

CREATING AND MANAGING TABLES

EX-NO :1

DATE:

1. Create the DEPT table based on the DEPARTMENT following the table instance chart below. Confirm that the table is created.

Column name	ID	NAME
Key Type		
Nulls/Unique		
FK table		
FK column		
Data Type	Number	Varchar
Length	7	25

QUERY:

```
CREATE TABLE DEPT(ID NUMBER(7) NOT NULL, NAME VARCHAR(25));
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, Gallery, and a search bar. The user is logged in as 'Afrin Fathima' (af). The main workspace is titled 'SQL Commands'. The schema is set to 'WKSP_AFRINFATHIMA014'. The command entered is 'CREATE TABLE DEPT(ID NUMBER(7) NOT NULL, NAME VARCHAR(25));'. The results section shows the message 'Table created.' and a execution time of '0.04 seconds'. The bottom footer includes copyright information for Oracle and the APEX version 'Oracle APEX 23.2.4'.

2. Create the EMP table based on the following instance chart. Confirm that the table is created.

Column name	ID	LAST_NAME	FIRST_NAME	DEPT_ID
Key Type				
Nulls/Unique				
FK table				
FK column				
Data Type	Number	Varchar	Varchar	Number
Length	7	25	25	7

QUERY:

```
CREATE TABLE EMP(ID NUMBER(7) NOT NULL,  
LAST_NAME VARCHAR(25) NOT NULL,  
FIRST_NAME VARCHAR(25), DEPT_ID NUMBER(7));
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes 'APEX', 'App Builder', 'SQL Workshop' (selected), 'Team Development', 'Gallery', a search bar, and user information for 'Afrin Fathima' (afrinfathima014). The main workspace has a toolbar with icons for undo, redo, search, and run. Below the toolbar, the 'Language' dropdown is set to 'SQL', and the 'Rows' dropdown is set to '10'. The SQL command area contains the following code:

```
1 CREATE TABLE EMP(ID NUMBER(7) NOT NULL, LAST_NAME VARCHAR(25) NOT NULL,  
2 FIRST_NAME VARCHAR(25), DEPT_ID NUMBER(7));
```

The results tab shows the output: 'Table created.' and '0.05 seconds'. The bottom footer displays copyright information for Oracle and the APEX version: 'Copyright © 1999, 2023, Oracle and/or Its affiliates.' and 'Oracle APEX 23.2.4'.

3. Modify the EMP table to allow for longer employee last names. Confirm the modification.
(Hint: Increase the size to 50)

QUERY:

```
ALTER TABLE EMP MODIFY(LAST_NAME VARCHAR(50));
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery. A search bar and user profile are also present. The main area is titled "SQL Commands". The language is set to SQL, and the number of rows to display is 10. The command entered is "ALTER TABLE EMP MODIFY(LAST_NAME VARCHAR(50));". Below the command, the results tab is selected, showing the output "Table altered." and a execution time of "0.06 seconds". The bottom footer displays copyright information for Oracle and the APEX version.

4. Create the EMPLOYEES2 table based on the structure of EMPLOYEES table. Include Only the Employee_id, First_name, Last_name, Salary and Dept_id coloumns. Name the columns Id, First_name, Last_name, salary and Dept_id respectively.

QUERY:

```
CREATE TABLE EMPLOYEES2(ID NUMBER(6) NOT NULL,  
FIRST_NAME VARCHAR(20), LAST_NAME VARCHAR(25) NOT NULL,  
SALARY NUMBER(8,2), DEPT_ID NUMBER(6) NOT NULL);
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery. A search bar and user profile are also present. The main area is titled "SQL Commands". The language is set to SQL, and the number of rows to display is 10. The command entered is "CREATE TABLE EMPLOYEES2(ID NUMBER(6) NOT NULL, FIRST_NAME VARCHAR(20), LAST_NAME VARCHAR(25) NOT NULL, SALARY NUMBER(8,2), DEPT_ID NUMBER(6) NOT NULL);". Below the command, the results tab is selected, showing the output "Table created." and a execution time of "0.05 seconds". The bottom footer displays copyright information for Oracle and the APEX version.

5. Drop the EMP table.

QUERY:

DROP TABLE EMP;

OUTPUT:

The screenshot shows the Oracle APEX interface with the SQL Workshop tab selected. In the SQL Commands section, the command `1 DROP TABLE EMP;` is entered. The Results tab shows the output: `Table dropped.`. The bottom status bar indicates the operation took `0.09 seconds`.

6. Rename the EMPLOYEES2 table as EMP.

QUERY:

RENAME EMPLOYEES2 TO EMP;

OUTPUT:

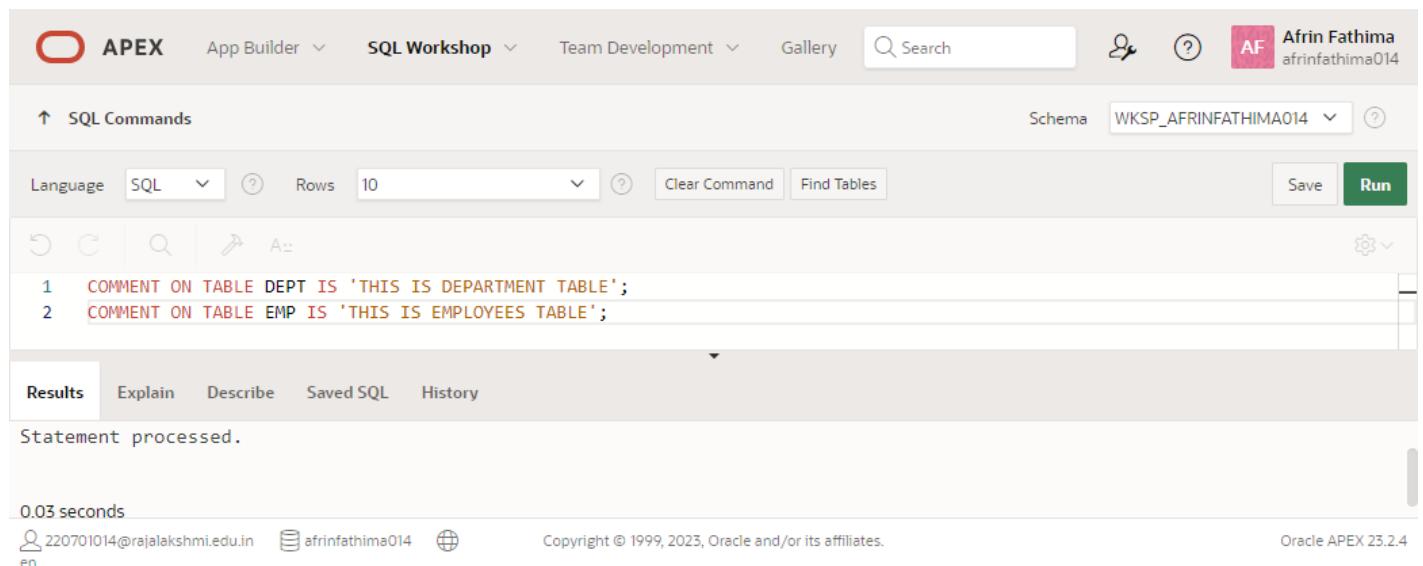
The screenshot shows the Oracle APEX interface with the SQL Workshop tab selected. In the SQL Commands section, the command `1 RENAME EMPLOYEES2 TO EMP;` is entered. The Results tab shows the output: `Statement processed.`. The bottom status bar indicates the operation took `0.05 seconds`.

7. Add a comment on DEPT and EMP tables. Confirm the modification by describing the table.

QUERY:

```
COMMENT ON TABLE DEPT IS 'THIS IS DEPARTMENT TABLE';
COMMENT ON TABLE EMP IS 'THIS IS EMPLOYEES TABLE';
```

OUTPUT:



The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery, along with a search bar and user information for 'Afrin Fathima'. The main area is titled 'SQL Commands' and contains a code editor with the following content:

```
1 COMMENT ON TABLE DEPT IS 'THIS IS DEPARTMENT TABLE';
2 COMMENT ON TABLE EMP IS 'THIS IS EMPLOYEES TABLE';
```

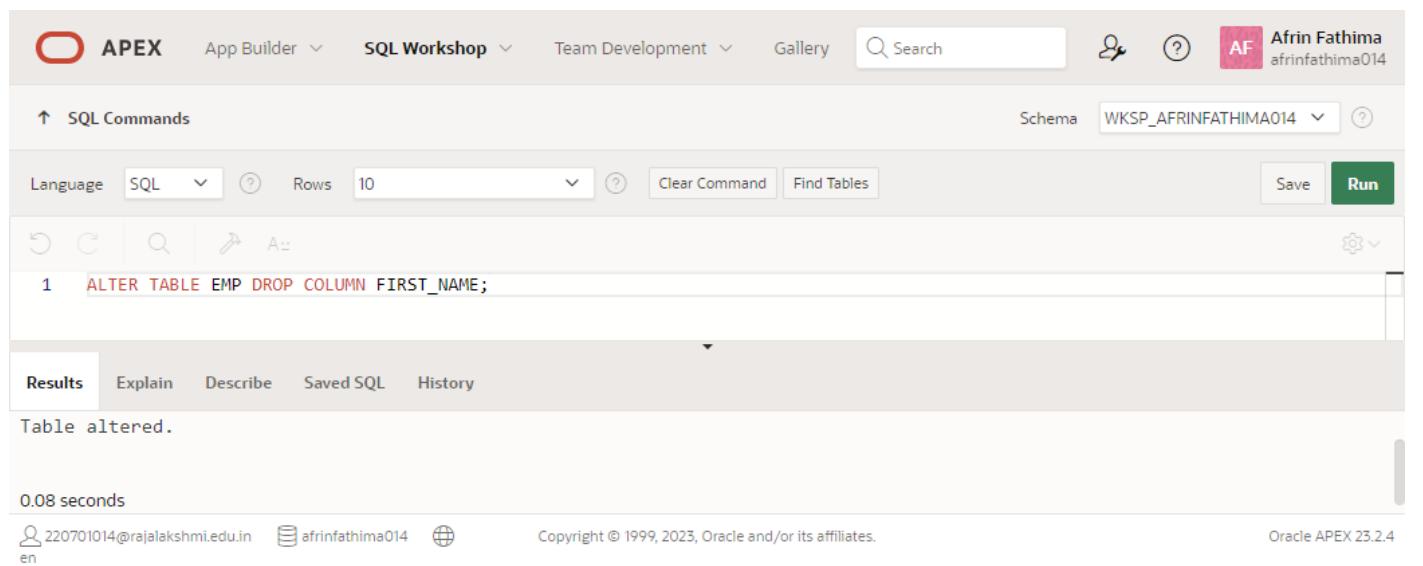
Below the code editor, there are tabs for 'Results', 'Explain', 'Describe', 'Saved SQL', and 'History'. The 'Results' tab is selected, displaying the message 'Statement processed.' The bottom of the screen shows execution time '0.03 seconds' and copyright information for Oracle APEX 23.2.4.

8. Drop the First_name column from the EMP table and confirm it.

QUERY:

```
ALTER TABLE EMP DROP COLUMN FIRST_NAME;
```

OUTPUT:



The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery, along with a search bar and user information for 'Afrin Fathima'. The main area is titled 'SQL Commands' and contains a code editor with the following content:

```
1 ALTER TABLE EMP DROP COLUMN FIRST_NAME;
```

Below the code editor, there are tabs for 'Results', 'Explain', 'Describe', 'Saved SQL', and 'History'. The 'Results' tab is selected, displaying the message 'Table altered.' The bottom of the screen shows execution time '0.08 seconds' and copyright information for Oracle APEX 23.2.4.

Evaluation Procedure	Marks awarded
Query(5)	
Execution (5)	
Viva(5)	
Total (15)	
Faculty Signature	

RESULT :

MANIPULATING DATA

EX-NO : 2

DATE:

1. Create MY_EMPLOYEE table with the following structure.

NAME	NULL?	TYPE
ID	Not null	Number(4)
Last_name		Varchar(25)
First_name		Varchar(25)
Userid		Varchar(25)
Salary		Number(9,2)

QUERY:

```
CREATE TABLE MY_EMPLOYEE(ID NUMBER(4) NOT NULL,  
LAST_NAME VARCHAR(25),FIRST_NAME VARCHAR(25),  
USERID VARCHAR(25), SALARY NUMBER(9,2));
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for App Builder, SQL Workshop, Team Development, and Gallery, along with a search bar and user profile information for 'Afrin Fathima'.

The main workspace is titled 'SQL Commands'. It features a toolbar with icons for Undo, Redo, Find, and Refresh, and dropdown menus for Language (set to SQL), Rows (set to 10), Clear Command, and Find Tables. On the right side of the toolbar are Save and Run buttons.

The SQL command area contains the following code:

```
1 CREATE TABLE MY_EMPLOYEE(ID NUMBER(4) NOT NULL,  
2 LAST_NAME VARCHAR(25),FIRST_NAME VARCHAR(25),  
3 USERID VARCHAR(25), SALARY NUMBER(9,2));
```

The results tab shows the output of the command: "Table created." Below the results, it indicates the execution time was 0.04 seconds. The bottom footer includes copyright information for Oracle and links for 220701014@rajalakshmi.edu.in and afrinfathima014.

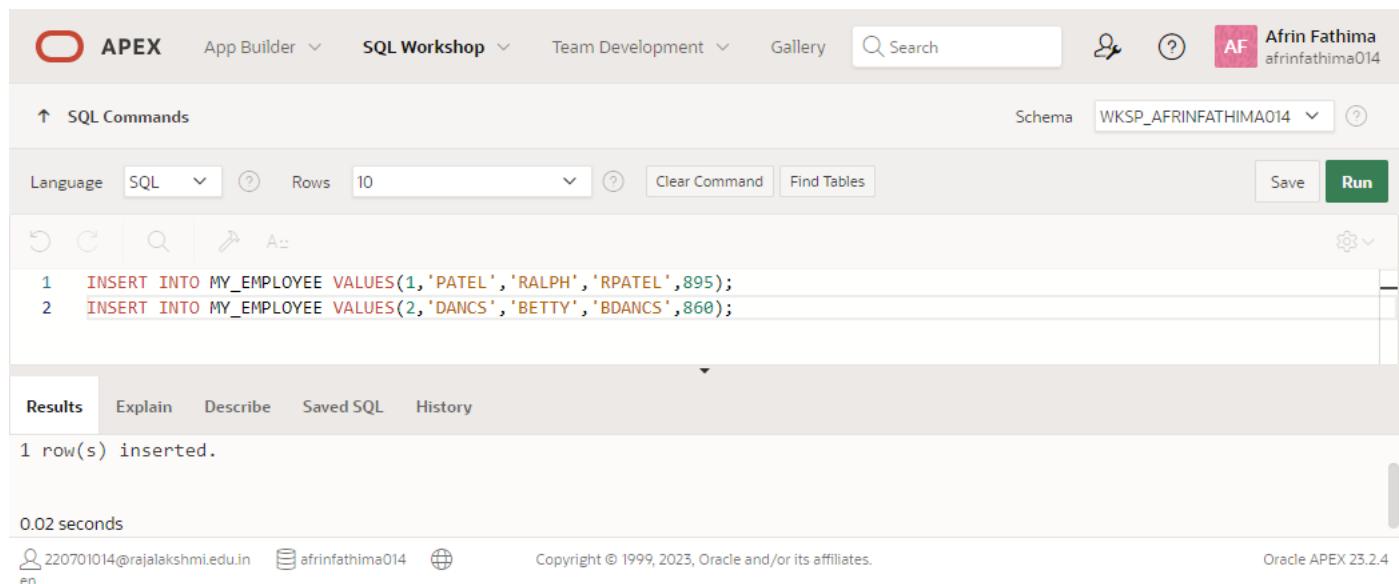
2. Add the first and second rows data to MY_EMPLOYEE table from the following sample data.

ID	Last_name	First_name	Userid	salary
1	Patel	Ralph	rpatel	895
2	Dancs	Betty	bdancs	860
3	Biri	Ben	bbiri	1100
4	Newman	Chad	Cnewman	750
5	Ropebur	Audrey	aropebur	1550

QUERY:

```
INSERT INTO MY_EMPLOYEE VALUES(1,'PATEL','RALPH','RPATEL',895);
INSERT INTO MY_EMPLOYEE VALUES(2,'DANCS','BETTY','BDANCS',860);
```

OUTPUT:



The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for App Builder, SQL Workshop (selected), Team Development, Gallery, and a search bar. The user profile 'Afrin Fathima' is visible on the right. Below the toolbar, the schema is set to 'WKSP_AFRINFATHIMA014'. The main area is titled 'SQL Commands' and contains the following SQL code:

```
1  INSERT INTO MY_EMPLOYEE VALUES(1,'PATEL','RALPH','RPATEL',895);
2  INSERT INTO MY_EMPLOYEE VALUES(2,'DANCS','BETTY','BDANCS',860);
```

The results section shows the output: '1 row(s) inserted.' Below the results, the execution time is listed as '0.02 seconds'. The footer displays copyright information for Oracle and the APEX version 'Oracle APEX 23.2.4'.

3. Display the table with values.

QUERY:

```
SELECT * FROM MY_EMPLOYEE;
```

OUTPUT:

The screenshot shows the Oracle APEX interface with the SQL Workshop tab selected. The command entered is `SELECT * FROM MY_EMPLOYEE;`. The results section displays two rows of data:

ID	LAST_NAME	FIRST_NAME	USERID	SALARY
1	PATEL	RALPH	RPATEL	895
2	DANCS	BETTY	BDANCS	860

Below the table, it says "2 rows returned in 0.00 seconds".

4. Populate the next three rows of data from the sample data. Concatenate the first letter of the first_name with the first seven characters of the last_name to produce Userid.

QUERY:

```
INSERT INTO MY_EMPLOYEE VALUES(3,'BIRI','BEN','BBIRI',1100);
INSERT INTO MY_EMPLOYEE VALUES(4,'NEWMAN','CHAD','CNEWMAN',750);
INSERT INTO MY_EMPLOYEE VALUES(5,'ROPEBUR','AUDREY','AROPEBUR',1550);
```

OUTPUT:

The screenshot shows the Oracle APEX interface with the SQL Workshop tab selected. The commands entered are the three `INSERT` statements provided in the question. The results section shows the message "1 row(s) inserted.".

5. Make the data additions permanent.

QUERY:

```
SELECT * FROM MY_EMPLOYEE;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for App Builder, SQL Workshop, Team Development, and Gallery, along with a search bar and user profile information for 'Afrin Fathima' (afrinfathima014). The main workspace is titled 'SQL Commands' and contains a SQL editor with the following command:

```
1 SELECT * FROM MY_EMPLOYEE;
```

Below the editor, there are tabs for 'Results', 'Explain', 'Describe', 'Saved SQL', and 'History'. The 'Results' tab is selected, displaying a table with the following data:

ID	LAST_NAME	FIRST_NAME	USERID	SALARY
1	PATEL	RALPH	RPATEL	895
2	DANCS	BETTY	BDANCS	860
3	BIRI	BEN	BBIRI	1100
4	NEWMAN	CHAD	CNEWMAN	750
5	ROPEBUR	AUDREY	AROPEBUR	1550

Below the table, it says '5 rows returned in 0.00 seconds' and provides a 'Download' link. At the bottom, there are footer links for 'en', 'afrinfathima014', and 'Copyright © 1999, 2023, Oracle and/or its affiliates.' On the right, it says 'Oracle APEX 23.2.4'.

