ASSIGNMENT III

Problem 1:

In an organization, number of departments exists. Each department has a name & unique code. Number of employees work in each department. Each employee has unique employee code. Detailed information like name, address, city, basic, date of join are also stored. In a leave register for each employee leave records are kept showing leave type (CL/EL/ML etc.), fromdate and to-date. When an employee retires or resigns then all the leave information pertaining to him are also deleted. Basic salary must be within Rs.5000 to Rs.9000. A department cannot be deleted if any employee record refers to it. Valid grades are A/B/C. Employee name must be in uppercase only. Default value for joining date is system date. Design & implement the tables with necessary constraints to support the scenario depicted above.

Solution:

```
SQL> CREATE TABLE DEPARTMENT(
2  DEPT_CODE char(10) PRIMARY KEY,
3  DEPT_NAME char(30),
4  NO_OF_EMPLOYEE INTEGER);

Table created.

SQL> CREATE TABLE EMPLOYEE(
2    EMP_CODE char(10) PRIMARY KEY,
3    DEPT_CODE char(10),
4    EMP_NAME char(30),
5    ADDRESS char(50),
6    CITY char(20),
7    BASIC number(5),
8    JOIN_DATE DATE DEFAULT SYSDATE,
9    GRADE char(1),
```

```
10  CONSTRAINT UPPER_CASE_NAME CHECK(EMP_NAME =
    UPPER(EMP_NAME)),

11  CONSTRAINT GRADE_VALID CHECK(GRADE IN ('A', 'B', 'C')),

12  CONSTRAINT SALARY_RANGE CHECK(BASIC BETWEEN 5000 AND
9000),

13  CONSTRAINT DEPT_FOREIGN_KEY FOREIGN KEY(DEPT_CODE)
REFERENCES DEPARTMENT(DEPT_CODE));


Table created.


SQL> CREATE TABLE LEAVE(

2  RECORD_ID char(5) PRIMARY KEY,

3  EMP_CODE char(10),

4  TYPE char(2),

5  FROM_DATE DATE,

6  TO_DATE DATE,

7  CONSTRAINT VALID_TYPE CHECK(TYPE IN ('CL', 'EL', 'ML')),

8  FOREIGN KEY(EMP_CODE) REFERENCES EMPLOYEE(EMP_CODE));


Table created.
```

- ***Problem 2:***

Try to violate the constraints that you have implemented in the table & note, what happens. [Try with suitable INSERT/UPDATE/DELETE instruction]

Solution:

```
SQL> INSERT INTO DEPARTMENT VALUES('D1', 'ACCOUNTS', 5);

1 row created.


SQL> INSERT INTO DEPARTMENT VALUES('D2', 'MARKETING', 12);
```

1 row created.

SQL> INSERT INTO DEPARTMENT VALUES('D3', 'DEVELOPMENT', 10);

1 row created.

SQL> INSERT INTO DEPARTMENT VALUES('D4', 'SALES', 20);

1 row created.

SQL> INSERT INTO DEPARTMENT VALUES('D5', 'PURCHASE', 18);

1 row created.

SQL> INSERT INTO EMPLOYEE VALUES('E1', 'D2', 'Dhruv', '20B Old P.O. Road','Kolkata',5500,'06-JAN-2019', 'B');

INSERT INTO EMPLOYEE VALUES('E1', 'D2', 'Dhruv', '20B Old P.O. Road','Kolkata',5500,'06-JAN-2019', 'B')

*

ERROR at line 1:

ORA-02290: check constraint (SYSTEM.UPPER_CASE_NAME) violated

NOTE: Here I have intentionally violated my mentioned constraint that employee names must be in uppercase letters. I note that I get an error saying “check constraint violated”.

SQL> INSERT INTO EMPLOYEE VALUES('E1', 'D2', 'DHRUV', '20B Old P.O. Road','Kolkata',5500,'06-JAN-2019', 'B');

1 row created.

SQL> INSERT INTO EMPLOYEE VALUES('E2', 'D1', 'KAVYA', '35 Pali Hill','Mumbai',6000,'02-JUNE-2020', 'A');

1 row created.

