.Length Petal.Width Species

1 5.1 3.5 1.4 0.2 setosa

2 4.9 3.0 1.4 0.2 setosa

3 4.7 3.2 1.3 0.2 setosa

4 4.6 3.1 1.5 0.2 setosa

5 5.0 3.6 1.4 0.2 setosa

6 5.4 3.9 1.7 0.4 setosa

> clusters <- hclust(dist(iris[, 3:4]))

> plot(clusters)

> clusterCut<- cutree(clusters, 3)

> table(clusterCut, iris$Species)

clusterCut setosa versicolor virginica

1 50 0 0

2 0 21 50

3 0 29 0

> clusters <- hclust(dist(iris[,3:4]), method = 'average')

> plot(clusters)

> clusterCut <- cutree(clusters, 3)

> table(clusterCut, iris$Species)

clusterCut setosa versicolor virginica

1 50 0 0

2 0 45 1

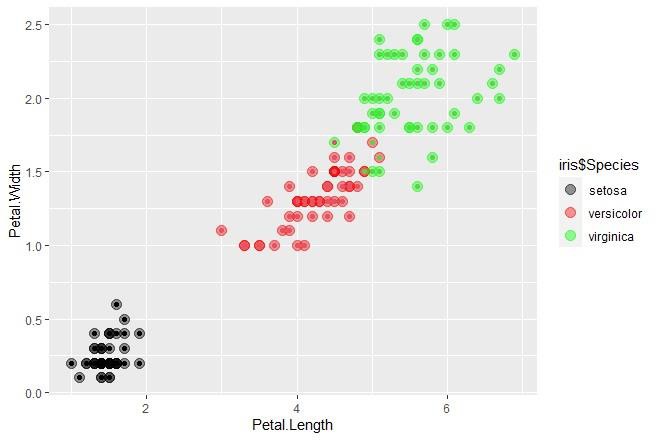
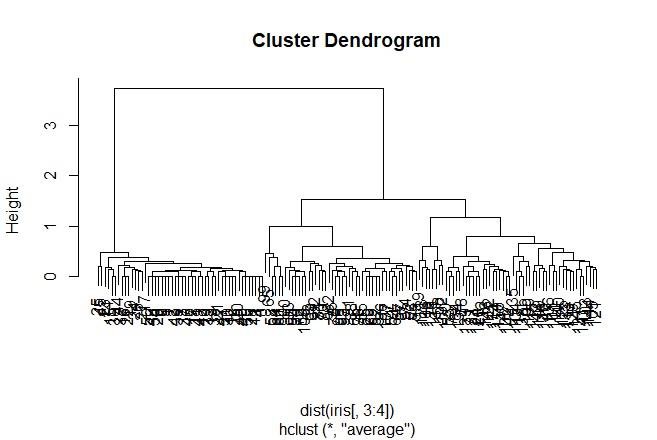
3 0 5 49

> ggplot(iris, aes(Petal.Length, Petal.Width, color = iris$Species)) + geom\_point(alpha = 0.4,

size = 3.5) + geom\_point(col = clusterCut) + scale\_color\_manual(values = c('black', 'red', 'gre

en'))

129



MCA Sem I: ADBMS Lab DES’s NMITD Roll No: C23071

130