6. Change the last name of employee 3 to Drexler.

QUERY:

```
UPDATE MY_EMPLOYEE SET LAST_NAME='DREXLER' WHERE ID=3;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface, similar to the previous one but with a different command in the editor. The top navigation bar and user profile are identical.

The main workspace is titled 'SQL Commands' and contains a SQL editor with the following command:

```
1 UPDATE MY_EMPLOYEE SET LAST_NAME='DREXLER' WHERE ID=3;
```

Below the editor, there are tabs for 'Results', 'Explain', 'Describe', 'Saved SQL', and 'History'. The 'Results' tab is selected, displaying the message '1 row(s) updated.' and '0.01 seconds'.

At the bottom, there are footer links for 'en', 'afrinfathima014', and 'Copyright © 1999, 2023, Oracle and/or its affiliates.' On the right, it says 'Oracle APEX 23.2.4'.

7. Change the salary to 1000 for all the employees with a salary less than 900.

QUERY:

```
UPDATE MY_EMPLOYEE SET SALARY=1000 WHERE SALARY < 900;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes 'APEX', 'App Builder', 'SQL Workshop' (which is selected), 'Team Development', 'Gallery', a search bar, and a user profile for 'Afrin Fathima'. The main area is titled 'SQL Commands' with a sub-section '1 UPDATE MY_EMPLOYEE SET SALARY=1000 WHERE SALARY<900;'. Below the command is a results section showing '3 row(s) updated.' The bottom footer displays copyright information for Oracle and the APEX version.

8. Delete Betty Dancs from MY_EMPLOYEE table.

QUERY:

```
DELETE FROM MY_EMPLOYEE WHERE FIRST_NAME='BETTY' AND LAST_NAME='DANCS';
```

OUTPUT:

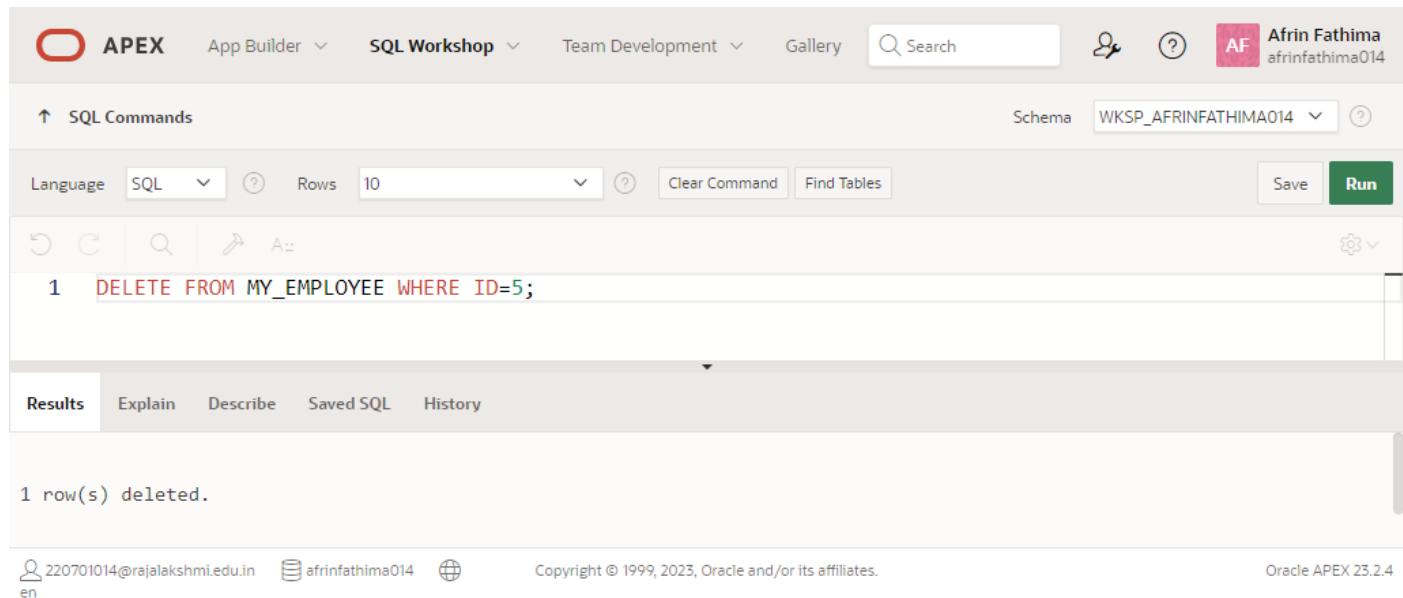
The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes 'APEX', 'App Builder', 'SQL Workshop' (selected), 'Team Development', 'Gallery', a search bar, and a user profile for 'Afrin Fathima'. The main area is titled 'SQL Commands' with a sub-section '1 DELETE FROM MY_EMPLOYEE WHERE FIRST_NAME='BETTY' AND LAST_NAME='DANCS';'. Below the command is a results section showing '1 row(s) deleted.' The bottom footer displays copyright information for Oracle and the APEX version.

9. Empty the fourth row of the emp table.

QUERY:

DELETE FROM MY_EMPLOYEE WHERE ID=5;

OUTPUT:



The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for App Builder, SQL Workshop, Team Development, and Gallery, along with a search bar and user information for 'Afrin Fathima' (afrinfathima014). The main area is titled 'SQL Commands' and contains a command input field with the following content:

```
1  DELETE FROM MY_EMPLOYEE WHERE ID=5;
```

Below the command, there are tabs for 'Results', 'Explain', 'Describe', 'Saved SQL', and 'History'. The 'Results' tab is selected, displaying the output: '1 row(s) deleted.' At the bottom of the page, footer information includes the user's email (220701014@rajalakshmi.edu.in), the schema name (afrinfathima014), copyright notice (Copyright © 1999, 2023, Oracle and/or its affiliates), and the software version (Oracle APEX 23.2.4).

Evaluation Procedure	Marks awarded
Query(5)	
Execution (5)	
Viva(5)	
Total (15)	
Faculty Signature	

RESULT :

INCLUDING CONSTRAINTS

EX-NO : 3

DATE:

1. Add a table-level PRIMARY KEY constraint to the EMP table on the ID column. The constraint should be named at creation. Name the constraint my_emp_id_pk.

QUERY:

```
ALTER TABLE EMP ADD CONSTRAINT my_emp_id_pk PRIMARY KEY(ID);
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery. A search bar and user profile are also present. The main workspace is titled "SQL Commands". The schema dropdown is set to "WKSP_AFRINFATHIMA014". The SQL language dropdown is set to "SQL". The command input field contains the SQL statement: "1 ALTER TABLE EMP ADD CONSTRAINT my_emp_id_pk PRIMARY KEY(ID);". Below the command, the results section displays the output: "Table altered." and "0.08 seconds". The bottom footer includes copyright information for Oracle and the APEX version: "Copyright © 1999, 2023, Oracle and/or its affiliates." and "Oracle APEX 23.2.4".

2. Create a PRIMARY KEY constraint to the DEPT table using the ID column. The constraint should be named at creation. Name the constraint my_dept_id_pk.

QUERY:

```
ALTER TABLE DEPT ADD CONSTRAINT my_dept_id_pk PRIMARY KEY(ID);
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for App Builder, SQL Workshop, Team Development, and Gallery, along with a search bar and user profile information for 'Afrin Fathima' (afirnfathima014). The main workspace is titled 'SQL Commands' and shows the schema 'WKSP_AFRINFATHIMA014'. The SQL editor contains the command: `ALTER TABLE DEPT ADD CONSTRAINT my_dept_id_pk PRIMARY KEY(ID);`. Below the editor, the 'Results' tab is selected, displaying the output: 'Table altered.' At the bottom, the footer includes user details (220701014@rajalakshmi.edu.in, afrinfathima014), copyright information (Copyright © 1999, 2023, Oracle and/or its affiliates.), and the version 'Oracle APEX 23.2.4'.

3. Add a column DEPT_ID to the EMP table. Add a foreign key reference on the EMP table that ensures that the employee is not assigned to nonexistent department. Name the constraint my_emp_dept_id_fk.

QUERY:

```
ALTER TABLE EMP ADD CONSTRAINT MY_EMP_DEPT_ID_FK FOREIGN  
KEY(DEPT_ID) REFERENCES DEPT(ID);
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery, along with a search bar and user profile information for 'Afrin Fathima' (afrinfathima014). The main workspace is titled 'SQL Commands'. It features a toolbar with icons for undo, redo, search, and other database operations. Below the toolbar, a command line displays the SQL statement: 'ALTER TABLE EMP ADD CONSTRAINT MY_EMP_DEPT_ID_FK FOREIGN KEY(DEPT_ID) REFERENCES DEPT(ID);'. The 'Run' button is highlighted in green. At the bottom of the workspace, tabs for 'Results', 'Explain', 'Describe', 'Saved SQL', and 'History' are visible. The results section shows the output: 'Table altered.'.

```
ALTER TABLE EMP ADD CONSTRAINT MY_EMP_DEPT_ID_FK FOREIGN KEY(DEPT_ID) REFERENCES DEPT(ID);
```

Table altered.

4. Modify the EMP table. Add a COMMISSION column of NUMBER data type, precision 2, scale 2. Add a constraint to the commission column that ensures that a commission value is greater than zero.

QUERY:

```
ALTER TABLE EMP ADD COMMISSION NUMBER(2,2) CONSTRAINT  
my_emp_commission CHECK(COMMISSION>0);
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery, along with a search bar and user profile information for 'Afrin Fathima'.

The main workspace is titled 'SQL Commands'. It features a toolbar with icons for Undo, Redo, Find, Replace, and Run. The schema dropdown is set to 'WKSP_AFRINFATHIMA014'. The SQL editor contains the following command:

```
1  ALTER TABLE EMP ADD COMMISSION NUMBER(2,2) CONSTRAINT my_emp_commission CHECK(COMMISSION>0);
```

The results section shows the output of the command:

```
Table altered.
```

Execution details at the bottom indicate it took '0.06 seconds' and was run by user '220701014@rajalakshmi.edu.in' (en) on 'afrinfathima014'.

Evaluation Procedure	Marks awarded
Query(5)	
Execution (5)	
Viva(5)	
Total (15)	
Faculty Signature	

RESULT :

WRITING BASIC SQL SELECT STATEMENTS

EX-NO : 4

DATE:

1. The following statement executes successfully.

Identify the Errors

```
SELECT employee_id, last_name  
sal*12 ANNUAL SALARY  
FROM employees;
```

QUERY:

```
SELECT EMPLOYEE_ID, LAST_NAME, SALARY*12 "ANNUAL SALARY"  
FROM EMPLOYEES;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery, along with a search bar and user profile information for 'Afrin Fathima'.

The main workspace is titled 'SQL Commands'. It features a toolbar with icons for undo, redo, search, and other functions. Below the toolbar, the query is displayed:

```
1  SELECT EMPLOYEE_ID, LAST_NAME, SALARY*12 "ANNUAL SALARY"  
2  FROM EMPLOYEES;
```

The results tab is selected, showing the output of the query:

EMPLOYEE_ID	LAST_NAME	ANNUAL SALARY
176	EMANUEL	144000
1	JAY	160801.44
2	UMA	84009.84
4	JANE	280801.44
3	PARTHI	127809.84

At the bottom of the results pane, it says '5 rows returned in 0.02 seconds' and provides download options. The footer contains copyright information for Oracle and the APEX version.

2. Show the structure of departments the table. Select all the data from it.

QUERY:

DESC DEPARTMENT;

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for App Builder, SQL Workshop, Team Development, and Gallery, along with a search bar and user profile information for 'Afrin Fathima' (afrafathima014). The main workspace is titled 'SQL Commands' and contains the SQL command 'DESC DEPARTMENT;'. Below the command, the results are displayed in a table format under the 'Describe' tab. The table shows the structure of the 'DEPARTMENT' table with four columns: DEPT_ID, DEPT_NAME, MANAGER_ID, and LOCATION_ID. The 'DEPT_ID' column is defined as a NUMBER type with precision 6 and scale 0, and is marked as a primary key and nullable. The 'DEPT_NAME' column is defined as a VARCHAR2 type with length 20 and is marked as nullable. The 'MANAGER_ID' column is defined as a NUMBER type with precision 6 and scale 0, and is marked as a primary key and nullable. The 'LOCATION_ID' column is defined as a NUMBER type with precision 4 and scale 0, and is marked as nullable. At the bottom of the page, there are footer links for copyright information and Oracle APEX version 23.2.4.

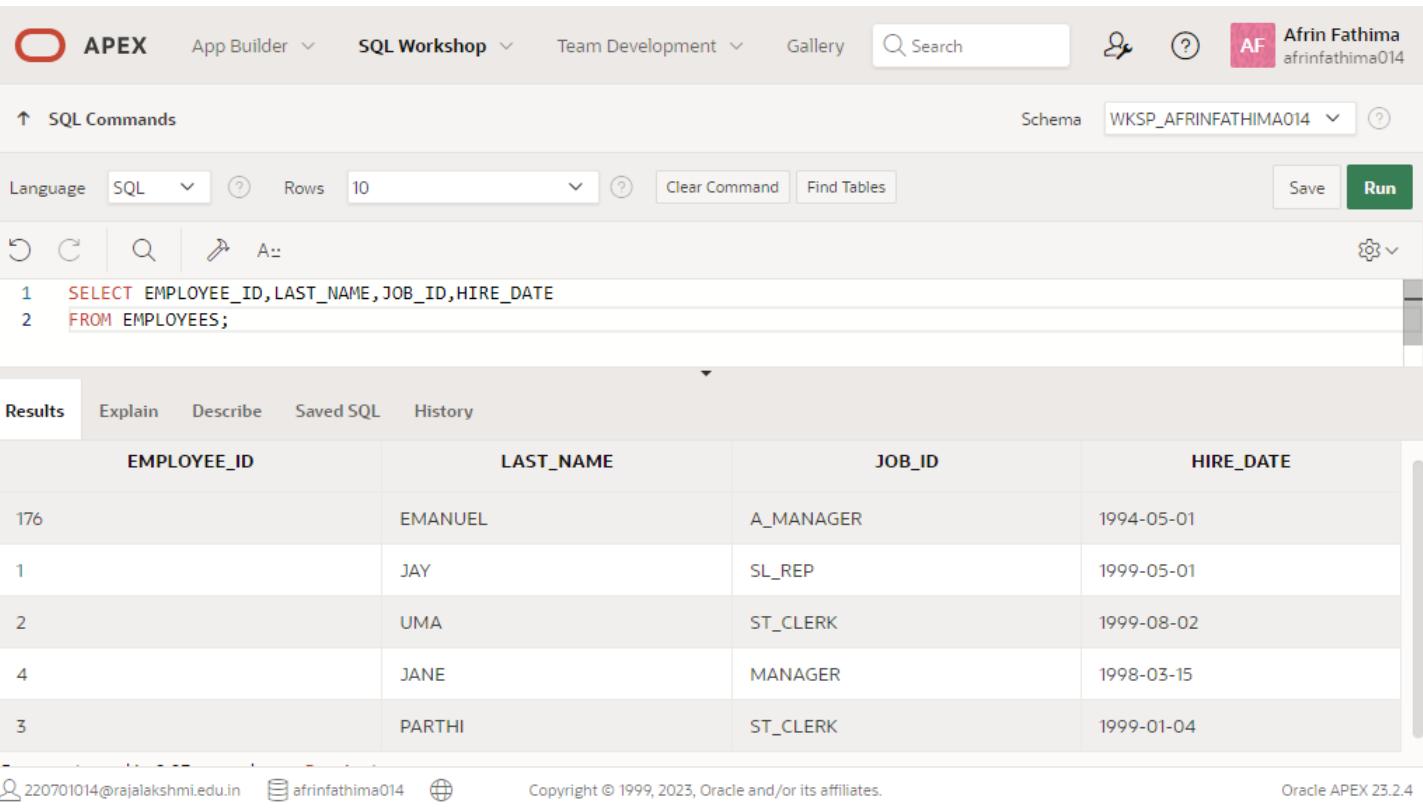
Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
DEPARTMENT	DEPT_ID	NUMBER	-	6	0	-	-	-	-
	DEPT_NAME	VARCHAR2	20	-	-	-	✓	-	-
	MANAGER_ID	NUMBER	-	6	0	-	✓	-	-
	LOCATION_ID	NUMBER	-	4	0	-	✓	-	-

3. Create a query to display the last name, job code, hire date, and employee number for each employee, with employee number appearing first.

QUERY:

```
SELECT EMPLOYEE_ID, LAST_NAME, JOB_ID, HIRE_DATE  
FROM EMPLOYEES;
```

OUTPUT:



The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for App Builder, SQL Workshop, Team Development, and Gallery, along with a search bar and user profile information for 'Afrin Fathima' (afrafathima014). The main workspace is titled 'SQL Commands' and shows the following SQL code:

```
1 SELECT EMPLOYEE_ID, LAST_NAME, JOB_ID, HIRE_DATE  
2 FROM EMPLOYEES;
```

The results tab is selected, displaying the output of the query:

EMPLOYEE_ID	LAST_NAME	JOB_ID	HIRE_DATE
176	EMANUEL	A_MANAGER	1994-05-01
1	JAY	SL REP	1999-05-01
2	UMA	ST_CLERK	1999-08-02
4	JANE	MANAGER	1998-03-15
3	PARTHI	ST_CLERK	1999-01-04



220701014@rajalakshmi.edu.in

afrafathima014



Copyright © 1999, 2023, Oracle and/or its affiliates.

Oracle APEX 23.2.4

4. Provide an alias STARTDATE for the hire date.

QUERY:

```
SELECT HIRE_DATE AS "STARTDATE"  
FROM EMPLOYEES;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. At the top, there are navigation tabs: APEX, App Builder, SQL Workshop (which is selected), Team Development, and Gallery. There is also a search bar and a user profile for Afrin Fathima (afrafathima014). Below the tabs, the schema is set to WKSP_AFRINFATHIMA014. The main area shows the SQL command entered:

```
1 SELECT HIRE_DATE AS "STARTDATE"  
2 FROM EMPLOYEES;
```

Below the command, the results tab is selected, showing the output:

STARTDATE
1994-05-01
1999-05-01
1999-08-02
1998-03-15
1999-01-04

At the bottom, it says "5 rows returned in 0.01 seconds" and provides a "Download" link. The footer includes copyright information for Oracle and links to user profiles and help.

5. Create a query to display unique job codes from the employee table.

QUERY:

```
SELECT DISTINCT JOB_ID  
FROM EMPLOYEES;
```

OUTPUT:

The screenshot shows the Oracle APEX interface with the SQL Workshop module selected. The query `SELECT DISTINCT JOB_ID FROM EMPLOYEES;` is entered in the command editor. The results page displays the unique job codes: A_MANAGER, ST_CLERK, MANAGER, and SL_REP.

JOB_ID
A_MANAGER
ST_CLERK
MANAGER
SL_REP

4 rows returned in 0.01 seconds

Copyright © 1999, 2023, Oracle and/or its affiliates.

6. Display the last name concatenated with the job ID , separated by a comma and space, and name the column EMPLOYEE and TITLE.

QUERY:

```
SELECT LAST_NAME || ',' || ' ' || JOB_ID as "EMPLOYEE and TITLE"  
from EMPLOYEES;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for App Builder, SQL Workshop, Team Development, and Gallery, along with a search bar and user profile information for 'Afrin Fathima'.

In the SQL Commands section, the schema is set to 'WKSP_AFRINFATHIMA014'. The query entered is:

```
1 SELECT LAST_NAME||','||' '||JOB_ID as "EMPLOYEE and TITLE"  
2 from EMPLOYEES;
```

The Results tab displays the output of the query:

EMPLOYEE and TITLE
EMANUEL, A_MANAGER
JAY, SL_REP
UMA, ST_CLERK
JANE, MANAGER
PARTHI, ST_CLERK

Below the results, it says '5 rows returned in 0.01 seconds' and provides a download link. The bottom footer includes copyright information for Oracle and the APEX version 'Oracle APEX 23.2.4'.

7. Create a query to display all the data from the employees table. Separate each column by a comma. Name the column THE_OUTPUT.

QUERY:

```
SELECT EMPLOYEE_ID||','||FIRST_NAME||','||LAST_NAME||','||EMAIL||',
'||PHONE_NUMBER||','||HIRE_DATE||','||JOB_ID||','||SALARY||','||COMMISSION_PCT||',
'||MANAGER_ID||','||DEPARTMENT_ID AS "THE_OUTPUT"
FROM EMPLOYEES;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery, along with a search bar and user profile information for 'Afrin Fathima'.

In the SQL Commands section, the schema is set to 'WKSP_AFRINFATHIMA014'. The query is:

```
1 SELECT EMPLOYEE_ID||','||FIRST_NAME||','||LAST_NAME||','||EMAIL||',
2      ||PHONE_NUMBER||','||HIRE_DATE||','||JOB_ID||','||SALARY||','||COMMISSION_PCT||',
3      ||MANAGER_ID||','||DEPARTMENT_ID AS "THE_OUTPUT"
4 FROM EMPLOYEES;
```

The Results tab displays the output of the query:

THE_OUTPUT
176 , SANA , EMANUEL , 123@GMAIL.COM , 1237 , 1994-05-01 , A_MANAGER , 12000 , .2 , , 30
1 , SAHANA , JAY , 123@GMAIL.COM , 1237 , 1999-05-01 , SL_REP , 13400.12 , .1 , 56 , 10
2 , TARA , UMA , 423@GMAIL.COM , 789 , 1999-08-02 , ST_CLERK , 7000.82 , , 22 , 50
4 , MARY , JANE , 123@GMAIL.COM , 1237 , 1998-03-15 , MANAGER , 23400.12 , .28 , , 50
3 , SANJ , PARTHI , 789@GMAIL.COM , 134 , 1999-01-04 , ST_CLERK , 10650.82 , , 71 , 20

At the bottom, it says '5 rows returned in 0.01 seconds' and provides a 'Download' link.

The footer includes copyright information for Oracle and user details: 'en 220701014@rajalakshmi.edu.in afrinfathima014' and 'Copyright © 1999, 2023, Oracle and/or its affiliates.'

Evaluation Procedure	Marks awarded
Query(5)	
Execution (5)	
Viva(5)	
Total (15)	
Faculty Signature	

RESULT :

RESTRICTING AND SORTING DATA

EX-NO : 5

DATE:

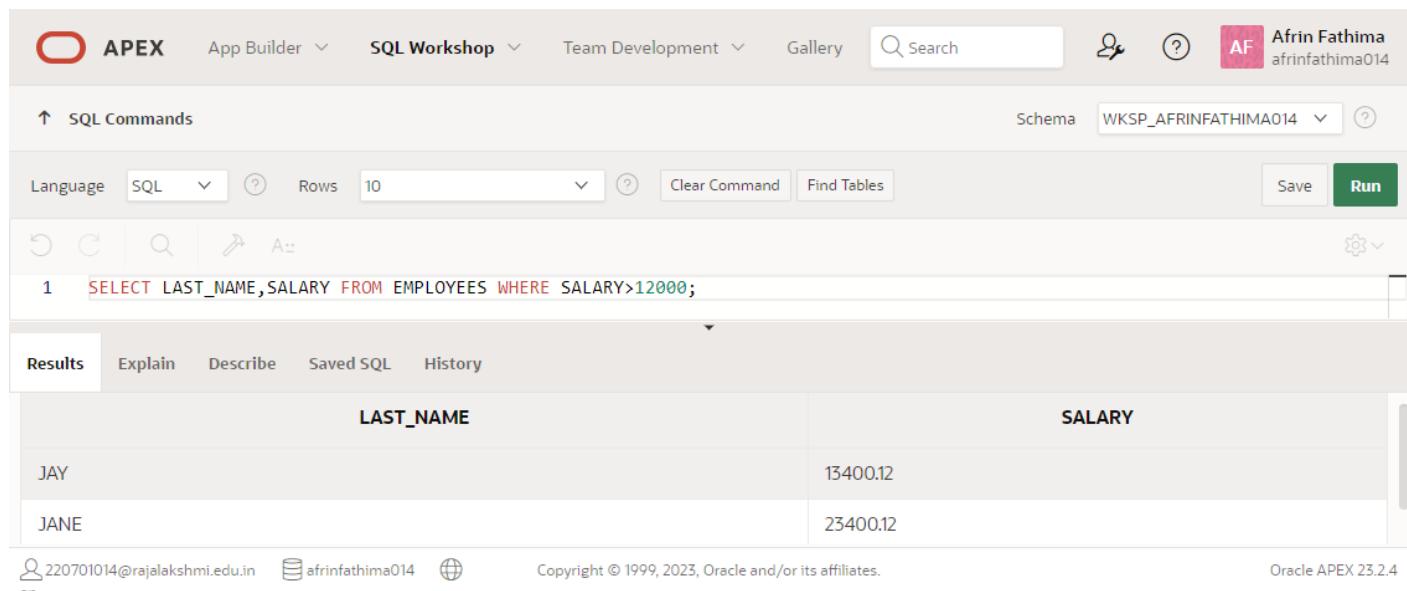
Find the Solution for the following :

1. Create a query to display the last name and salary of employees earning more than 12000.

QUERY:

```
SELECT LAST_NAME, SALARY FROM EMPLOYEES WHERE SALARY>12000;
```

OUTPUT:



The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for App Builder, SQL Workshop, Team Development, and Gallery, along with a search bar and user profile information for 'Afrin Fathima' (afirinfathima014). The main workspace is titled 'SQL Commands'. It features a toolbar with icons for undo, redo, search, and refresh, and dropdown menus for Language (set to SQL), Rows (set to 10), Clear Command, Find Tables, Save, and Run. Below the toolbar, the SQL command is entered: '1 SELECT LAST_NAME, SALARY FROM EMPLOYEES WHERE SALARY>12000;'. The results section displays the output in a table:

LAST_NAME	SALARY
JAY	13400.12
JANE	23400.12

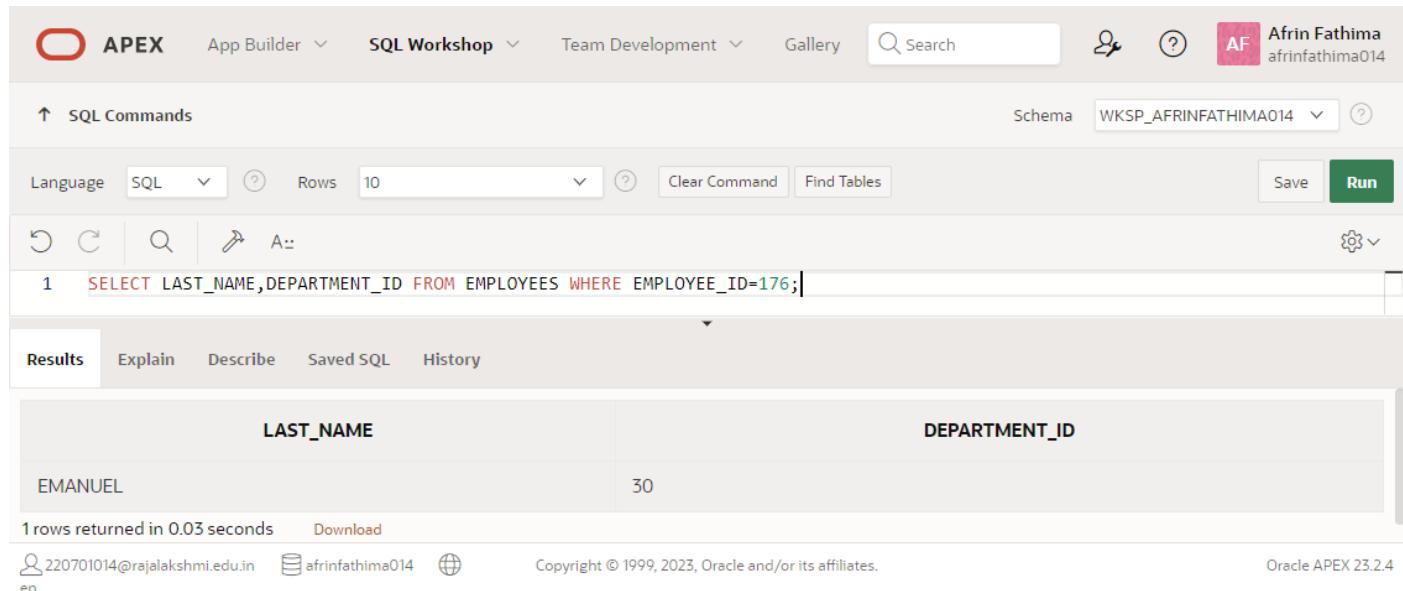
At the bottom of the page, there are footer links for 220701014@rajalakshmi.edu.in, afirinfathima014, and Oracle APEX 23.2.4.

2. Create a query to display the employee last name and department number for employee number 176.

QUERY:

```
SELECT LAST_NAME, DEPARTMENT_ID FROM EMPLOYEES WHERE  
EMPLOYEE_ID=176;
```

OUTPUT:



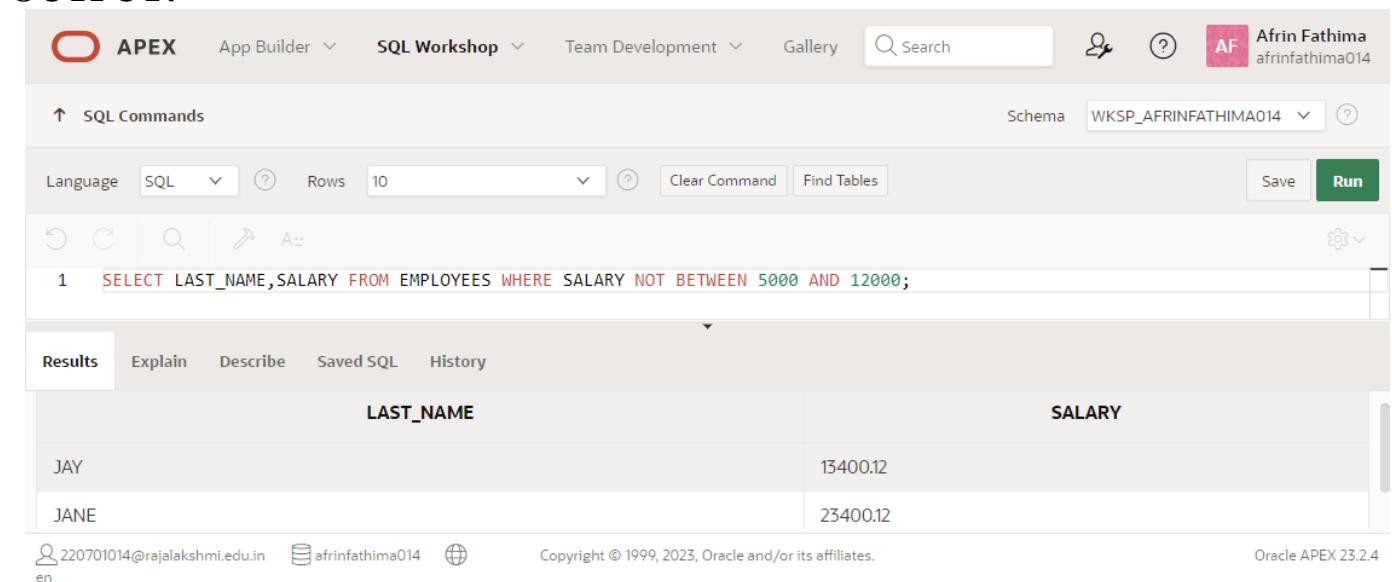
The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, Gallery, and a search bar. The user is logged in as 'Afrin Fathima' (afrafathima014). The SQL Commands tab is selected, showing the query: 'SELECT LAST_NAME,DEPARTMENT_ID FROM EMPLOYEES WHERE EMPLOYEE_ID=176;'. The results section displays a single row: EMANUEL in the LAST_NAME column and 30 in the DEPARTMENT_ID column. The bottom status bar indicates '1 rows returned in 0.03 seconds' and shows the copyright notice 'Copyright © 1999, 2023, Oracle and/or its affiliates.' and the version 'Oracle APEX 23.2.4'.