SQL> INSERT INTO EMPLOYEE VALUES('E3', 'D2', 'PRIYA', '14A Gopal Banerjee Lane','Kolkata',7000,'06-JAN-2020', 'C');

1 row created.

SQL> INSERT INTO EMPLOYEE VALUES('E4', 'D5', 'DIVYA', 'Lajpat Nagar','Delhi',6500,'10-AUG-2019', 'B');

1 row created.

SQL> INSERT INTO EMPLOYEE VALUES('E5', 'D3', 'JIGYASA', 'New Palace Lane','Pune',8000,'01-MAR-2019', 'A');

1 row created.

SQL> INSERT INTO EMPLOYEE VALUES('E6', 'D3', 'SOURAV', '51A Shyama Prosad Road','Kolkata',5000,'04-MAR-2020', 'C');

1 row created.

SQL> INSERT INTO EMPLOYEE VALUES('E7', 'D4', 'SHREYA', 'Mahatma Gandhi Sarani','Gujarat',8000,'10-OCT-2020', 'A');

1 row created.

SQL> INSERT INTO EMPLOYEE VALUES('E7', 'D4', 'ANWESHA', 'Hill Road','Darjeeling',2000,'03-FEB-2019', 'B');

INSERT INTO EMPLOYEE VALUES('E7', 'D4', 'ANWESHA', 'Hill Road','Darjeeling',2000,'03-FEB-2019', 'B')

*

ERROR at line 1:

ORA-02290: check constraint (SYSTEM.SALARY_RANGE) violated

NOTE: Here I have intentionally violated my mentioned constraint that BASIC must be between 5000-9000. I note that I get an error saying "Check constraint violated".

SQL> INSERT INTO EMPLOYEE VALUES('E8', 'D4', 'ANWESHA', 'Hill Road','Darjeeling',5500,'03-FEB-2019', 'B');

1 row created.

SQL> INSERT INTO LEAVE VALUES('L1', 'E4','EL', '02-MAR-2020','01-JUN2020');

1 row created.

SQL> INSERT INTO LEAVE VALUES('L2', 'E2','ML', '02-JUL-2020','10-SEP2020');

1 row created.

SQL> INSERT INTO LEAVE VALUES('L3', 'E5','ML', '15-NOV-2019','10-DEC2019');

1 row created.

SQL> INSERT INTO LEAVE VALUES('L4', 'E7','CL', '01-APR-2020','20-APR2020');

1 row created.

SQL> SELECT * FROM EMPLOYEE;

EMP_CODE	DEPT_CODE	EMP_NAME	ADDRESS	CITY	BASIC	JOIN_DATE	GRADE
E1	D2	DHRUV	20B Old P.O. Road	Kolkata	5500	06-JAN-19	B
E2	D1	KAVYA	35 Pali Hill	Mumbai	6000	02-JUN-20	A
E3	D2	PRIYA	14A Gopal Banerjee Lane	Kolkata	7000	06-JAN-20	C
E4	D5	DIVYA	Lajpat Nagar	Delhi	6500	10-AUG-19	B
E5	D3	JIGYASA	New Palace Lane	Pune	8000	01-MAR-19	A
E6	D3	SOURAV	51A Shyama Prosad Road	Kolkata	5000	04-MAR-20	C
E7	D4	SHREYA	Mahatma Gandhi Sarani	Gujarat	8000	10-OCT-20	A
E8	D4	ANWESHA	Hill Road	Darjeeling	5500	03-FEB-19	B

8 rows selected.

Problem 3:

3. a) Create a view showing employee code, name, dcode & Basic For a particular department.

b) Try to ensure a row into the view with valid department & also with invalid ones.

c) Find the newly inserted row in the table From which view was created.

d) Try to increment basic by Rs.100/-

e) Check it in the original table.

f) Delete the view.

