

ASSESSMENT- 3

NAME: ALOK SINHA

REG NO.: 17BCE2380

LAB: L21+L22

Create the following tables and insert the values emp_department
DPT_CODE DPT_NAME DPT_ALLOTMENT

57	IT	65000
63	Finance	15000
47	HR	240000
27	RD	55000
89	QC	75000

emp_details

EMP_IDNO EMP_FNAME EMP_LNAME EMP_DEPT

127323	Michale Robbin	57
526689	Carlos Snares	63
843795	Enric Dosio	57
328717	Jhon Snares	63
444527	Joseph Dosni	47
659831	Zanifer Emily	47
847674	Kuleswar Sitaraman	57
748681	Henrey Gabriel	47
555935	Alex Manuel	57
539569	George Mardy	27
733843	Mario Saule	63
631548	Alan Snappy	27
839139	Maria Foster	57

SQL*Plus: Release 11.2.0.1.0 Production on Thu Sep 06 09:46:52 2018

Copyright (c) 1982, 2010, Oracle. All rights reserved.

Enter user-name: 17BCE2380@VITORA

Enter password:

Connected to:

Oracle Database 11g Enterprise Edition Release 11.1.0.7.0 - Production

With the Partitioning, OLAP, Data Mining and Real Application Testing options

JOINS

```
SQL> CREATE TABLE EMP_DEPARTMENT (DPT_CODE NUMBER, DPT_NAME VARCHAR (20),  
DPT_ALLOTMENT NUMBER);
```

Table created.

```
SQL> SELECT * FROM EMP_DEPARTMENT;
```

DPT_CODE	DPT_NAME	DPT_ALLOTMENT
57	IT	65000
63	FINANCE	15000
47	HR	240000
27	RD	55000

89 QC

75000

```
SQL> CREATE TABLE EMP_DETAILS (EMP_IDNO NUMBER, EMP_FNAME VARCHAR (10), EMP_LNAME  
VARCHAR (10), EMP_DEPT NUMBER);
```

Table created.

```
SQL> DESC EMP_DETAILS;
```

Name	Null?	Type

EMP_IDNO		NUMBER
EMP_FNAME		VARCHAR2(10)
EMP_LNAME		VARCHAR2(10)
EMP_DEPT		NUMBER

```
SQL> SELECT * FROM EMP_DETAILS;
```

EMP_IDNO	EMP_FNAME	EMP_LNAME	EMP_DEPT

127323	MICHALE	ROBBIN	57
526689	CARLOS	SNARES	63
843795	ENRIC	DOSIO	57
328717	JHON	SNARES	63
444527	JOSEPH	DOSNI	47
659831	ZANIFER	EMILY	47
847674	KULESWAR	SITARAMAN	57
748681	HENREY	GABRIEL	47
555935	ALEX	MANUEL	57
539569	GEORGE	MARDY	27

733843 MARIO SAULE 63

EMP_IDNO EMP_FNAME EMP_LNAME EMP_DEPT

631548 ALAN SNAPPY 27

839139 MARIA FOSTER 57

13 rows selected.

1. Write a query in SQL to find the names of departments where more than two employees are working.

```
SQL> SELECT emp_department.dpt_name
2 FROM emp_details
3 INNER JOIN emp_department
4 ON emp_dept =dpt_code
5 GROUP BY emp_department.dpt_name
6 HAVING COUNT(*) > 2;
```

DPT_NAME

IT

HR

FINANCE

2. Write a query in SQL to find the first name and last name of employees working for departments with a budget more than Rs. 50000.

```
SQL> SELECT emp_fname, emp_lname
2  FROM emp_details
3  WHERE emp_dept IN
4  (SELECT dpt_code
5    FROM emp_department
6    WHERE dpt_allotment > 50000);
```

```
EMP_FNAME EMP_LNAME
```

```
-----
```

```
MARIA  FOSTER
```

```
ALEX   MANUEL
```

```
KULESWAR SITARAMAN
```

```
ENRIC   DOSIO
```

```
MICHALE ROBBIN
```

```
HENREY  GABRIEL
```

```
ZANIFER EMILY
```

```
JOSEPH  DOSNI
```

```
ALAN    SNAPPY
```

```
GEORGE  MARDY
```

```
10 rows selected.
```

3. Write a query in SQL to display the first name and last name of each employee, along with the department name and sanction amount for their department+

```
.
```

```
SELECT emp_details.emp_fname AS "First Name", emp_lname AS "Last Name",
       emp_department.dpt_name AS "Department",
       dpt_allotment AS "Amount Allotted"
```

```

FROM emp_details
INNER JOIN emp_department
ON emp_details.emp_dept = emp_department.dpt_code;

```

First Name	Last Name	Department	Amount Allotted
------------	-----------	------------	-----------------

MICHALE	ROBBIN	IT	65000
CARLOS	SNARES	FINANCE	15000
ENRIC	DOSIO	IT	65000
JHON	SNARES	FINANCE	15000
JOSEPH	DOSANI	HR	240000
ZANIFER	EMILY	HR	240000
KULESWAR	SITARAMAN	IT	65000
HENREY	GABRIEL	HR	240000
ALEX	MANUEL	IT	65000
GEORGE	MARDY	RD	55000
MARIO	SAULE	FINANCE	15000

First Name	Last Name	Department	Amount Allotted
------------	-----------	------------	-----------------

ALAN	SNAPPY	RD	55000
MARIA	FOSTER	IT	65000

13 rows selected.