3. Create a query to display the last name and salary of employees whose salary is not in the range of 5000 and 12000. (Hint: not between)

QUERY:

```
SELECT LAST_NAME,SALARY FROM EMPLOYEES WHERE SALARY NOT  
BETWEEN 5000 AND 12000;
```

OUTPUT:



The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, Gallery, and a search bar. The user is logged in as 'Afrin Fathima' (afrafathima014). The SQL Commands tab is selected, showing the query: 'SELECT LAST_NAME,SALARY FROM EMPLOYEES WHERE SALARY NOT BETWEEN 5000 AND 12000;'. The results section displays two rows: JAY with a salary of 13400.12 and JANE with a salary of 23400.12. The bottom status bar indicates '2 rows returned in 0.03 seconds' and shows the copyright notice 'Copyright © 1999, 2023, Oracle and/or its affiliates.' and the version 'Oracle APEX 23.2.4'.

4. Display the employee last name, job ID, and start date of employees hired between February 20,1998 and May 1,1998. Order the query in ascending order by start date.

QUERY:

```
SELECT LAST_NAME, JOB_ID, TO_CHAR(HIRE_DATE, 'YYYY-MM-DD') AS HIRE_DATE  
FROM EMPLOYEES WHERE HIRE_DATE BETWEEN TO_DATE('1998-02-20','YYYY-MM-DD')  
AND TO_DATE('1998-05-01','YYYY-MM-DD') ORDER BY HIRE_DATE ;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes 'APEX', 'App Builder', 'SQL Workshop' (selected), 'Team Development', 'Gallery', a search bar, and user information for 'Afrin Fathima' (afrinfathima014). Below the navigation is a toolbar with icons for undo, redo, search, and other functions, followed by dropdowns for 'Language' (set to 'SQL'), 'Rows' (set to 10), and buttons for 'Clear Command', 'Find Tables', 'Save', and 'Run'. The main area contains the SQL query:

```
1 SELECT LAST_NAME, JOB_ID, TO_CHAR(HIRE_DATE, 'YYYY-MM-DD') AS HIRE_DATE FROM EMPLOYEES WHERE HIRE_DATE  
2 BETWEEN TO_DATE('1998-02-20','YYYY-MM-DD') AND TO_DATE('1998-05-01','YYYY-MM-DD') ORDER BY HIRE_DATE ;
```

Below the query is a results table with three columns: 'LAST_NAME', 'JOB_ID', and 'HIRE_DATE'. The data shows two rows:

LAST_NAME	JOB_ID	HIRE_DATE
JANE	MANAGER	1998-03-15
PARTHI	ST_CLERK	1998-04-12

At the bottom left are user details: '220701014@rajalakshmi.edu.in' and 'afrinfathima014'. At the bottom center is the copyright notice: 'Copyright © 1999, 2023, Oracle and/or its affiliates.' On the right side, it says 'Oracle APEX 23.2.4'.

5. Display the last name and department number of all employees in departments 20 and 50 in alphabetical order by name. (Hint: in, order by)

QUERY:

```
SELECT LAST_NAME, DEPARTMENT_ID FROM EMPLOYEES WHERE DEPARTMENT_ID IN  
(20,50) ORDER BY LAST_NAME;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface, identical to the previous one but with a different query. The top navigation bar, toolbar, and results table structure are the same. The query in the main area is:

```
1 SELECT LAST_NAME, DEPARTMENT_ID FROM EMPLOYEES WHERE DEPARTMENT_ID IN (20,50) ORDER BY LAST_NAME;
```

The results table has two columns: 'LAST_NAME' and 'DEPARTMENT_ID'. The data shows three rows:

LAST_NAME	DEPARTMENT_ID
JANE	50
PARTHI	20
UMA	50

At the bottom left are user details: '220701014@rajalakshmi.edu.in' and 'afrinfathima014'. At the bottom center is the copyright notice: 'Copyright © 1999, 2023, Oracle and/or its affiliates.' On the right side, it says 'Oracle APEX 23.2.4'.

6. Display the last name and salary of all employees who earn between 5000 and 12000 and are in departments 20 and 50 in alphabetical order by name. Label the columns EMPLOYEE, MONTHLY SALARY respectively. (Hint: between, in)

QUERY:

```
SELECT LAST_NAME AS EMPLOYEE, SALARY AS "MONTHLY SALARY" FROM EMPLOYEES  
WHERE SALARY BETWEEN 5000 AND 12000 AND DEPARTMENT_ID IN (20,50)  
ORDER BY LAST_NAME;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery, along with a search bar and user profile information. The main area is titled 'SQL Commands' and shows the following SQL code:

```
1 SELECT LAST_NAME AS EMPLOYEE, SALARY AS "MONTHLY SALARY" FROM EMPLOYEES  
2 WHERE SALARY BETWEEN 5000 AND 12000 AND DEPARTMENT_ID IN (20,50)  
3 ORDER BY LAST_NAME;
```

The 'Results' tab is selected, displaying the output:

EMPLOYEE	MONTHLY SALARY
PARTHI	10650.82
UMA	7000

At the bottom, there are footer links for support, copyright notice (Copyright © 1999, 2023, Oracle and/or its affiliates), and version information (Oracle APEX 23.2.4).

7. Display the last name and hire date of every employee who was hired in 1994.(Hint: like)

QUERY:

```
SELECT LAST_NAME, TO_CHAR(HIRE_DATE, 'YYYY-MM-DD') AS HIRE_DATE FROM  
EMPLOYEES WHERE HIRE_DATE LIKE '%1994';
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery, along with a search bar and user profile information. The main area is titled 'SQL Commands' and shows the following SQL code:

```
1 SELECT LAST_NAME, TO_CHAR(HIRE_DATE, 'YYYY-MM-DD') AS HIRE_DATE FROM EMPLOYEES WHERE HIRE_DATE LIKE '%1994';
```

The 'Results' tab is selected, displaying the output:

LAST_NAME	HIRE_DATE
EMANUEL	1994-05-01

At the bottom, there are footer links for support, copyright notice (Copyright © 1999, 2023, Oracle and/or its affiliates), and version information (Oracle APEX 23.2.4). A note at the bottom left indicates '1 rows returned in 0.01 seconds'.

8. Display the last name and job title of all employees who do not have a manager.(Hint: is null)

QUERY:

```
SELECT LAST_NAME, JOB_ID FROM EMPLOYEES WHERE MANAGER_ID IS NULL;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes 'APEX', 'App Builder', 'SQL Workshop', 'Team Development', 'Gallery', a search bar, and user information for 'Afrin Fathima'. Below the navigation is a toolbar with 'Language' set to 'SQL', 'Rows' set to 10, and buttons for 'Clear Command' and 'Find Tables'. The main area contains the SQL command: 'SELECT LAST_NAME, JOB_ID FROM EMPLOYEES WHERE MANAGER_ID IS NULL;'. The results tab is selected, displaying a table with two rows:

LAST_NAME	JOB_ID
JANE	MANAGER
EMANUEL	A_MANAGER

At the bottom, it shows the user's email (220701014@rajalakshmi.edu.in), session ID (afrinfathima014), and copyright information (Copyright © 1999, 2023, Oracle and/or its affiliates). The version is Oracle APEX 23.2.4.

9. Display the last name, salary, and commission for all employees who earn commissions. Sort data in descending order of salary and commissions. (Hint: is not null,order by)

QUERY:

```
SELECT LAST_NAME, SALARY, COMMISSION_PCT FROM EMPLOYEES WHERE  
COMMISSION_PCT IS NOT NULL ORDER BY SALARY DESC, COMMISSION_PCT DESC;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes 'APEX', 'App Builder', 'SQL Workshop', 'Team Development', 'Gallery', a search bar, and user information for 'Afrin Fathima'. Below the navigation is a toolbar with 'Language' set to 'SQL', 'Rows' set to 10, and buttons for 'Clear Command' and 'Find Tables'. The main area contains the SQL command: 'SELECT LAST_NAME, SALARY, COMMISSION_PCT FROM EMPLOYEES WHERE COMMISSION_PCT IS NOT NULL ORDER BY SALARY DESC, COMMISSION_PCT DESC;'. The results tab is selected, displaying a table with three rows:

LAST_NAME	SALARY	COMMISSION_PCT
JANE	23400.12	.28
JAY	13400.12	.1
EMANUEL	12000	.2

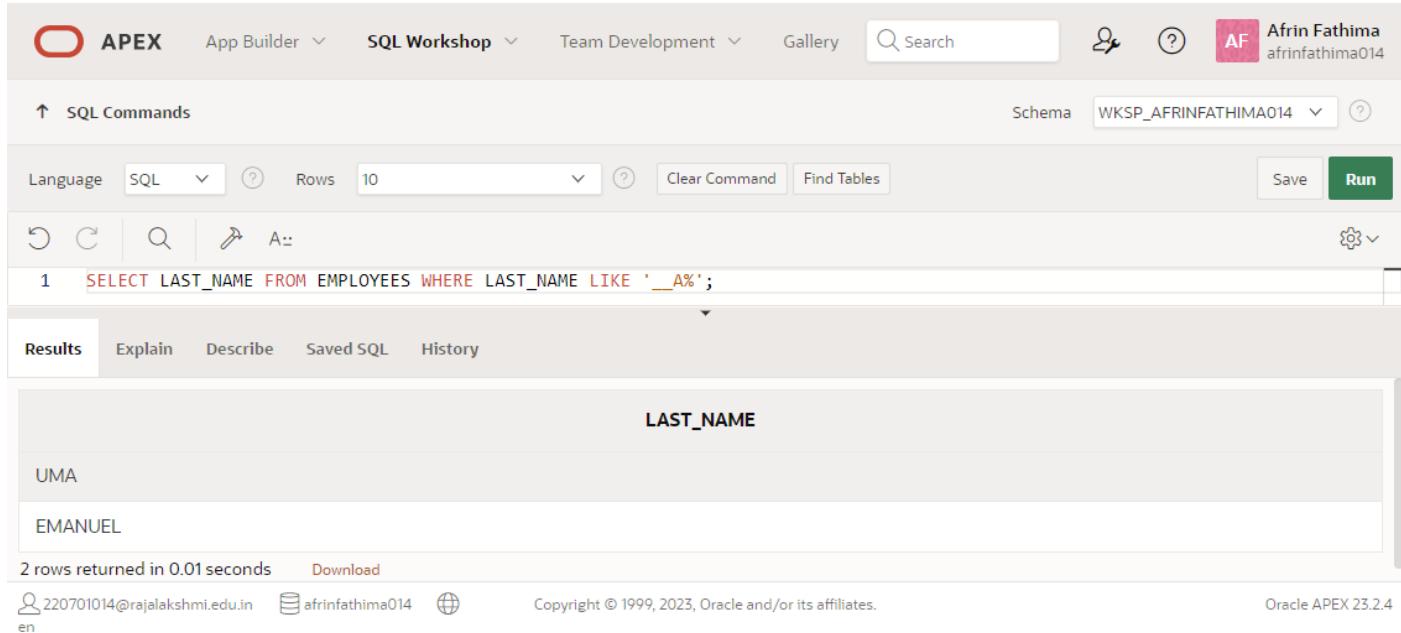
At the bottom, it shows the user's email (220701014@rajalakshmi.edu.in), session ID (afrinfathima014), and copyright information (Copyright © 1999, 2023, Oracle and/or its affiliates). The version is Oracle APEX 23.2.

10. Display the last name of all employees where the third letter of the name is *a*.

QUERY:

```
SELECT LAST_NAME FROM EMPLOYEES WHERE LAST_NAME LIKE '__A%';
```

OUTPUT:



The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for App Builder, SQL Workshop, Team Development, and Gallery, along with a search bar and user profile information. The main workspace is titled "SQL Commands" and contains the executed SQL query: "SELECT LAST_NAME FROM EMPLOYEES WHERE LAST_NAME LIKE '__A%'". The results section displays two rows of data: "UMA" and "EMANUEL". The bottom status bar indicates "2 rows returned in 0.01 seconds" and shows the copyright notice "Copyright © 1999, 2023, Oracle and/or its affiliates."

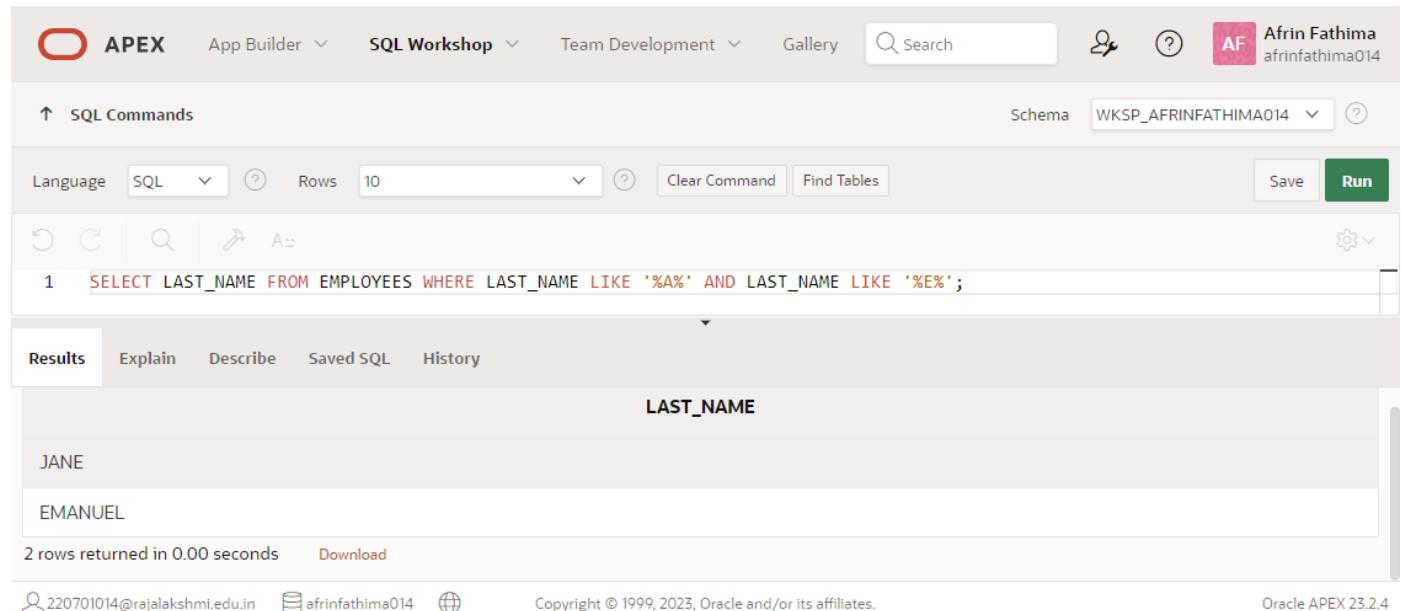
LAST_NAME
UMA
EMANUEL

11. Display the last name of all employees who have an *a* and an *e* in their last name.

QUERY:

```
SELECT LAST_NAME FROM EMPLOYEES WHERE LAST_NAME LIKE '%A%' AND LAST_NAME LIKE '%E%';
```

OUTPUT:



The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for App Builder, SQL Workshop, Team Development, and Gallery, along with a search bar and user profile information. The main workspace is titled "SQL Commands" and contains the executed SQL query: "SELECT LAST_NAME FROM EMPLOYEES WHERE LAST_NAME LIKE '%A%' AND LAST_NAME LIKE '%E%'". The results section displays two rows of data: "JANE" and "EMANUEL". The bottom status bar indicates "2 rows returned in 0.00 seconds" and shows the copyright notice "Copyright © 1999, 2023, Oracle and/or its affiliates."

LAST_NAME
JANE
EMANUEL

12. Display the last name and job and salary for all employees whose job is sales representative or stock clerk and whose salary is not equal to 2500 ,3500 or 7000.