Solution:

(a) SQL> CREATE VIEW EMPLOYEE_DETAILS AS

2 SELECT EMP_CODE, EMP_NAME, DEPT_CODE, BASIC

3 FROM EMPLOYEE 4 WHERE DEPT_CODE =

'D3';

View created.

SQL> SELECT * FROM EMPLOYEE_DETAILS;

EMP_CODE	EMP_NAME	DEPT_CODE	BASIC
E5	JIGYASA	D3	8000
E6	SOURAV	D3	5000

(b) SQL> INSERT INTO EMPLOYEE_DETAILS VALUES('E9', 'ANANYA', 'D7', 6700);

INSERT INTO EMPLOYEE_DETAILS VALUES('E9', 'ANANYA', 'D7', 6700)

*

ERROR at line 1:

ORA-02291: integrity constraint (SYSTEM.DEPT_FOREIGN_KEY) violated - parent key not found

NOTE: Here I have intentionally tried to insert a row in the view with an invalid DEPT_CODE. I note that I have an error saying integrity constraint violated. This happens because there is no record in the Department table corresponding to the DEPT_CODE 'D7'.

SQL> INSERT INTO EMPLOYEE_DETAILS VALUES('E9', 'ANANYA', 'D4', 6700);

1 row created.

(c) SQL> SELECT * FROM EMPLOYEE

2 WHERE EMP_CODE = 'E9';

EMP_CODE DEPT_CODE EMP_NAME ADDRESS CITY BASIC JOIN_DATE GRADE

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

E9 D4 ANANYA

6700

(d) SQL> UPDATE EMPLOYEE_DETAILS

2 SET BASIC = BASIC + 100;

2 rows updated.

(e) SQL> SELECT * FROM EMPLOYEE_DETAILS;

EMP_CODE EMP_NAME DEPT_CODE BASIC

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

E5 JIGYASA D3 8100

E6 SOURAV D3 5100

SQL> SELECT * FROM EMPLOYEE;

EMP_CODE	DEPT_CODE	EMP_NAME	ADDRESS	CITY	BASIC	JOIN_DATE	GRADE
E1	D2	DHRUV	20B Old P.O. Road	Kolkata	5500	06-JAN-19	B
E2	D1	KAVYA	35 Pali Hill	Mumbai	6000	02-JUN-20	A
E3	D2	PRIYA	14A Gopal Banerjee Lane	Kolkata	7000	06-JAN-20	C
E4	D5	DIVYA	Lajpat Nagar	Delhi	6500	10-AUG-19	B
E5	D3	JIGYASA	New Palace Lane	Pune	8100	01-MAR-19	A
E6	D3	SOURAV	51A Shyama Prosad Road	Kolkata	5100	04-MAR-20	C
E7	D4	SHREYA	Mahatma Gandhi Sarani	Gujarat	8000	10-OCT-20	A
E8	D4	ANWESHA	Hill Road	Darjeeling	5500	03-FEB-19	B

(f) SQL> DROP VIEW EMPLOYEE_DETAILS;

View dropped.

 Problem 4:

(a) Create a view showing Emp_Code, name, Dept_Name, basic, leave type, From date & to date.

(b) Try to insert a row in the view. Check what happens?

(c) Try to increment basic by Rs.100.

(d) Delete the view.

Solution:

(a) SQL> CREATE VIEW EMPLOYEE_LEAVE_DETAILS AS

2 SELECT e.EMP_CODE, e.EMP_NAME, e. BASIC, e.DEPT_CODE, d.DEPT_NAME, l.TYPE, l.FROM_DATE, l.TO_DATE

3 FROM EMPLOYEE e, DEPARTMENT d, LEAVE l

4 WHERE e.DEPT_CODE = d.DEPT_CODE AND

5 e.EMP_CODE = l.EMP_CODE; View created.