4. Write a query in SQL to display all the data of employees including their department.

```

SQL> SELECT emp_idno, A.emp_fname AS "First Name", emp_lname AS "Last Name",
2    B.dpt_name AS "Department", emp_dept, dpt_code, dpt_allotment
3    FROM emp_details A
4    INNER JOIN emp_department B
5    ON A.emp_dept = B.dpt_code;

```

```

EMP_IDNO First Name    Last Name    Department  EMP_DEPT  DPT_CODE
-----

```

DPT_ALLOTMENT

```

-----
127323 MICHAEL    ROBBIN    IT        57    57
65000

526689 CARLOS    SNARES    FINANCE    63    63
15000

843795 ENRIC     DOSIO     IT        57    57
65000

```

```

EMP_IDNO First Name    Last Name    Department  EMP_DEPT  DPT_CODE
-----

```

DPT_ALLOTMENT

```

-----
328717 JHON      SNARES    FINANCE    63    63
15000

444527 JOSEPH    DOSANI    HR        47    47
240000

```

659831	ZANIFER	EMILY	HR	47	47
--------	---------	-------	----	----	----

240000

EMP_IDNO	First Name	Last Name	Department	EMP_DEPT	DPT_CODE
----------	------------	-----------	------------	----------	----------

DPT_ALLOTMENT

847674	KULESWAR	SITARAMAN	IT	57	57
--------	----------	-----------	----	----	----

65000

748681	HENREY	GABRIEL	HR	47	47
--------	--------	---------	----	----	----

240000

555935	ALEX	MANUEL	IT	57	57
--------	------	--------	----	----	----

65000

EMP_IDNO	First Name	Last Name	Department	EMP_DEPT	DPT_CODE
----------	------------	-----------	------------	----------	----------

DPT_ALLOTMENT

539569	GEORGE	MARDY	RD	27	27
--------	--------	-------	----	----	----

55000

733843	MARIO	SAULE	FINANCE	63	63
--------	-------	-------	---------	----	----

15000

631548	ALAN	SNAPPY	RD	27	27
--------	------	--------	----	----	----

55000

EMP_IDNO	First Name	Last Name	Department	EMP_DEPT	DPT_CODE
----------	------------	-----------	------------	----------	----------

DPT_ALLOTMENT

839139 MARIA	FOSTER	IT	57	57
65000				

13 rows selected.

SUB QUERY PROBLEMS

1. Creation of the tables

```
SQL> CREATE TABLE booking (  
  2 id DECIMAL NOT NULL,  
  3 booking_number DECIMAL NOT NULL,  
  4 version DECIMAL NOT NULL,  
  5 state CHAR (10) NOT NULL,  
  6 enter_ts TIMESTAMP NOT NULL,  
  7 enter_by CHAR(20) NOT NULL,  
  8 CONSTRAINT booking_pk PRIMARY KEY (id),  
  9 CONSTRAINT booking_unique UNIQUE (booking_number, version)  
 10);
```

SQL> SELECT * FROM BOOKING;

ID	BOOKING_NUMBER	VERSION	STATE
----	----------------	---------	-------

ENTER_TS

ENTER_BY

1	4711	1	CREATED
---	------	---	---------

02-FEB-14 10.01.01.000000 AM

EMILY

2 4711 2 modified
03-FEB-14 11.10.01.000000 AM
EMILY

ID	BOOKING_NUMBER	VERSION	STATE

ENTER_TS			

ENTER_BY			

3 4711 3 cancelled
10-FEB-14 09.01.01.000000 AM
John

4 4712 1 created
10-MAR-14 12.12.12.000000 PM

ID	BOOKING_NUMBER	VERSION	STATE

ENTER_TS			

ENTER_BY			

Emily

5 4712 2 delivered
12-MAR-14 06.01.00.000000 AM
Charles

6 4713 1 created

ID BOOKING_NUMBER VERSION STATE

ENTER_TS

ENTER_BY

11-MAR-14 08.50.02.000000 AM

Emily

7 4713 2 cancelled

12-MAR-14 08.40.12.000000 AM

Emily

ID BOOKING_NUMBER VERSION STATE

ENTER_TS

ENTER_BY

8 4713 3 reopend

13-MAR-14 10.04.32.000000 AM

Jack

9 4713 4 delivered

15-MAR-14 06.40.12.000000 AM

Jack

ID BOOKING_NUMBER VERSION STATE

ENTER_TS

ENTER_BY

9 rows selected.