QUERY:

```
SELECT LAST_NAME,JOB_ID,SALARY FROM EMPLOYEES WHERE JOB_ID='SA_REP'  
OR JOB_ID='ST_CLERK' AND SALARY NOT IN (2500, 3500, 7000);
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, Gallery, and a search bar. The user is logged in as 'Afrin Fathima' (af). The schema selected is 'WKSP_AFRINFATHIMA014'. The main area is titled 'SQL Commands' with a 'Language' dropdown set to 'SQL'. The query entered is: `SELECT LAST_NAME,JOB_ID,SALARY FROM EMPLOYEES WHERE JOB_ID='SA_REP' OR JOB_ID='ST_CLERK' AND SALARY NOT IN (2500, 3500, 7000);`. The results section displays a single row: PARTHI, ST_CLERK, 10650.82. A message at the bottom indicates 1 row was returned in 0.01 seconds.

LAST_NAME	JOB_ID	SALARY
PARTHI	ST_CLERK	10650.82

13. Display the last name, salary, and commission for all employees whose commission amount is 20%.

QUERY:

```
SELECT LAST_NAME,SALARY,COMMISSION_PCT FROM EMPLOYEES WHERE  
COMMISSION_PCT = .20;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, Gallery, and a search bar. The user is logged in as 'Afrin Fathima' (af). The schema selected is 'WKSP_AFRINFATHIMA014'. The main area is titled 'SQL Commands' with a 'Language' dropdown set to 'SQL'. The query entered is: `SELECT LAST_NAME,SALARY,COMMISSION_PCT FROM EMPLOYEES WHERE COMMISSION_PCT = .20;`. The results section displays a single row: EMANUEL, 12000, .2. A message at the bottom indicates 1 row was returned in 0.01 seconds.

LAST_NAME	SALARY	COMMISSION_PCT
EMANUEL	12000	.2

Evaluation Procedure	Marks awarded
Query(5)	
Execution (5)	
Viva(5)	
Total (15)	
Faculty Signature	

RESULT :

SINGLE ROW FUNCTIONS

EX-NO : 6

DATE:

1. Write a query to display the current date. Label the column Date

QUERY:

```
SELECT TO_CHAR(SYSDATE,'YYYY-MM-DD') AS "DATE" FROM DUAL;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery, along with a search bar and user profile information. The main workspace is titled 'SQL Commands' and shows the schema 'WKSP_AFRINFATHIMA014'. The SQL editor contains the following code:

```
1  SELECT TO_CHAR(SYSDATE,'YYYY-MM-DD') AS "DATE" FROM DUAL;
2
```

The results section displays the output of the query:

DATE
2024-04-10

Below the results, it says '1 rows returned in 0.03 seconds' and provides a 'Download' link. The bottom of the page shows copyright information for Oracle and the APEX version 'Oracle APEX 23.2.4'.

2. The HR department needs a report to display the employee number, last name, salary, and increased by 15.5% (expressed as a whole number) for each employee. Label the column New Salary.

QUERY:

```
SELECT EMPLOYEE_ID, LAST_NAME, SALARY, SALARY+(SALARY*0.155) AS "NEW SALARY"  
FROM EMPLOYEES;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop (selected), Team Development, and Gallery. A search bar and user profile for 'Afrin Fathima' are also present. The main area is titled 'SQL Commands' with a schema dropdown set to 'WKSP_AFRINFATHIMA014'. The SQL editor contains the following code:

```
3  SELECT EMPLOYEE_ID, LAST_NAME, SALARY, SALARY+(SALARY*0.155) AS "NEW SALARY"  
4  FROM EMPLOYEES;  
c
```

The 'Results' tab is selected, displaying the query output as a table:

EMPLOYEE_ID	LAST_NAME	SALARY	NEW SALARY
109	KOHLI	20000	23100
111	JOBS	20000	23100
101	RAVI	20000	23100
102	UMA	7000	8085
103	PARTHI	8650.82	9991.6971
108	DEV	15000	17325
112	RAVI	5500	6352.5
100	JAY	13400.12	15477.1386
104	JANE	23400.12	27027.1386
105	EMANUEL	22000	25410
106	VIJAY	33400.12	38577.1386
110	DAVIES	30000	34650
107	JAM	15000	17325

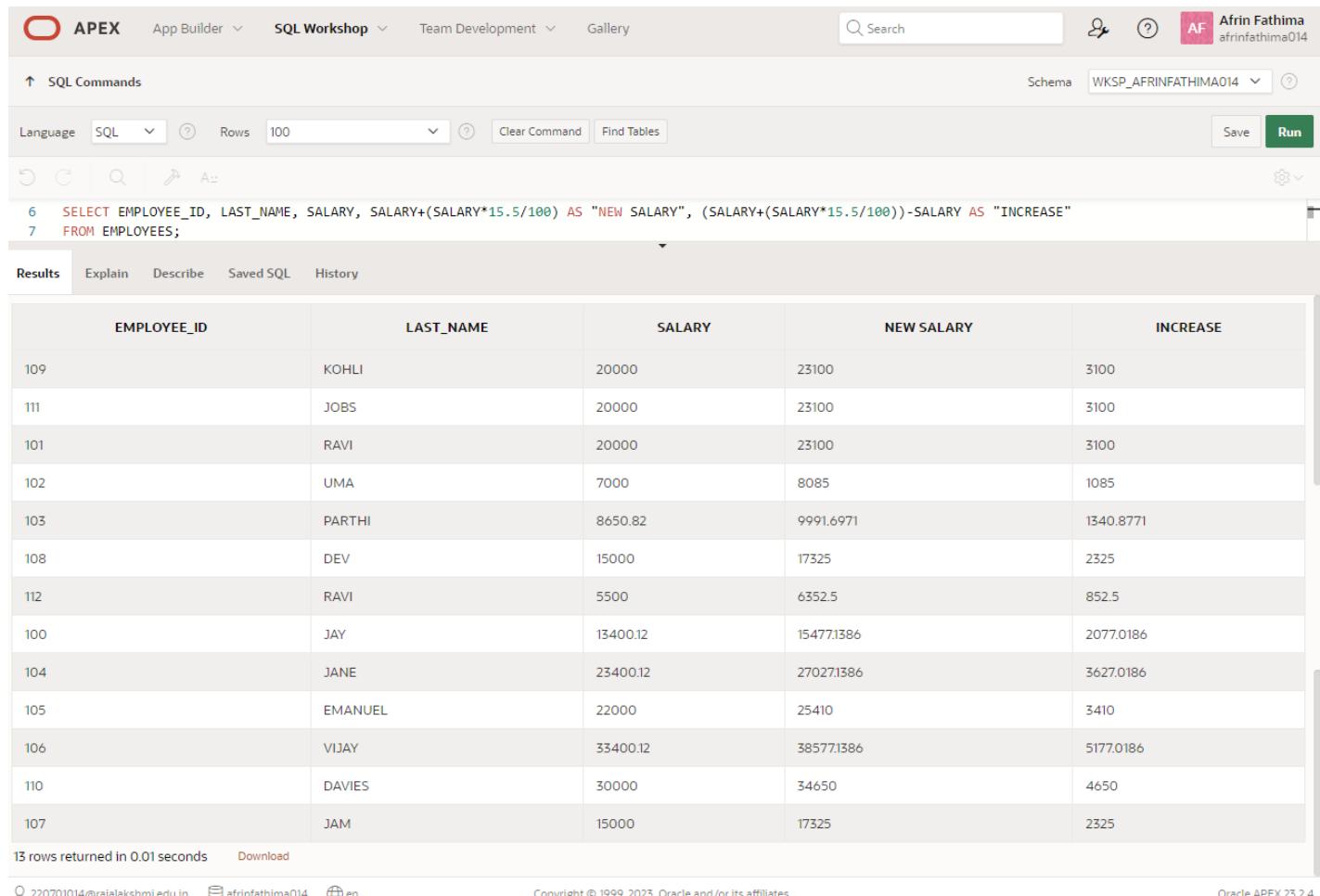
At the bottom, it says '13 rows returned in 0.00 seconds' and provides download options. The footer includes copyright information for Oracle and the APEX version '23.2.4'.

3. Modify your query lab_03_02.sql to add a column that subtracts the old salary from the new salary. Label the column Increase.

QUERY:

```
SELECT EMPLOYEE_ID, LAST_NAME, SALARY, SALARY+(SALARY*15.5/100) AS "NEW SALARY", (SALARY+(SALARY*15.5/100))-SALARY AS "INCREASE"  
FROM EMPLOYEES;
```

OUTPUT:



The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for App Builder, SQL Workshop, Team Development, and Gallery, along with a search bar and user profile information for 'Afrin Fathima'.

The SQL Commands section contains the following query:

```
6  SELECT EMPLOYEE_ID, LAST_NAME, SALARY, SALARY+(SALARY*15.5/100) AS "NEW SALARY", (SALARY+(SALARY*15.5/100))-SALARY AS "INCREASE"  
7  FROM EMPLOYEES;
```

The Results tab is selected, displaying the query results in a grid format:

EMPLOYEE_ID	LAST_NAME	SALARY	NEW SALARY	INCREASE
109	KOHLI	20000	23100	3100
111	JOBSS	20000	23100	3100
101	RAVI	20000	23100	3100
102	UMA	7000	8085	1085
103	PARTHII	8650.82	9991.6971	1340.8771
108	DEV	15000	17325	2325
112	RAVI	5500	6352.5	852.5
100	JAY	13400.12	15477.1386	2077.0186
104	JANE	23400.12	27027.1386	3627.0186
105	EMANUEL	22000	25410	3410
106	VIJAY	33400.12	38577.1386	5177.0186
110	DAVIES	30000	34650	4650
107	JAM	15000	17325	2325

At the bottom left, it says '13 rows returned in 0.01 seconds'. At the bottom right, it says 'Download' and 'Oracle APEX 23.2.4'.

4. Write a query that displays the last name (with the first letter uppercase and all other letters lowercase) and the length of the last name for all employees whose name starts with the letters J, A, or M. Give each column an appropriate label. Sort the results by the employees' last names.

QUERY:

```
SELECT INITCAP(LAST_NAME) AS "NAME",
LENGTH(LAST_NAME) AS "LENGTH OF NAME"
FROM EMPLOYEES
WHERE LAST_NAME LIKE 'J%' OR
LAST_NAME LIKE 'A%' OR
LAST_NAME LIKE 'M%'
ORDER BY LAST_NAME;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery, along with a search bar and user profile information for 'Afrin Fathima'. The main workspace is titled 'SQL Commands' and shows the executed SQL code. The results tab is selected, displaying a table with two columns: 'NAME' and 'LENGTH OF NAME'. The data rows are Jam (length 3), Jane (length 4), Jay (length 3), and Jobs (length 4).

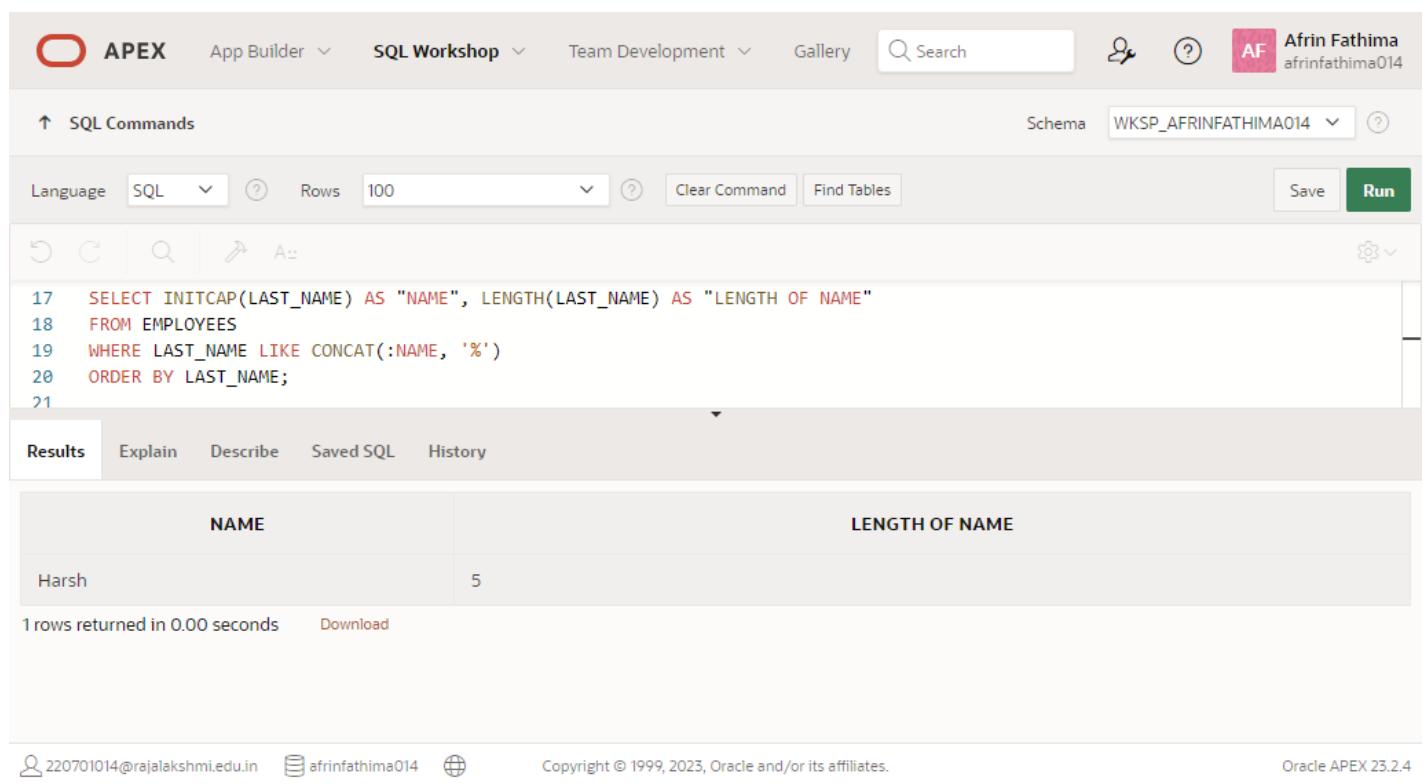
NAME	LENGTH OF NAME
Jam	3
Jane	4
Jay	3
Jobs	4

5. Rewrite the query so that the user is prompted to enter a letter that starts the last name. For example, if the user enters H when prompted for a letter, then the output should show all employees whose last name starts with the letter H.

QUERY:

```
SELECT INITCAP(LAST_NAME) AS "NAME", LENGTH(LAST_NAME) AS  
"LENGTH OF NAME"  
FROM EMPLOYEES  
WHERE LAST_NAME LIKE CONCAT(:NAME, '%')  
ORDER BY LAST_NAME;
```

OUTPUT:



The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery, along with a search bar and user profile information for 'Afrin Fathima' (afrafathima014). The main workspace is titled 'SQL Commands' and contains the following SQL code:

```
17  SELECT INITCAP(LAST_NAME) AS "NAME", LENGTH(LAST_NAME) AS "LENGTH OF NAME"  
18  FROM EMPLOYEES  
19  WHERE LAST_NAME LIKE CONCAT(:NAME, '%')  
20  ORDER BY LAST_NAME;  
21
```

The 'Results' tab is selected, displaying the output of the query:

NAME	LENGTH OF NAME
Harsh	5

Below the results, it says '1 rows returned in 0.00 seconds' and provides a 'Download' link. At the bottom of the page, there are footer links for user profiles (220701014@rajalakshmi.edu.in, afrinfathima014), copyright information (Copyright © 1999, 2023, Oracle and/or its affiliates.), and the Oracle APEX version (Oracle APEX 23.2.4).

6. The HR department wants to find the length of employment for each employee. For each employee, display the last name and calculate the number of months between today and the date on which the employee was hired. Label the column MONTHS_WORKED. Order your results by the number of months employed. Round the number of months up to the closest whole number.

QUERY:

```
SELECT LAST_NAME, ROUND(MONTHS_BETWEEN(SYSDATE,HIRE_DATE),0)
MONTHS_WORKED FROM EMPLOYEES
ORDER BY 2;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery. The SQL Workshop tab is selected. The search bar contains the placeholder "Search". The schema dropdown is set to "WKSP_AFRINFATHIMA014". The SQL command input area contains the following code:

```
28 SELECT LAST_NAME, ROUND(MONTHS_BETWEEN(SYSDATE,HIRE_DATE),0) MONTHS_WORKED FROM EMPLOYEES
29 ORDER BY 2;
```

The results section displays the output of the query:

LAST_NAME	MONTHS_WORKED
DEV	234
RAVI	243
DAVIES	263
VIJAY	280
JAM	280
UMA	296
JAY	299
PARTHI	312
JANE	313
HARSH	314
RAVI	338
JOBS	346
EMANUEL	359
KOHLI	425

At the bottom of the results panel, it says "14 rows returned in 0.01 seconds" and has a "Download" link. The footer of the page includes copyright information: "Copyright © 1999, 2023, Oracle and/or its affiliates." and "Oracle APEX 23.2.4".

7. Create a report that produces the following for each employee: earns monthly but wants . Label the column Dream Salaries.