(b) SQL> INSERT INTO EMPLOYEE_LEAVE_DETAILS VALUES

2 ('E10', 'PRAKASH', 5600,'D3', 'DEVELOPMENT', 'CL','02-JAN-2022', '20JAN-2022');

INSERT INTO EMPLOYEE_LEAVE_DETAILS VALUES

*

ERROR at line 1:

ORA-01779: cannot modify a column which maps to a non key-preserved table

NOTE: Here I have tried to insert a row in the view EMPLOYEE_LEAVE_DETAILS and I have faced an error. This action is not possible as I am trying to insert some values in three tables simultaneously of which two table's Primary Key is not mentioned. ***(c)***

SQL> UPDATE EMPLOYEE_LEAVE_DETAILS

2 SET BASIC = BASIC + 100;

SET BASIC = BASIC + 100

*

ERROR at line 2:

ORA-01779: cannot modify a column which maps to a non key-preserved table

NOTE: Here I have tried to update the basic of each row in the view, the action is not possible.

(d) SQL> DROP VIEW EMPLOYEE_LEAVE_DETAILS;

View dropped.

Problem 5:

(a) Create a table having Emp_code , Name, Dept_name, & basic From the existing tables along with the records of the employee who are in a particular department (say, d1) and with a basic Rs. 7000/-

(b) From the existing table, add the employees with the basic salary greater than or equal to 7000/-

(c) Alter the table to add a net pay column.

(d) Replace net pay with 1.5* Basic.

(e) Try to remove the net net pay column. [It may require no. of steps]

Solution:

(a) SQL> CREATE TABLE PERSONNEL(

2 EMP_CODE PRIMARY KEY, EMP_NAME, DEPT_NAME, BASIC)

3 AS SELECT EMP_CODE, EMP_NAME, DEPT_NAME, BASIC FROM EMPLOYEE, DEPARTMENT

4 WHERE EMPLOYEE.DEPT_CODE = DEPARTMENT.DEPT_CODE

5 AND BASIC = 7000 AND DEPARTMENT.DEPT_CODE = 'D3';

Table created.

(b) SQL> INSERT INTO PERSONNEL

2 SELECT EMP_CODE, EMP_NAME, DEPT_NAME, BASIC FROM EMPLOYEE, DEPARTMENT

3 WHERE EMPLOYEE.DEPT_CODE = DEPARTMENT.DEPT_CODE

4 AND BASIC >= 7000

```
5  AND EMP_CODE NOT IN
6  (SELECT DISTINCT EMP_CODE FROM PERSONNEL); 3 rows
   created.
```

(c) SQL> ALTER TABLE PERSONNEL ADD (NET_PAY NUMBER(9,2));
Table altered.

(d) SQL> UPDATE PERSONNEL SET NET_PAY = 1.5 * BASIC; 3 rows updated.

(e) SQL> ALTER TABLE PERSONNEL
2 DROP COLUMN NET_PAY;
Table altered.

- ***Problem 6:***

Drop all the tables that you have created.

- ***Solution:***

SQL> DROP TABLE PERSONNEL;

Table dropped.

SQL> DROP TABLE LEAVE;

Table dropped.

SQL> DROP TABLE EMPLOYEE;

Table dropped.

SQL> DROP TABLE DEPARTMENT;

Table dropped.

ASSIGNMENT IV

Problem 1:

(a) Create EMP table with ECODE (Primary key), ENAME, DCODE, GRADE, BASIC & JN-DT as the columns. [Except BASIC & JN-DT, all columns are of char type and site of Grade is 1.]

(b) Insert number of rows.

- ***Solution:***

```
SQL> CREATE TABLE EMP(
2  ECODE char(10) PRIMARY KEY,
3  ENAME char(30),
4  DCODE char(10),
5  GRADE char(1),
6  BASIC number,
7  JN_DT DATE);

Table created.

SQL> INSERT INTO EMP VALUES('E1', 'Priya', 'D2', 'B', 5000, '02-JAN2020');

1 row created.

SQL> INSERT INTO EMP VALUES('E2', 'Manish', 'D1', 'C', 5500, '02-JAN2020');

1 row created.

SQL> INSERT INTO EMP VALUES('E3', 'Akash', 'D2', 'B', 5500, '06-FEB2020');

1 row created.

SQL> INSERT INTO EMP VALUES('E4', 'Rachna', 'D4', 'A', 7000, '06-APR2019');
```

1 row created.