2) ADDING VALUES TO THE TABLE

```
INSERT INTO booking VALUES (1, 4711, 1, 'created', TIMESTAMP'2014-02-02
10:01:01', 'Emily');
INSERT INTO booking VALUES (2, 4711, 2, 'modified', TIMESTAMP'2014-02-03
11:10:01',
'Emily');
INSERT INTO booking VALUES (3, 4711, 3, 'cancelled', TIMESTAMP'2014-02-10
09:01:01', 'John');
INSERT INTO booking VALUES (4, 4712, 1, 'created', TIMESTAMP'2014-03-10
12:12:12', 'Emily');
INSERT INTO booking VALUES (5, 4712, 2, 'delivered', TIMESTAMP'2014-03-12
06:01:00',
'Charles');
INSERT INTO booking VALUES (6, 4713, 1, 'created', TIMESTAMP'2014-03-11
08:50:02', 'Emily');
INSERT INTO booking VALUES (7, 4713, 2, 'cancelled', TIMESTAMP'2014-03-12
08:40:12',
'Emily');
INSERT INTO booking VALUES (8, 4713, 3, 'reopened', TIMESTAMP'2014-03-13
10:04:32', 'Jack');
INSERT INTO booking VALUES (9, 4713, 4, 'delivered', TIMESTAMP'2014-03-15 06:40:12', 'Jack');
```

3) PRACTISE THE FOLLOWING SUB QUERY

```
SQL> SELECT * FROM booking b
```

```
2 WHERE (booking_number, version) IN
```

```
3 (SELECT booking_number, MAX(version) FROM booking sq GROUP BY
```

4 booking_number)

5 ORDER BY booking_number;

ID BOOKING_NUMBER VERSION STATE

ENTER_TS

ENTER_BY

3 4711 3 cancelled
10-FEB-14 09.01.01.000000 AM
John

5 4712 2 delivered
12-MAR-14 06.01.00.000000 AM
Charles

ID BOOKING_NUMBER VERSION STATE

ENTER_TS

ENTER_BY

9 4713 4 delivered
15-MAR-14 06.40.12.000000 AM
Jack

```
SQL> SELECT * FROM booking b
2 WHERE (booking_number, version) IN
3 (SELECT booking_number, MAX(version) FROM booking sq GROUP BY
4 booking_number)
5 ORDER BY booking_number;
```

ID	BOOKING_NUMBER	VERSION	STATE
----	----------------	---------	-------

ENTER_TS

ENTER_BY

3	4711		3 cancelled
10-FEB-14 09.01.01.000000 AM			
John			

5	4712		2 delivered
12-MAR-14 06.01.00.000000 AM			
Charles			

ID	BOOKING_NUMBER	VERSION	STATE
----	----------------	---------	-------

ENTER_TS

ENTER_BY

9	4713		4 delivered
15-MAR-14 06.40.12.000000 AM			
Jack			

b)SQL> SELECT * FROM booking b

2 WHERE version != (SELECT max(version) FROM booking sq WHERE

3 sq.booking_number =

4 b.booking_number) AND booking_number = 4711 ORDER BY version;

ID	BOOKING_NUMBER	VERSION	STATE
----	----------------	---------	-------

ENTER_TS

ENTER_BY

1	4711		1 CREATED
---	------	--	-----------

02-FEB-14 10.01.01.000000 AM

EMILY

2	4711		2 modified
---	------	--	------------

03-FEB-14 11.10.01.000000 AM

EMILY

ID	BOOKING_NUMBER	VERSION	STATE
----	----------------	---------	-------

ENTER_TS

ENTER_BY

```
SQL> SELECT * FROM booking b
2 WHERE version != (SELECT max(version) FROM booking sq WHERE
3 sq.booking_number =
4 b.booking_number) AND booking_number = 4711 ORDER BY version;
```

ID	BOOKING_NUMBER	VERSION	STATE
----	----------------	---------	-------

ENTER_TS

ENTER_BY

1	4711	1	CREATED
02-FEB-14 10.01.01.000000 AM			
EMILY			

2	4711	2	modified
03-FEB-14 11.10.01.000000 AM			
EMILY			

ID	BOOKING_NUMBER	VERSION	STATE
----	----------------	---------	-------

ENTER_TS

ENTER_BY

4. DO THE FOLLOWING USING SUB QUERIES

A) FIND THE BOOKING WITH THE MOST VERSIONS.

SQL> SELECT *

2 FROM booking

3 WHERE version = (SELECT MAX (version) FROM booking);

ID	BOOKING_NUMBER	VERSION	STATE
----	----------------	---------	-------

ENTER_TS

ENTER_BY

3 4711 3 cancelled

10-FEB-14 09.01.01.000000 AM

John

8 4713 3 reopend

13-MAR-14 10.04.32.000000 AM

Jack

ID BOOKING_NUMBER VERSION STATE

ENTER_TS

ENTER_BY

**B) FIND ALL BOOKINGS WITH ARE CANCELLED (IN THE LATEST
VERSION).**


```

SQL> SELECT *
2 FROM booking b
3 WHERE version =
4 (SELECT MAX (version) FROM booking sq WHERE
sq.booking_number = b.booking_number)
5 AND state = 'cancelled';

```

ID	BOOKING_NUMBER	VERSION	STATE

ENTER_TS			

ENTER_BY			

3	4711	3	cancelled
10-FEB-14 09.01.01.000000 AM			
John			