QUERY:

```
SELECT LAST_NAME||' EARNS $'||SALARY||' MONTHLY BUT WANTS  
$'||SALARY*3 "DREAM SALARY"  
FROM EMPLOYEES;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop (selected), Team Development, and Gallery. A search bar and user profile are also present. The main area is titled 'SQL Commands'. The schema is set to 'WKSP_AFRINFATHIMA014'. The code entered is:

```
31  SELECT LAST_NAME||' EARNS $'||SALARY||' MONTHLY BUT WANTS $'||SALARY*3 "DREAM SALARY"  
32  FROM EMPLOYEES;
```

The results section displays the output for 14 rows, labeled 'DREAM SALARY'. The results are:

DREAM SALARY
HARSH EARNS \$15000 MONTHLY BUT WANTS \$45000
KOHLI EARNS \$20000 MONTHLY BUT WANTS \$60000
JOBZ EARNS \$20000 MONTHLY BUT WANTS \$60000
RAVI EARNS \$20000 MONTHLY BUT WANTS \$60000
UMA EARNS \$7000 MONTHLY BUT WANTS \$21000
PARTHI EARNS \$8650.82 MONTHLY BUT WANTS \$25952.46
DEV EARNS \$15000 MONTHLY BUT WANTS \$45000
RAVI EARNS \$5500 MONTHLY BUT WANTS \$16500
JAY EARNS \$13400.12 MONTHLY BUT WANTS \$40200.36
JANE EARNS \$23400.12 MONTHLY BUT WANTS \$70200.36
EMANUEL EARNS \$22000 MONTHLY BUT WANTS \$66000
VIJAY EARNS \$33400.12 MONTHLY BUT WANTS \$100200.36
DAVIES EARNS \$30000 MONTHLY BUT WANTS \$90000
JAM EARNS \$15000 MONTHLY BUT WANTS \$45000

At the bottom, it says '14 rows returned in 0.01 seconds' and provides download options.

8. Create a query to display the last name and salary for all employees. Format the salary to be 15 characters long, left-padded with the \$ symbol. Label the column SALARY.

QUERY:

```
SELECT LAST_NAME, LPAD(SALARY,15,'$') SALARY  
FROM EMPLOYEES;
```

OUTPUT:

The screenshot shows the Oracle SQL Workshop interface. The top navigation bar includes APEX, App Builder, SQL Workshop, Team Development, and Gallery. The right side shows a user profile for 'Afrin Fatima' with the schema 'WKSP_AFRINFATHIMA014'. The main area has tabs for Language (SQL selected), Rows (100), Clear Command, and Find Tables. On the far right are Save and Run buttons. The SQL command entered is:

```
34 SELECT LAST_NAME, LPAD(SALARY,15,'$') SALARY  
35 FROM EMPLOYEES;  
36
```

The Results tab is selected, displaying the output:

LAST_NAME	SALARY
HARSH	\$\$\$\$\$\$\$\$\$\$15000
KOHU	\$\$\$\$\$\$\$\$\$\$20000
JOBS	\$\$\$\$\$\$\$\$\$\$20000
RAVI	\$\$\$\$\$\$\$\$\$\$20000
UMA	\$\$\$\$\$\$\$\$\$\$7000
PARTHI	\$\$\$\$\$\$\$\$\$\$850.12
DEV	\$\$\$\$\$\$\$\$\$\$15000
RAVI	\$\$\$\$\$\$\$\$\$\$5500
JAY	\$\$\$\$\$\$13400.12
JANE	\$\$\$\$\$\$25400.12
EMANUEL	\$\$\$\$\$\$\$\$\$\$22000
VIJAY	\$\$\$\$\$\$33400.12
DAVIES	\$\$\$\$\$\$\$\$\$\$30000
JAM	\$\$\$\$\$\$\$\$\$\$15000

At the bottom left, it says '14 rows returned in 0.00 seconds' and there is a 'Download' button.

9. Display each employee's last name, hire date, and salary review date, which is the first Monday after six months of service. Label the column REVIEW. Format the dates to appear in the format similar to "Monday, the Thirty-First of July, 2000."

QUERY:

```
SELECT LAST_NAME, HIRE_DATE,  
TO_CHAR(NEXT_DAY(HIRE_DATE,'MONDAY')),'FMDAY," THE "DDSPHTH "OF"  
MONTH,YYYY') AS "REVIEW" FROM EMPLOYEES;
```

OUTPUT:

The screenshot shows the Oracle SQL Workshop interface. The top navigation bar includes APEX, App Builder, SQL Workshop, Team Development, and Gallery. On the right, there is a search bar, user profile, and schema information (WKSP_AFRINFATHIMA014). The main area shows the SQL command being run:

```
37 SELECT LAST_NAME, HIRE_DATE, TO_CHAR(NEXT_DAY(HIRE_DATE,'MONDAY')),'FMDAY," THE "DDSPHTH "OF"  
38 FROM EMPLOYEES;
```

The Results tab is selected, displaying the output:

LAST_NAME	HIRE_DATE	REVIEW
HARSH	02/23/1998	MONDAY, THE SECOND OF MARCH,1998
KOHLI	11/05/1988	MONDAY, THE SEVENTH OF NOVEMBER,1988
JOBSS	06/13/1995	MONDAY, THE NINETEENTH OF JUNE,1995
RAVI	01/07/2004	MONDAY, THE TWELFTH OF JANUARY,2004
UMA	08/02/1999	MONDAY, THE NINTH OF AUGUST,1999
PARTHI	04/12/1998	MONDAY, THE THIRTEENTH OF APRIL,1998
DEV	10/12/2004	MONDAY, THE EIGHTEENTH OF OCTOBER,2004
RAVI	02/15/1996	MONDAY, THE NINETEENTH OF FEBRUARY,1996
JAY	05/01/1999	MONDAY, THE THIRD OF MAY,1999
JANE	03/15/1998	MONDAY, THE SIXTEENTH OF MARCH,1998
EMANUEL	05/01/1994	MONDAY, THE SECOND OF MAY,1994
VIJAY	11/30/2000	MONDAY, THE FOURTH OF DECEMBER,2000
DAVIES	05/17/2002	MONDAY, THE TWENTIETH OF MAY,2002
JAM	12/04/2000	MONDAY, THE ELEVENTH OF DECEMBER,2000

14 rows returned in 0.01 seconds [Download](#)

10. Display the last name, hire date, and day of the week on which the employee started. Label the column DAY. Order the results by the day of the week, starting with Monday

QUERY:

```
SELECT LAST_NAME, HIRE_DATE, TO_CHAR(HIRE_DATE,'DAY') "DAY"  
FROM EMPLOYEES  
ORDER BY TO_CHAR(HIRE_DATE-1,'D');
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery. The right side shows a user profile for 'Afrin Fathima' (afrafathima014). The main area has tabs for SQL Commands, Results, Explain, Describe, Saved SQL, and History. The SQL Commands tab contains the executed SQL code:

```
40  SELECT LAST_NAME, HIRE_DATE, TO_CHAR(HIRE_DATE,'DAY') "DAY"  
41  FROM EMPLOYEES  
42  ORDER BY TO_CHAR(HIRE_DATE-1,'D');
```

The Results tab displays the query output as a table:

LAST_NAME	HIRE_DATE	DAY
UMA	08/02/1999	MONDAY
JAM	12/04/2000	MONDAY
HARSH	02/23/1998	MONDAY
JOBS	06/13/1995	TUESDAY
DEV	10/12/2004	TUESDAY
RAVI	01/07/2004	WEDNESDAY
VIJAY	11/30/2000	THURSDAY
RAVI	02/15/1996	THURSDAY
DAVIES	05/17/2002	FRIDAY
JAY	05/01/1999	SATURDAY
KOHLI	11/05/1988	SATURDAY
EMANUEL	05/01/1994	SUNDAY
PARTHI	04/12/1998	SUNDAY
JANE	03/15/1998	SUNDAY

At the bottom, it says '14 rows returned in 0.00 seconds' and provides download and history options.

Evaluation Procedure	Marks awarded
Query(5)	
Execution (5)	
Viva(5)	
Total (15)	
Faculty Signature	

RESULT :

DISPLAYING DATA FROM MULTIPLE TABLES

EX-NO : 6

DATE:

1. Write a query to display the last name, department number, and department name for all employees.

QUERY:

```
SELECT E.LAST_NAME, E.DEPARTMENT_ID, D.DEPT_NAME  
FROM EMPLOYEES E, DEPARTMENT D  
WHERE E.DEPARTMENT_ID = D.DEPARTMENT_ID;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The query entered is:

```
43 SELECT E.LAST_NAME, E.DEPARTMENT_ID, D.DEPT_NAME  
44 FROM EMPLOYEES E, DEPARTMENT D  
45 WHERE E.DEPARTMENT_ID = D.DEPARTMENT_ID;
```

The results table displays the following data:

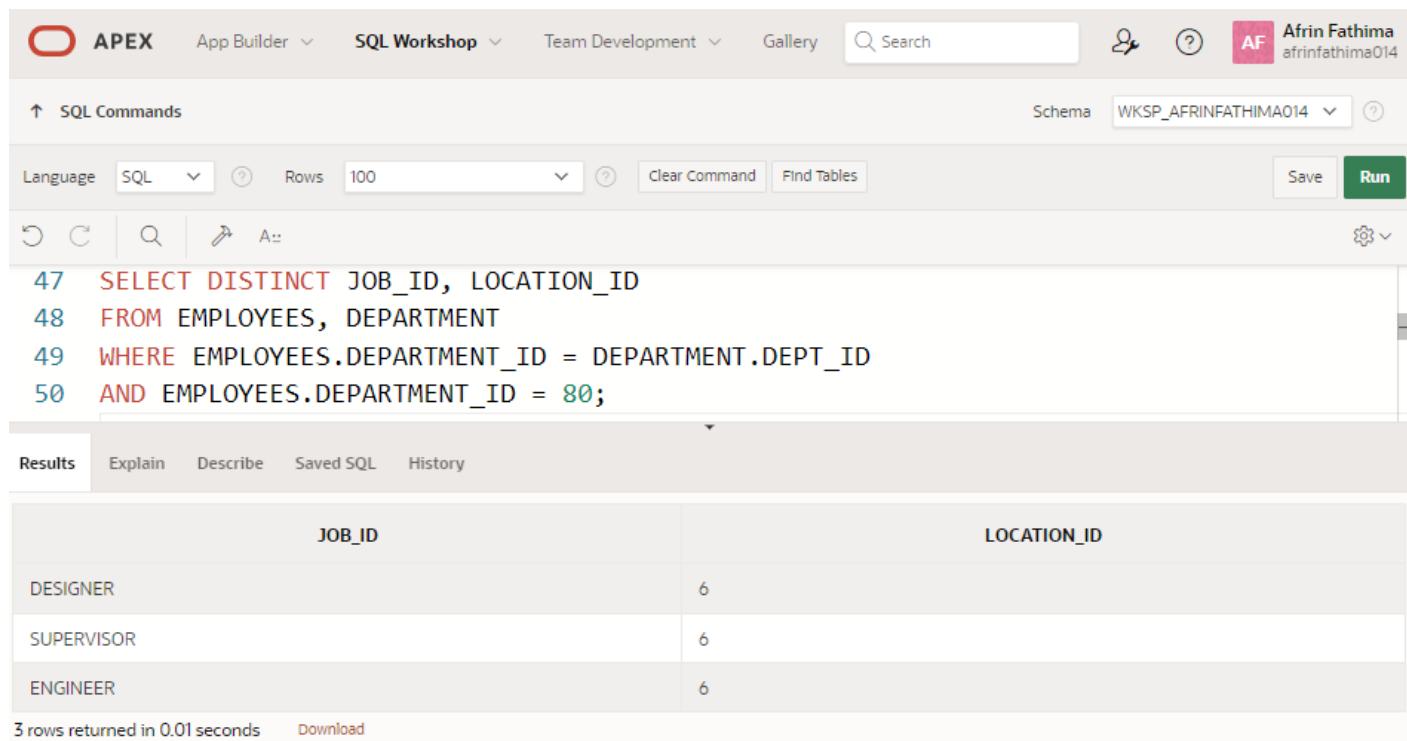
LAST_NAME	DEPARTMENT_ID	DEPT_NAME
KOHLI	80	MANUFACTURING
JOBSS	90	IT
RAVI	10	MARKETING
UMA	20	STOCK
PARTHI	20	STOCK
DEV	80	MANUFACTURING
RAVI	90	IT
JAY	10	MARKETING
HARSH	60	EXECUTIVE
SIDDHU	60	EXECUTIVE
JANE	50	HR
EMANUEL	30	FINANCE
VIJAY	40	MANAGEMENT
DAVIES	30	FINANCE
JAM	80	MANUFACTURING

2. Create a unique listing of all jobs that are in department 80. Include the location of the department in the output.

QUERY:

```
SELECT DISTINCT JOB_ID, LOCATION_ID  
FROM EMPLOYEES, DEPARTMENT  
WHERE EMPLOYEES.DEPARTMENT_ID = DEPARTMENT.DEPT_ID  
AND EMPLOYEES.DEPARTMENT_ID = 80;
```

OUTPUT:



The screenshot shows the Oracle SQL Workshop interface. The top navigation bar includes APEX, App Builder, SQL Workshop, Team Development, Gallery, a search bar, and a user profile for Afrin Fathima. The SQL Workshop tab is selected. Below the toolbar, the schema is set to WKSP_AFRINFATHIMA014. The main area displays the SQL command and its execution results.

SQL Commands

Language: SQL Rows: 100

Clear Command Find Tables Save Run

47 SELECT DISTINCT JOB_ID, LOCATION_ID
48 FROM EMPLOYEES, DEPARTMENT
49 WHERE EMPLOYEES.DEPARTMENT_ID = DEPARTMENT.DEPT_ID
50 AND EMPLOYEES.DEPARTMENT_ID = 80;

Results Explain Describe Saved SQL History

JOB_ID	LOCATION_ID
DESIGNER	6
SUPERVISOR	6
ENGINEER	6

3 rows returned in 0.01 seconds Download

3. Write a query to display the employee last name, department name, location ID, and city of all employees who earn a commission

QUERY:

```
SELECT E.LAST_NAME, D.DEPT_NAME, D.LOCATION_ID, L.CITY  
FROM EMPLOYEES E, DEPARTMENT D, LOCATION L  
WHERE DEPARTMENT_ID = DEPT_ID  
AND D.LOCATION_ID = L.LOCATION_ID  
AND COMMISSION_PCT IS NOT NULL;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop (selected), Team Development, and Gallery. The right side shows the user's profile: 'Afrin Fathima' with the identifier 'AF afrinfathima014'. The main area has tabs for SQL Commands, SQL (selected), Clear Command, Find Tables, Run, Save, and Schema (set to 'WKSP_AFRINFATHIMA014'). The SQL editor contains the query from the previous step. The Results tab is selected, displaying the following data:

LAST_NAME	DEPT_NAME	LOCATION_ID	CITY
KOHLI	MANUFACTURING	6	TORONTO
JOBS	IT	1700	LONDON
RAVI	MARKETING	1	CHENNAI
DEV	MANUFACTURING	6	TORONTO
RAVI	IT	1700	LONDON
JAY	MARKETING	1	CHENNAI
JANE	HR	1700	LONDON
EMANUEL	FINANCE	3	VALHALLA
VIJAY	MANAGEMENT	4	DC
DAVIES	FINANCE	3	VALHALLA
JAM	MANUFACTURING	6	TORONTO

At the bottom, there are footer links for 220701014@rajalakshmi.edu.in, afrinfathima014, and en, along with copyright information: Copyright © 1999, 2023, Oracle and/or its affiliates. and Oracle APEX 23.2.4.

4. Display the employee last name and department name for all employees who have an a(lowercase) in their last names.

QUERY:

```
SELECT LAST_NAME, DEPT_NAME  
FROM EMPLOYEES, DEPARTMENT  
WHERE DEPARTMENT_ID = DEPT_ID  
AND LAST_NAME LIKE '%A%';
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes 'APEX', 'App Builder', 'SQL Workshop' (selected), 'Team Development', and 'Gallery'. The right side shows a user profile for 'Afrin Fathima' (af@fathima014). The main area has tabs for 'SQL Commands' (selected), 'Results' (selected), 'Explain', 'Describe', 'Saved SQL', and 'History'. The SQL command entered is:

```
57  
58 SELECT LAST_NAME, DEPT_NAME  
59 FROM EMPLOYEES, DEPARTMENT  
60 WHERE DEPARTMENT_ID = DEPT_ID  
61 AND LAST_NAME LIKE '%A%';  
62
```

The 'Results' tab displays the output as a table:

LAST_NAME	DEPT_NAME
RAVI	MARKETING
UMA	STOCK
PARTHI	STOCK
RAVI	IT
JAY	MARKETING
HARSH	EXECUTIVE
JANE	HR
EMANUEL	FINANCE
VIJAY	MANAGEMENT
DAVIES	FINANCE
JAM	MANUFACTURING

At the bottom, there are footer links for '220701014@rajalakshmi.edu.in', 'af@fathima014', and 'en'. The copyright notice reads 'Copyright © 1999, 2023, Oracle and/or Its affiliates.' and the version 'Oracle APEX 23.2.4'.

5. Write a query to display the last name, job, department number, and department name for all employees who work in Toronto.

QUERY:

```
SELECT LAST_NAME, JOB_ID, DEPARTMENT_ID, DEPT_NAME  
FROM EMPLOYEES JOIN DEPARTMENT D  
ON (DEPARTMENT_ID = DEPT_ID)  
JOIN LOCATION L  
ON (D.LOCATION_ID = L.LOCATION_ID)  
WHERE LOWER(L.CITY) = 'toronto';
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery, along with a search bar and user profile icons. The main workspace is titled "SQL Commands" and shows the schema "WKSP_AFRINFATHIMA014". The SQL editor contains the query from the previous step, with line numbers 65 through 70. The "Run" button is highlighted in green at the bottom right of the editor. Below the editor, the "Results" tab is selected, displaying a table with four columns: LAST_NAME, JOB_ID, DEPARTMENT_ID, and DEPT_NAME. The table data shows three rows for employees Kohli, Dev, and Jam, all working in the department with ID 80, which is labeled "MANUFACTURING". A tooltip indicates a "14 min to full charge". At the bottom of the page, there are footer links for user information and copyright notice.