SQL> INSERT INTO EMP VALUES('E5', 'Rachna', 'D3', 'B', 7500, '10AUG-2019');

1 row created.

SQL> INSERT INTO EMP VALUES('E6', 'Ritu', 'D4', 'C', 6000, '12-SEP2019');

1 row created.

SQL> INSERT INTO EMP VALUES('E7', 'Aritra', 'D3', 'A',8000, '06-JAN2021');

1 row created.

SQL> INSERT INTO EMP VALUES('E8', 'Piyush', 'D2', 'C', 7500, '17-OCT2021');

1 row created.

SQL> INSERT INTO EMP VALUES('E9', 'Navin', 'D1', 'B', 6800, '13-MAR2020');

1 row created.

SQL> INSERT INTO EMP VALUES('E10', 'Pritha', 'D4', 'A', 8500, '12-JUN2019');

1 row created.

Problem 2:

Change the column heading as shown below, So that in subsequent SELECT statement newly set heading will be shown:

ECODE EMPLOYEE CODE

ENAME NAME

DCODE DEPT.CODE

JN-DT JONING DATE

- ***Solution:***

```
SQL> COLUMN ECODE HEADING 'EMPLOYEE_CODE';
SQL> COLUMN ENAME HEADING 'NAME';
SQL> COLUMN DCODE HEADING 'DEPT_CODE';
SQL> COLUMN JN_DT HEADING 'JOINING DATE';
SQL> SELECT * FROM EMP;
```

EMPLOYEE_C	NAME	DEPT_CODE	G	BASIC	JOINING DATE
E1	Priya	D2	B	5000	02-JAN-20
E2	Manish	D1	C	5500	02-JAN-20
E3	Akash	D2	B	5500	06-FEB-20
E4	Rachna	D4	A	7000	06-APR-19
E5	Rachna	D3	B	7500	10-AUG-19
E6	Ritu	D4	C	6000	12-SEP-19
E7	Aritra	D3	A	8000	06-JAN-21
E8	Piyush	D2	C	7500	17-OCT-21
E9	Navin	D1	B	6800	13-MAR-20
E10	Pritha	D4	A	8500	12-JUN-19

10 rows selected.

- ***Problem 3:***

Set the format of columns as mentioned below, So that in subsequent SELECT statement, values appear in the specified format:

*format of BASIC is such that a value of 7000 will be shown as 7,000

*Format of GRADE will be such that full column name appears in the display.

*For JN-DT format is such that 01-JAN-00 will be shown as JANURY 01, 2000.

- ***Solution:***

```
SQL> COLUMN BASIC FORMAT '99,999';
```

SQL> COLUMN GRADE FORMAT A5;
SQL> ALTER SESSION SET NLS_DATE_FORMAT = 'MONTH DD,YYYY';

Session altered.

SQL> SELECT * FROM EMP;

EMPLOYEE_C	NAME	DEPT_CODE	GRADE	BASIC	JOINING DATE
E1	Priya	D2	B	5,000	JANUARY 02, 2020
E2	Manish	D1	C	5,500	JANUARY 02, 2020
E3	Akash	D2	B	5,500	FEBRUARY 06, 2020
E4	Rachna	D4	A	7,000	APRIL 06, 2019
E5	Rachna	D3	B	7,500	AUGUST 10, 2019
E6	Ritu	D4	C	6,000	SEPTEMBER 12, 2019
E7	Aritra	D3	A	8,000	JANUARY 06, 2021
E8	Piyush	D2	C	7,500	OCTOBER 17, 2021
E9	Navin	D1	B	6,800	MARCH 13, 2020
E10	Pritha	D4	A	8,500	JUNE 12, 2019

10 rows selected.

Problem 4:

(a) Show the display attributes of all the columns.
(b) Show the display attributes of particular column.
(c) Suppress the newly set attributes of JN-DT .Try a select statement.
(d) Reset the newly set attributes of JN-DT (e) Reset the newly set attributes of all columns.
(f) Shown the display attributes of all columns.

 Solution:

(a) SQL> COLUMN;

```
COLUMN   GRADE ON
FORMAT   A5

COLUMN   BASIC ON
FORMAT   99,999

COLUMN   JN_DT ON
HEADING  'JOINING DATE'

COLUMN   DCODE ON
HEADING  'DEPT_CODE'

COLUMN   ENAME ON
HEADING  'NAME'

COLUMN   ECODE ON
HEADING  'EMPLOYEE_CODE'

COLUMN   result_plus_xquery ON
HEADING  'Result Sequence'

COLUMN   other_plus_exp ON
FORMAT   a44

COLUMN   other_tag_plus_exp ON
FORMAT   a29

COLUMN   object_node_plus_exp ON
```

```
FORMAT   a8

COLUMN   plan_plus_exp ON
FORMAT   a60

COLUMN   parent_id_plus_exp ON
HEADING  'p'
FORMAT   990

COLUMN   id_plus_exp ON
HEADING  'i'
FORMAT   990

COLUMN   droptime_plus_show_recyc ON
HEADING  'DROP TIME'
FORMAT   a19

COLUMN   objtype_plus_show_recyc ON
HEADING  'OBJECT TYPE'
FORMAT   a12

COLUMN   objectname_plus_show_recyc ON
HEADING  'RECYCLEBIN NAME'
```

```
FORMAT   a30

COLUMN   origname_plus_show_recyc ON
HEADING  'ORIGINAL NAME'
FORMAT   a16

COLUMN   value_col_plus_show_param ON
HEADING  'VALUE'
FORMAT   a30

COLUMN   name_col_plus_show_param ON
HEADING  'NAME'
FORMAT   a36

COLUMN   units_col_plus_show_sga ON
FORMAT   a15

COLUMN   name_col_plus_show_sga ON
FORMAT   a24

COLUMN   ERROR ON
FORMAT   A65 word_wrap

COLUMN   LINE/COL ON
FORMAT   A8

COLUMN   ROWLABEL ON
FORMAT   A15
```

(b)

```
SQL> COLUMN JN_DT;
```

```
COLUMN   JN_DT ON
HEADING  'JOINING DATE'
```

(c) SQL> COLUMN JN_DT OFF;

SQL> SELECT * FROM EMP;

EMPLOYEE_C	NAME	DEPT_CODE	GRADE	BASIC	JN_DT
E1	Priya	D2	B	5,000	JANUARY 02, 2020
E2	Manish	D1	C	5,500	JANUARY 02, 2020
E3	Akash	D2	B	5,500	FEBRUARY 06, 2020
E4	Rachna	D4	A	7,000	APRIL 06, 2019
E5	Rachna	D3	B	7,500	AUGUST 10, 2019
E6	Ritu	D4	C	6,000	SEPTEMBER 12, 2019
E7	Aritra	D3	A	8,000	JANUARY 06, 2021
E8	Piyush	D2	C	7,500	OCTOBER 17, 2021
E9	Navin	D1	B	6,800	MARCH 13, 2020
E10	Pritha	D4	A	8,500	JUNE 12, 2019

10 rows selected.

(d) SQL> COLUMN JN_DT CLEAR;

SQL> SELECT * FROM EMP;

EMPLOYEE_C	NAME	DEPT_CODE	GRADE	BASIC	JN_DT
E1	Priya	D2	B	5,000	JANUARY 02, 2020
E2	Manish	D1	C	5,500	JANUARY 02, 2020
E3	Akash	D2	B	5,500	FEBRUARY 06,2020
E4	Rachna	D4	A	7,000	APRIL 06, 2019
E5	Rachna	D3	B	7,500	AUGUST 10,2019

E6	Ritu	D4	C	6,000	SEPTEMBER 12,2019
E7	Aritra	D3	A	8,000	JANUARY 06,2021
E8	Piyush	D2	C	7,500	OCTOBER 17, 2021
E9	Navin	D1	B	6,800	MARCH 13, 2020
E10	Pritha	D4	A	8,500	JUNE 12,2019

10 rows selected.

(e) SQL> CLEAR COLUMN;

columns cleared

SQL> SELECT * FROM EMP;

ECODE	ENAME	DCODE	G	BASIC	JN_DT
E1	Priya	D2	B	5000	JANUARY 02, 2020
E2	Manish	D1	C	5500	JANUARY 02, 2020
E3	Akash	D2	B	5500	FEBRUARY 06,2020
E4	Rachna	D4	A	7000	APRIL 06,2019
E5	Rachna	D3	B	7500	AUGUST 10,2019
E6	Ritu	D4	C	6000	SEPTEMBER 12,2019
E7	Aritra	D3	A	8000	JANUARY 06 ,2021
E8	Piyush	D2	C	7500	OCTOBER 17,2021
E9	Navin	D1	B	6800	MARCH 13,2020
E10	Pritha	D4	A	8500	JUNE 12,2019

10 rows selected.

(f) SQL> COLUMN;

SP2-0045: * no COLUMN defined

Problem 5:

. (a) Show the records from EMP table in the ascending order of DCODE. DCODE value will be shown only for the first record of that department (same of DCODE is not repeated)

(b) Further take measures so that, after displaying the records of a department it skips one line.

(c)Further take measures so that records one also ordered on the basis of GRADE with in a department & same GRADE value is not repeated.

(d) Take measure so that at the end of each GRADE in a department it will show average Basic for that grade in that department. At the end of each department, it will show the average & total Basic for the department. At the end of all departments it will show the overall total basic & average basic.

Solution:

(a) SQL> BREAK ON DCODE;
SQL> SELECT * FROM EMP ORDER BY DCODE ASC;

```
ECODE      ENAME                DCODE      G      BASIC JN_DT
---------- ---------------------------- ---------- - ---------- ---------
E2         Manish                D1         C       5500 02-JAN ---------- 15
E9         Navin                            B       6800 13-MAR ---------- 23
E1         Priya                 D2         B       5000 02-JAN ---------- 23
E3         Akash                             B       5500 06-FEB ---------- 23
E8         Piyush                            C       7500 17-OCT ---------- 23
E5         Rachna                D3         B       7500 10-AUG-19 E7
Aritra                           A       8000 06-JAN ---------- 23

E10        Pritha                D4         A       8500 12-JUN-19
E4         Rachna                            A       7000 06-APR-19
E6         Ritu                              C       6000 12-SEP-19
10 rows selected.
```

(b) SQL> BREAK ON DCODE SKIP 1;

SQL> SELECT * FROM EMP ORDER BY DCODE ASC;

```
ECODE      ENAME   DCODE      G      BASIC                JN_DT
```

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

E2	Manish	D1	C	5500	02-JAN-20
E9	Navin		B	6800	13-MAR-20
E1	Priya	D2	B	5000	02-JAN-20
E3	Akash		B	5500	06-FEB-20
E8	Piyush		C	7500	17-OCT-21
E5	Rachna	D3	B	7500	10-AUG-19
E7	Aritra		A	8000	06-JAN-21
E10	Pritha	D4	A	8500	12-JUN-19
E4	Rachna		A	7000	06-APR-19
E6	Ritu		C	6000	12-SEP-19

10 rows selected.

(c)

SQL> BREAK ON DCODE SKIP 1 ON GRADE;

SQL> SELECT * FROM EMP ORDER BY DCODE ASC, GRADE ASC;

ECODE ENAME DCODE G BASIC JN_DT

---------- ---------- ---------- - ---------- --------- E9

Navin D1 B 6800 13-MAR-20

E2 Manish C 5500 02-JAN-20

E1 Priya D2 B 5000 02-JAN-20

E3 Akash 5500 06-FEB-20

E8 Piyush C 7500 17-OCT-21

E7 Aritra D3 A 8000 06-JAN-21

E5 Rachna B 7500 10-AUG-19

E10 Pritha D4 A 8500 12-JUN-19

E4 Rachna 7000 06-APR-19

E6 Ritu C 6000 12-SEP-19

10 rows selected.

(d) SQL> COLUMN GRADE FORMAT A20;

SQL> BREAK ON DCODE SKIP 1 ON GRADE SKIP 1;

SQL> COMPUTE AVG LABEL 'AVG BASIC ON GRADE' OF BASIC ON GRADE;

SQL> SELECT * FROM EMP ORDER BY DCODE ASC, GRADE ASC;

ECODE	ENAME	DCODE	GRADE	BASIC	JN_DT
E9	Navin	D1	B	6800	13-MAR-20
			**********	----------	
			AVG BASIC ON GRADE	6800	
E2	Manish	D1	C	5500	02-JAN-20
			**********	----------	
			AVG BASIC ON GRADE	5500	
E1	Priya	D2	B	5000	02-JAN-20
E3	Akash			5500	06-FEB-20
			**********	----------	
			AVG BASIC ON GRADE	5250	

E8	Piyush	D2	C	7500	17-OCT-21

********** ----------

AVG BASIC ON GRADE 7500

E7	Aritra	D3	A	8000	06-JAN-21

********** ----------

AVG BASIC ON GRADE 8000

E5	Rachna	D3	B	7500	10-AUG-19

********** ----------

AVG BASIC ON GRADE 7500

E10	Pritha	D4	A	8500	12-JUN-19
E4	Rachna			7000	06-APR-19

********** ----------

AVG BASIC ON GRADE 7750

E6	Ritu	D4	C	6000	12-SEP-19

********** ----------

AVG BASIC ON GRADE 6000

10 rows selected.

Overall Average and Overall total BASIC at the end of the table EMP:

SQL> BREAK ON REPORT SKIP 3 ON DCODE SKIP 3 ON GRADE SKIP 1;

```
SQL> COMPUTE AVG LABEL 'AVG BASIC ON GRADE' OF BASIC ON
GRADE;

SQL> COMPUTE AVG LABEL "AVG BASIC ON DEPT" SUM LABEL
"TOTAL BASIC ON DEPT" OF BASIC ON DCODE;

SQL> COMPUTE AVG LABEL "AVG BASIC" SUM LABEL "TOTAL BASIC"
OF BASIC ON REPORT;

SQL> SELECT * FROM EMP ORDER BY DCODE ASC, GRADE ASC;

 ECODE      ENAME        DCODE      GRADE BASIC        JN_DT
---------- ---------- ---------- ---------- ---------- ---------

E9         Navin                 D1         B          6800         13-MAR-20
                                             ********** ----------
                                 AVG BASIC ON GRADE     6800

E2         Manish                D1         C          5500         02-JAN-20
                                             ********** ----------
                                 AVG BASIC ON GRADE     5500
                                             ********** ----------
                                 AVG BASIC              6150
                                 TOTAL BASIC            12300

E1         Priya                 D2         B          5000         02-JAN-20
E3         Akash                                       5500         06-FEB-20
                                 ********** ----------
                                 AVG BASIC ON GRADE     5250

E8         Piyush                D2         C          7500         17-OCT-
********** ---------- AVG BASIC ON GRADE                7500
                                             ********** ----------
                                 AVG BASIC              6000
                                 TOTAL BASIC            18000
```

E7	Aritra	D3	A	8000	06-JAN-21
		**********		----------	
AVG BASIC ON GRADE	8000				
E5	Rachna		B	7500	10-AUG-19
		**********			
AVG BASIC ON GRADE	7500				
		*********		----------	
		AVG BASIC		7750	
		TOTAL BASI		15500	
E10	Pritha	D4	A	8500	12-JUN-19
E4	Rachna			7000	06-APR-19
		**********		----------	
		AVG BASIC ON GRADE		7750	
E6	Ritu	D4	C	6000	12-SEP-19
		**********		----------	
		AVG BASIC ON GRADE		6000	
		*********		----------	
		AVG BASIC		7166.66667	
		TOTAL BASIC		21500	

10 rows selected.