LAST_NAME	JOB_ID	DEPARTMENT_ID	DEPT_NAME
KOHLI	SUPERVISOR	80	MANUFACTURING
DEV	ENGINEER	80	MANUFACTURING
JAM	DESIGNER	80	MANUFACTURING

6. Display the employee last name and employee number along with their manager's last name and manager number. Label the columns Employee, Emp#, Manager, and Mgr#, Respectively

QUERY:

```
SELECT W.LAST_NAME "EMPLOYEE", W.EMPLOYEE_ID "EMP#",  
M.LAST_NAME "MANAGER", M.EMPLOYEE_ID "MGR#"  
FROM EMPLOYEES W JOIN EMPLOYEES M  
ON (W.MANAGER_ID = M.EMPLOYEE_ID);
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop (selected), Team Development, and Gallery. On the right, there is a search bar, user profile, and session information for 'Afrin Fatima' (afatima014). The main workspace has tabs for SQL Commands, SQL Workshop (selected), and Team Development. The SQL tab is active, showing the executed SQL query:

```
70  SELECT W.LAST_NAME"EMPLOYEE",W.EMPLOYEE_ID"EMP#",M.LAST_NAME"MANAGER",M.EMPLOYEE_ID"MGR#" FROM EMPLOYEES W JOIN EMPLOYEES M ON (W.MANAGER_ID = M.EMPLOYEE_ID);  
71
```

The Results tab is selected, displaying the query results in a grid format:

EMPLOYEE	EMP#	MANAGER	MGR#
UMA	102	RAVI	101
PARTHI	103	RAVI	101
JAY	100	RAVI	101
SIDDHU	114	HARSH	115
VIJAY	106	HARSH	115
JOBs	111	JANE	104
RAVI	112	JANE	104
DAVIES	110	EMANUEL	105
KOHLI	109	VIJAY	106
RAVI	101	VIJAY	106
JANE	104	VIJAY	106
EMANUEL	105	VIJAY	106
JAM	107	VIJAY	106
DEV	108	JAM	107

Below the grid, it says '14 rows returned in 0.01 seconds'. At the bottom, there are links for 'Download', 'Copyright © 1999, 2023, Oracle and/or its affiliates.', and 'Oracle APEX 23.2.4'.

7. Modify lab4_6.sql to display all employees including King, who has no manager. Order the results by the employee number.

QUERY:

```
SELECT W.LAST_NAME "EMPLOYEE", W.EMPLOYEE_ID "EMP#",  
M.LAST_NAME "MANAGER", M.EMPLOYEE_ID "MGR#" FROM EMPLOYEES W  
LEFT OUTER JOIN EMPLOYEES M ON (W.MANAGER_ID = M.EMPLOYEE_ID)  
ORDER BY W.EMPLOYEE_ID;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop (selected), Team Development, and Gallery. On the right, there's a search bar, a help icon, and a user profile for 'Afrin Fathima' (afrinfathima014). The main workspace is titled 'SQL Commands'. It shows the executed SQL code and its results.

SQL Commands

Schema: WKSP_AFRINFATHIMA014

Language: SQL Rows: 50

72 SELECT W.LAST_NAME "EMPLOYEE", W.EMPLOYEE_ID "EMP#", M.LAST_NAME "MANAGER", M.EMPLOYEE_ID "MGR#" FROM EMPLOYEES W
73 LEFT OUTER JOIN EMPLOYEES M ON (W.MANAGER_ID = M.EMPLOYEE_ID) ORDER BY W.EMPLOYEE_ID;

EMPLOYEE	EMP#	MANAGER	MGR#
JAY	100	RAVI	101
RAVI	101	VIJAY	106
UMA	102	RAVI	101
PARTHI	103	RAVI	101
JANE	104	VIJAY	106
EMANUEL	105	VIJAY	106
VIJAY	106	HARSH	113
JAM	107	VIJAY	106
DEV	108	JAM	107
KOHLI	109	VIJAY	106
DAVIES	110	EMANUEL	105
JOBS	111	JANE	104
RAVI	112	JANE	104
HARSH	113	-	-
SIDDHU	114	HARSH	113

15 rows returned in 0.01 seconds [Download](#)

220701014@rajalakshmi.edu.in afarinfathima014 en Copyright © 1999, 2023, Oracle and/or Its affiliates. Oracle APEX 23.2.4

8. Create a query that displays employee last names, department numbers, and all the employees who work in the same department as a given employee. Give each column an appropriate label.

QUERY:

```
SELECT E.LAST_NAME EMPLOYEE, E.DEPARTMENT_ID DEPARTMENT,
C.LAST_NAME COLLEAGUE
FROM EMPLOYEES E JOIN EMPLOYEES C
ON (E.DEPARTMENT_ID = C.DEPARTMENT_ID)
WHERE E.EMPLOYEE_ID <> C.EMPLOYEE_ID
ORDER BY E.DEPARTMENT_ID, E.LAST_NAME;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for App Builder, SQL Workshop, Team Development, and Gallery. The right side shows a user profile for 'Afrin Fathima' (afrinfathima014). The main area is titled 'SQL Commands' with a schema dropdown set to 'WKSP_AFRINFATHIMA014'. The SQL editor contains the following code:

```
75  SELECT E.LAST_NAME EMPLOYEE, E.DEPARTMENT_ID DEPARTMENT, C.LAST_NAME COLLEAGUE FROM EMPLOYEES E JOIN EMPLOYEES C ON (E.DEPARTMENT_ID = C.DEPARTMENT_ID)
76  WHERE E.EMPLOYEE_ID <> C.EMPLOYEE_ID ORDER BY E.DEPARTMENT_ID, E.LAST_NAME;
```

The results section displays the output of the query:

EMPLOYEE	DEPARTMENT	COLLEAGUE
JAY	10	RAVI
RAVI	10	JAY
PARTHI	20	UMA
UMA	20	PARTHI
DAVIES	30	EMANUEL
EMANUEL	30	DAVIES
HARSH	60	SIDDHU
SIDDHU	60	HARSH
DEV	80	JAM
DEV	80	KOHLI
JAM	80	DEV
JAM	80	KOHLI
KOHLI	80	DEV
KOHLI	80	JAM
JOBS	90	RAVI
RAVI	90	JOBS

At the bottom, it says '16 rows returned in 0.01 seconds' and provides download options. The footer includes copyright information for Oracle and the APEX version.

9. Show the structure of the JOB_GRADES table. Create a query that displays the name, job, department name, salary, and grade for all employees.

QUERY:

DESC JOB_GRADES;

```
SELECT E.LAST_NAME, E.JOB_ID, D.DEPT_NAME, E.SALARY, J.GRADE_LEVEL  
FROM EMPLOYEES E JOIN DEPARTMENT D ON (E.DEPARTMENT_ID =  
D.DEPT_ID) JOIN JOB_GRADES J ON (E.SALARY BETWEEN J.LOWEST_SAL AND  
J.HIGHEST_SAL);
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes 'APEX', 'App Builder', 'SQL Workshop', 'Team Development', 'Gallery', a search bar, and a user profile for 'Afrin Fathima'. The main area is titled 'SQL Commands' with a schema dropdown set to 'WKSP_AFRINFATHIMA014'. The SQL editor contains the command 'DESC JOB_GRADES;'. Below the editor, tabs for 'Results', 'Explain', 'Describe' (which is selected), 'Saved SQL', and 'History' are visible. The results section shows the structure of the 'JOB_GRADES' table with three columns: GRADE_LEVEL (VARCHAR2(2)), LOWEST_SAL (NUMBER(22)), and HIGHEST_SAL (NUMBER(22)). The primary key is GRADE_LEVEL, and both LOWEST_SAL and HIGHEST_SAL are nullable.

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes 'APEX', 'App Builder', 'SQL Workshop', 'Team Development', 'Gallery', a search bar, and a user profile for 'Afrin Fathima'. The main area is titled 'SQL Commands' with a schema dropdown set to 'WKSP_AFRINFATHIMA014'. The SQL editor contains the query: 'SELECT E.LAST_NAME, E.JOB_ID, D.DEPT_NAME, E.SALARY, J.GRADE_LEVEL FROM EMPLOYEES E JOIN DEPARTMENT D ON (E.DEPARTMENT_ID = D.DEPT_ID) JOIN JOB_GRADES J ON (E.SALARY BETWEEN J.LOWEST_SAL AND J.HIGHEST_SAL);'. Below the editor, tabs for 'Results' (selected), 'Explain', 'Describe', 'Saved SQL', and 'History' are visible. The results section displays a grid of employee information across five columns: LAST_NAME, JOB_ID, DEPT_NAME, SALARY, and GRADE_LEVEL. The data includes rows for UMA, PARTHI, VIJAY, JAY, RAVI, SIDDHU, JAM, DEV, KOHLI, EMANUEL, DAVIES, JANE, RAVI, and JOBS. The results indicate 14 rows returned in 0.04 seconds.

10. Create a query to display the name and hire date of any employee hired after employee Davies.

QUERY:

```
SELECT E.LAST_NAME, E.HIRE_DATE FROM EMPLOYEES E JOIN EMPLOYEES  
DAVIES ON (DAVIES.LAST_NAME = 'DAVIES') WHERE DAVIES.HIRE_DATE <  
E.HIRE_DATE;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery, along with a search bar and user profile information for 'Afrin Fathima'.

The SQL Workshop interface has tabs for SQL Commands, Results, Explain, Describe, Saved SQL, and History. The current tab is 'Results'.

The SQL command entered is:

```
89  SELECT E.LAST_NAME, E.HIRE_DATE FROM EMPLOYEES E JOIN EMPLOYEES DAVIES  
90  ON (DAVIES.LAST_NAME = 'DAVIES') WHERE DAVIES.HIRE_DATE < E.HIRE_DATE;  
91
```

The results table displays two rows:

LAST_NAME	HIRE_DATE
RAVI	01/07/2004
DEV	10/12/2004

Below the results, it says '2 rows returned in 0.00 seconds' and provides download options.

At the bottom, there are footer links for 220701014@rajalakshmi.edu.in, afrinfathima014, and Oracle APEX 23.2.4.

11. Display the names and hire dates for all employees who were hired before their managers, along with their manager's names and hire dates. Label the columns Employee, Emp Hired, Manager, and Mgr Hired, respectively.

QUERY:

```
SELECT W.LAST_NAME EMPLOYEE, W.HIRE_DATE "EMP HIRED", M.LAST_NAME  
MANAGER, M.HIRE_DATE "MGR HIRED" FROM EMPLOYEES W JOIN EMPLOYEES  
M ON (W.MANAGER_ID = M.EMPLOYEE_ID) WHERE W.HIRE_DATE <  
M.HIRE_DATE;
```

OUTPUT :

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery. On the right, there is a search bar, a user profile for 'Afrin Fathima' (afarinfathima014), and a schema dropdown set to 'WKSP_AFRINFATHIMA014'. The main workspace displays the SQL command and its results.

SQL Commands:

```
98  SELECT W.LAST_NAME EMPLOYEE, W.HIRE_DATE "EMP HIRED", M.LAST_NAME MANAGER, M.HIRE_DATE "MGR HIRED"  
99  FROM EMPLOYEES W JOIN EMPLOYEES M ON (W.MANAGER_ID = M.EMPLOYEE_ID) WHERE W.HIRE_DATE < M.HIRE_DATE;
```

Results:

EMPLOYEE	EMP HIRED	MANAGER	MGR HIRED
UMA	08/02/1999	RAVI	01/07/2004
PARTHI	04/12/1998	RAVI	01/07/2004
JAY	05/01/1999	RAVI	01/07/2004
JOBS	06/13/1995	JANE	03/15/1998
RAVI	02/15/1996	JANE	03/15/1998
KOHLI	11/05/1988	VIJAY	11/30/2000
JANE	03/15/1998	VIJAY	11/30/2000
EMANUEL	05/01/1994	VIJAY	11/30/2000

8 rows returned in 0.02 seconds [Download](#)

Copyright © 1999, 2023, Oracle and/or its affiliates. Oracle APEX 23.2.4

Evaluation Procedure	Marks awarded
Query(5)	
Execution (5)	
Viva(5)	
Total (15)	
Faculty Signature	

RESULT :

AGGREGATING DATA USING GROUP FUNCTIONS

EX-NO : 8

DATE:

Group functions work across many rows to produce one result per group.

True/False

TRUE

2. Group functions include nulls in calculations.

True/False

FALSE

3. The WHERE clause restricts rows prior to inclusion in a group calculation.

True/False

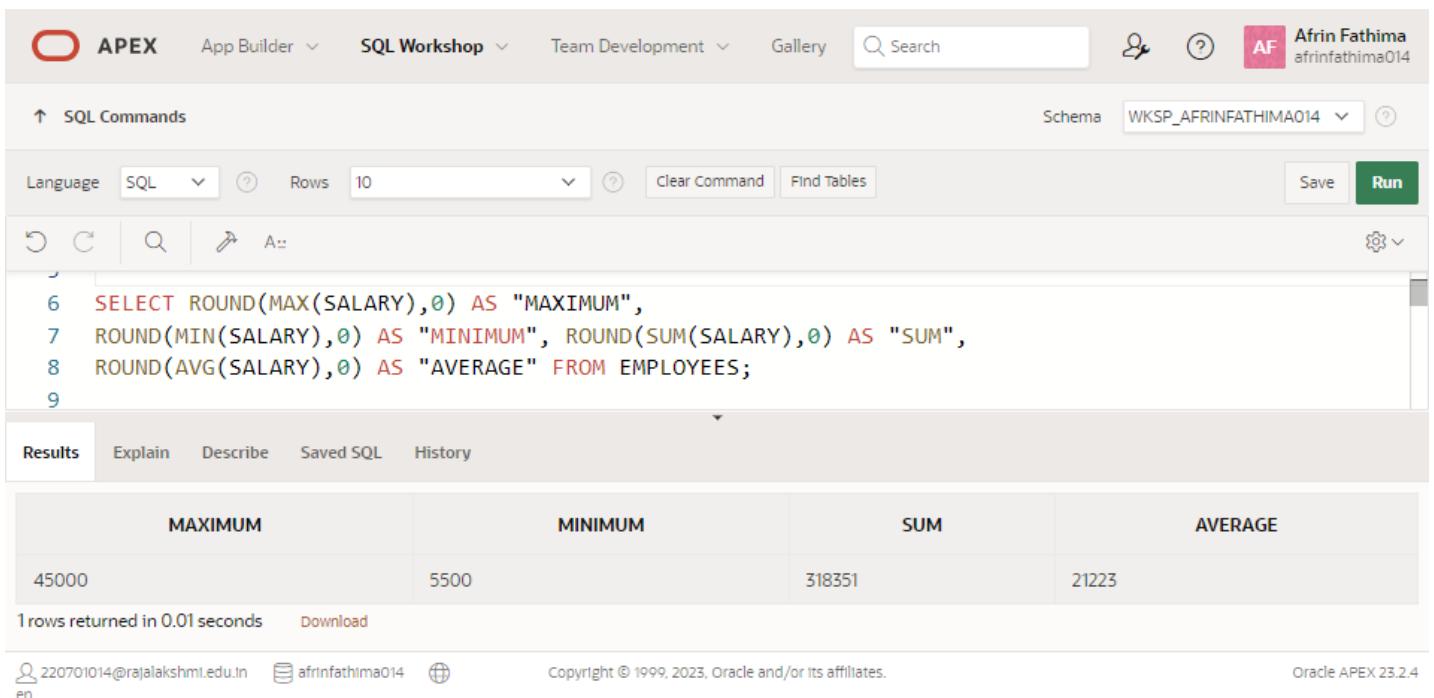
FALSE

4. Find the highest, lowest, sum, and average salary of all employees. Label the columns Maximum, Minimum, Sum, and Average, respectively. Round your results to the nearest whole number.

QUERY:

```
SELECT ROUND(MAX(SALARY),0) AS "MAXIMUM",
ROUND(MIN(SALARY),0) AS "MINIMUM", ROUND(SUM(SALARY),0) AS "SUM",
ROUND(AVG(SALARY),0) AS "AVERAGE" FROM EMPLOYEES;
```

OUTPUT:



The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery, along with a search bar and user profile information for 'Afrin Fathima'.

The SQL Workshop page displays the following details:

- Schema:** WKSP_AFRINFATHIMA014
- Language:** SQL
- Rows:** 10
- Buttons:** Clear Command, Find Tables, Save, Run

The SQL command entered is:

```
6  SELECT ROUND(MAX(SALARY),0) AS "MAXIMUM",
7  ROUND(MIN(SALARY),0) AS "MINIMUM", ROUND(SUM(SALARY),0) AS "SUM",
8  ROUND(AVG(SALARY),0) AS "AVERAGE" FROM EMPLOYEES;
9
```

The results section shows the output of the query:

MAXIMUM	MINIMUM	SUM	AVERAGE
45000	5500	318351	21223

Below the results, it says '1 rows returned in 0.01 seconds' and provides a 'Download' link.

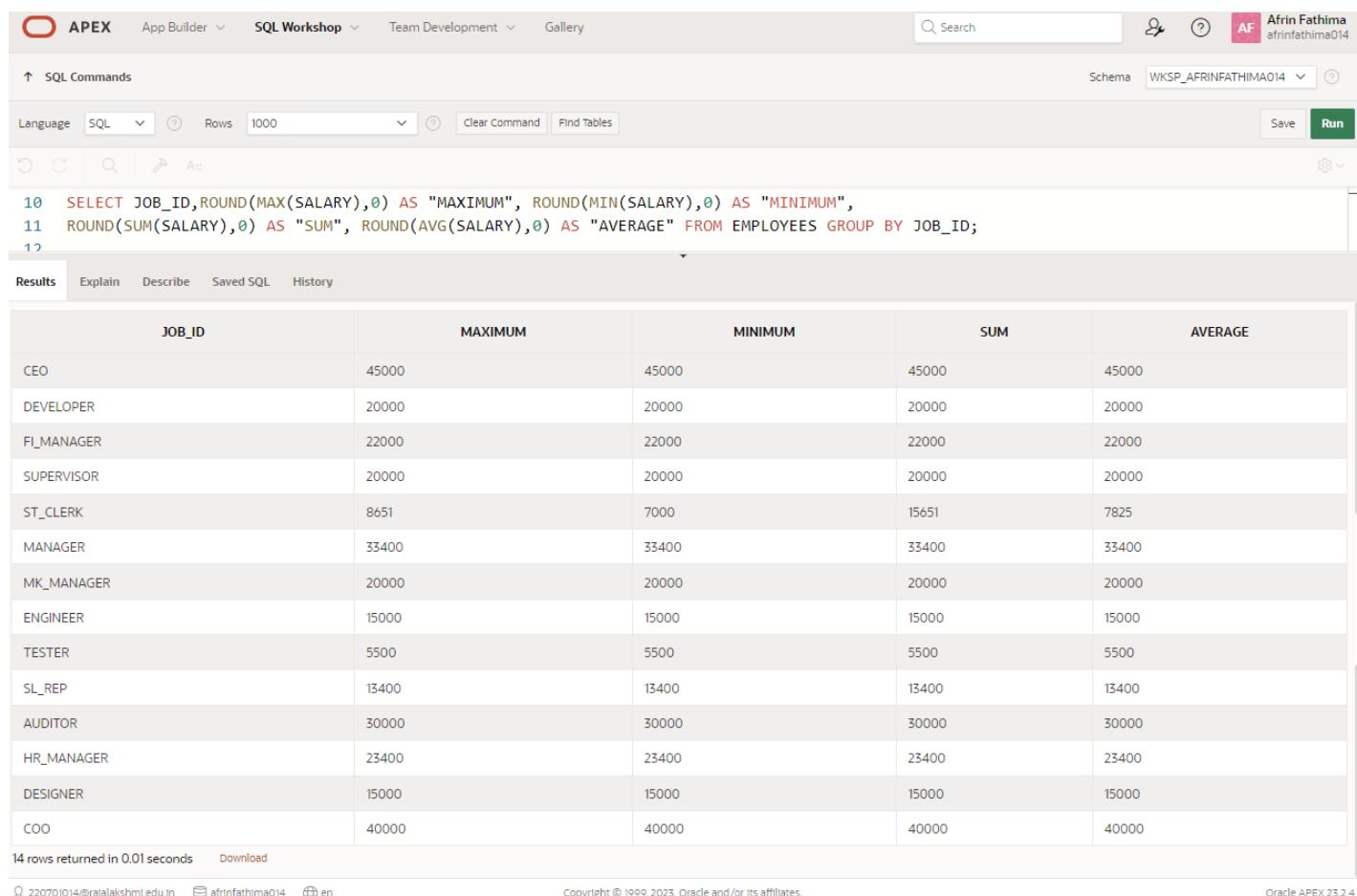
The footer includes links for user profiles (220701014@rajalakshmi.edu.in, afrinfathima014), a help icon, copyright information (Copyright © 1999, 2023, Oracle and/or its affiliates.), and the text 'Oracle APEX 23.2.4'.

5. Modify the above query to display the minimum, maximum, sum, and average salary for each job type.

QUERY:

```
SELECT JOB_ID,ROUND(MAX(SALARY),0) AS "MAXIMUM",
ROUND(MIN(SALARY),0) AS "MINIMUM",
ROUND(SUM(SALARY),0) AS "SUM", ROUND(AVG(SALARY),0) AS "AVERAGE"
FROM EMPLOYEES GROUP BY JOB_ID;
```

OUTPUT:



The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery. The right side shows the user profile 'Afrin Fathima' with the ID 'afarinfathima014'. The main area has tabs for SQL Commands, Results, Explain, Describe, Saved SQL, and History. The SQL tab is active, displaying the executed SQL code:

```
10  SELECT JOB_ID,ROUND(MAX(SALARY),0) AS "MAXIMUM", ROUND(MIN(SALARY),0) AS "MINIMUM",
11    ROUND(SUM(SALARY),0) AS "SUM", ROUND(AVG(SALARY),0) AS "AVERAGE" FROM EMPLOYEES GROUP BY JOB_ID;
12
```

The Results tab displays the query results in a table format:

JOB_ID	MAXIMUM	MINIMUM	SUM	AVERAGE
CEO	45000	45000	45000	45000
DEVELOPER	20000	20000	20000	20000
FI_MANAGER	22000	22000	22000	22000
SUPERVISOR	20000	20000	20000	20000
ST_CLERK	8651	7000	15651	7825
MANAGER	33400	33400	33400	33400
MK_MANAGER	20000	20000	20000	20000
ENGINEER	15000	15000	15000	15000
TESTER	5500	5500	5500	5500
SL_REP	13400	13400	13400	13400
AUDITOR	30000	30000	30000	30000
HR_MANAGER	23400	23400	23400	23400
DESIGNER	15000	15000	15000	15000
COO	40000	40000	40000	40000

At the bottom, it says '14 rows returned in 0.01 seconds' and provides download options. The footer includes copyright information for Oracle and the APEX version.

6. Write a query to display the number of people with the same job. Generalize the query so that the user in the HR department is prompted for a job title.

QUERY:

```
SELECT JOB_ID,COUNT(JOB_ID) FROM EMPLOYEES  
GROUP BY JOB_ID;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery. On the right, there's a search bar, user profile, and session information for 'Afrin Fathima' (afirinfathima014). The main workspace has tabs for SQL Commands, SQL (selected), Clear Command, Find Tables, and Run. The SQL editor contains the following code:

```
14  SELECT JOB_ID,COUNT(JOB_ID) FROM EMPLOYEES  
15  GROUP BY JOB_ID;  
16
```

The Results tab displays the output of the query:

JOB_ID	COUNT(JOB_ID)
CEO	1
DEVELOPER	1
FI_MANAGER	1
SUPERVISOR	1
ST_CLERK	2
MANAGER	1
MK_MANAGER	1
ENGINEER	1
TESTER	1
SL_REP	1
AUDITOR	1
HR_MANAGER	1
DESIGNER	1
COO	1

At the bottom, it says '14 rows returned in 0.00 seconds' and provides download and history options. The footer includes copyright information and the version 'Oracle APEX 23.2.4'.

7. Determine the number of managers without listing them. Label the column Number of Managers. Hint: Use the MANAGER_ID column to determine the number of managers.

QUERY:

```
SELECT COUNT(DISTINCT MANAGER_ID) "NUMBER OF MANAGERS"  
FROM EMPLOYEES;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery, along with a search bar and user profile information for 'Afrin Fathima'.

In the SQL Workshop section, the schema is set to 'WKSP_AFRINFATHIMA014'. The query entered is:

```
17  SELECT COUNT(DISTINCT MANAGER_ID) "NUMBER OF MANAGERS"  
18  FROM EMPLOYEES;
```

The results tab shows the output:

NUMBER OF MANAGERS
6

Below the results, it says '1 rows returned in 0.01 seconds'. The bottom of the page includes copyright information for Oracle and the APEX version 'Oracle APEX 23.2.4'.

8. Find the difference between the highest and lowest salaries. Label the column DIFFERENCE.

QUERY:

```
SELECT MAX(SALARY)-MIN(SALARY) AS "DIFFERENCE" FROM EMPLOYEES;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery, along with a search bar and user profile information for 'Afrin Fathima' (afrafathima014). The main workspace is titled 'SQL Commands'. The schema is set to 'WKSP_AFRINFATHIMA014'. The query entered is:

```
19
20  SELECT MAX(SALARY)-MIN(SALARY) AS "DIFFERENCE" FROM EMPLOYEES;
21
```

The results tab is selected, showing the output:

DIFFERENCE
39500

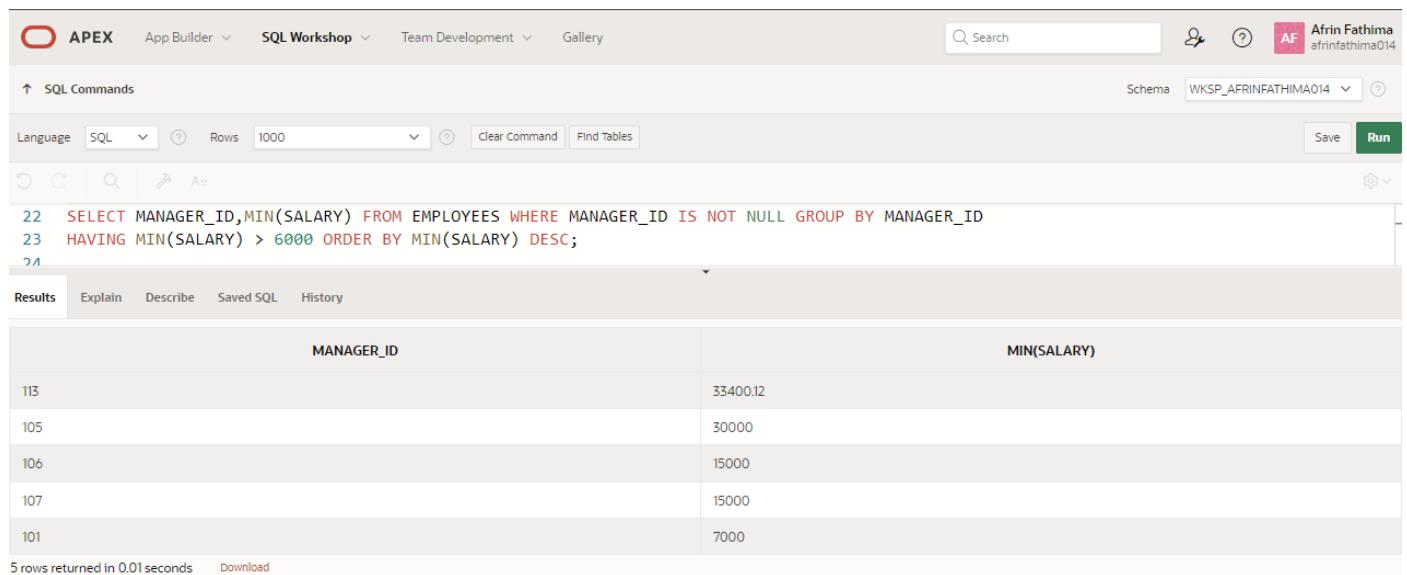
Below the results, it says '1 rows returned in 0.00 seconds' and provides download options. The bottom footer includes copyright information for Oracle and the APEX version 'Oracle APEX 23.2.4'.

9. Create a report to display the manager number and the salary of the lowest-paid employee for that manager. Exclude anyone whose manager is not known. Exclude any groups where the minimum salary is \$6,000 or less. Sort the output in descending order of salary.

QUERY:

```
SELECT MANAGER_ID,MIN(SALARY) FROM EMPLOYEES WHERE  
MANAGER_ID IS NOT NULL GROUP BY MANAGER_ID  
HAVING MIN(SALARY) > 6000 ORDER BY MIN(SALARY) DESC;
```

OUTPUT:



The screenshot shows the Oracle SQL Workshop interface. The top navigation bar includes APEX, App Builder, SQL Workshop (selected), Team Development, and Gallery. On the right, there's a search bar, a user profile for Afrin Fathima (afrafathima014), and a toolbar with Save and Run buttons. The main area shows the SQL command entered:

```
22  SELECT MANAGER_ID,MIN(SALARY) FROM EMPLOYEES WHERE MANAGER_ID IS NOT NULL GROUP BY MANAGER_ID  
23  HAVING MIN(SALARY) > 6000 ORDER BY MIN(SALARY) DESC;  
24
```

The Results tab is selected, displaying the query results in a table:

MANAGER_ID	MIN(SALARY)
113	33400.12
105	30000
106	15000
107	15000
101	7000

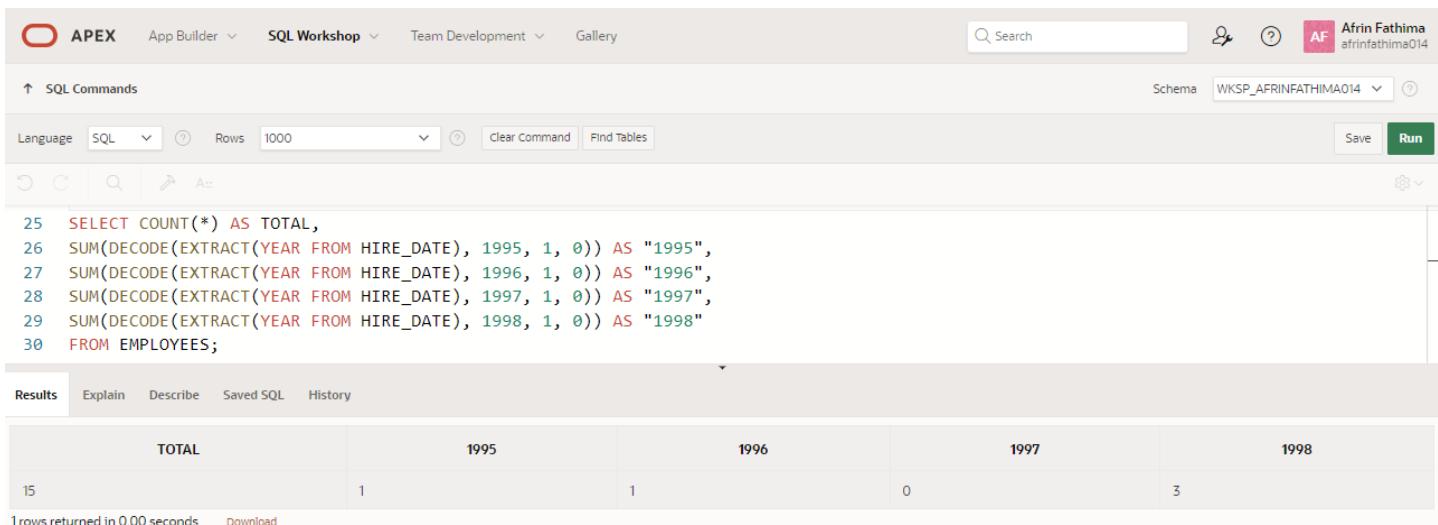
Below the table, it says "5 rows returned in 0.01 seconds" and has a Download button.

10. Create a query to display the total number of employees and, of that total, the number of employees hired in 1995, 1996, 1997, and 1998. Create appropriate column headings.

QUERY:

```
SELECT COUNT(*) AS TOTAL,  
SUM(DECODE(EXTRACT(YEAR FROM HIRE_DATE), 1995, 1, 0)) AS "1995",  
SUM(DECODE(EXTRACT(YEAR FROM HIRE_DATE), 1996, 1, 0)) AS "1996",  
SUM(DECODE(EXTRACT(YEAR FROM HIRE_DATE), 1997, 1, 0)) AS "1997",  
SUM(DECODE(EXTRACT(YEAR FROM HIRE_DATE), 1998, 1, 0)) AS "1998"  
FROM EMPLOYEES;
```

OUTPUT:



The screenshot shows the Oracle SQL Workshop interface. The top navigation bar includes APEX, App Builder, SQL Workshop, Team Development, and Gallery. On the right, there's a search bar, user information for 'Afrin Fathima', and a schema dropdown set to 'WKSP_AFRINFATHIMA014'. The main area is titled 'SQL Commands' with tabs for Language (set to SQL), Rows (set to 1000), Clear Command, Find Tables, Save, and Run. Below this is a toolbar with icons for Undo, Redo, Search, and Run. The SQL editor contains the query from the previous step, numbered 25 to 30. The results tab is selected, showing a table with five columns: TOTAL, 1995, 1996, 1997, and 1998. The data row shows values 15, 1, 1, 0, and 3 respectively. At the bottom, it says '1 rows returned in 0.00 seconds' and has a 'Download' link.

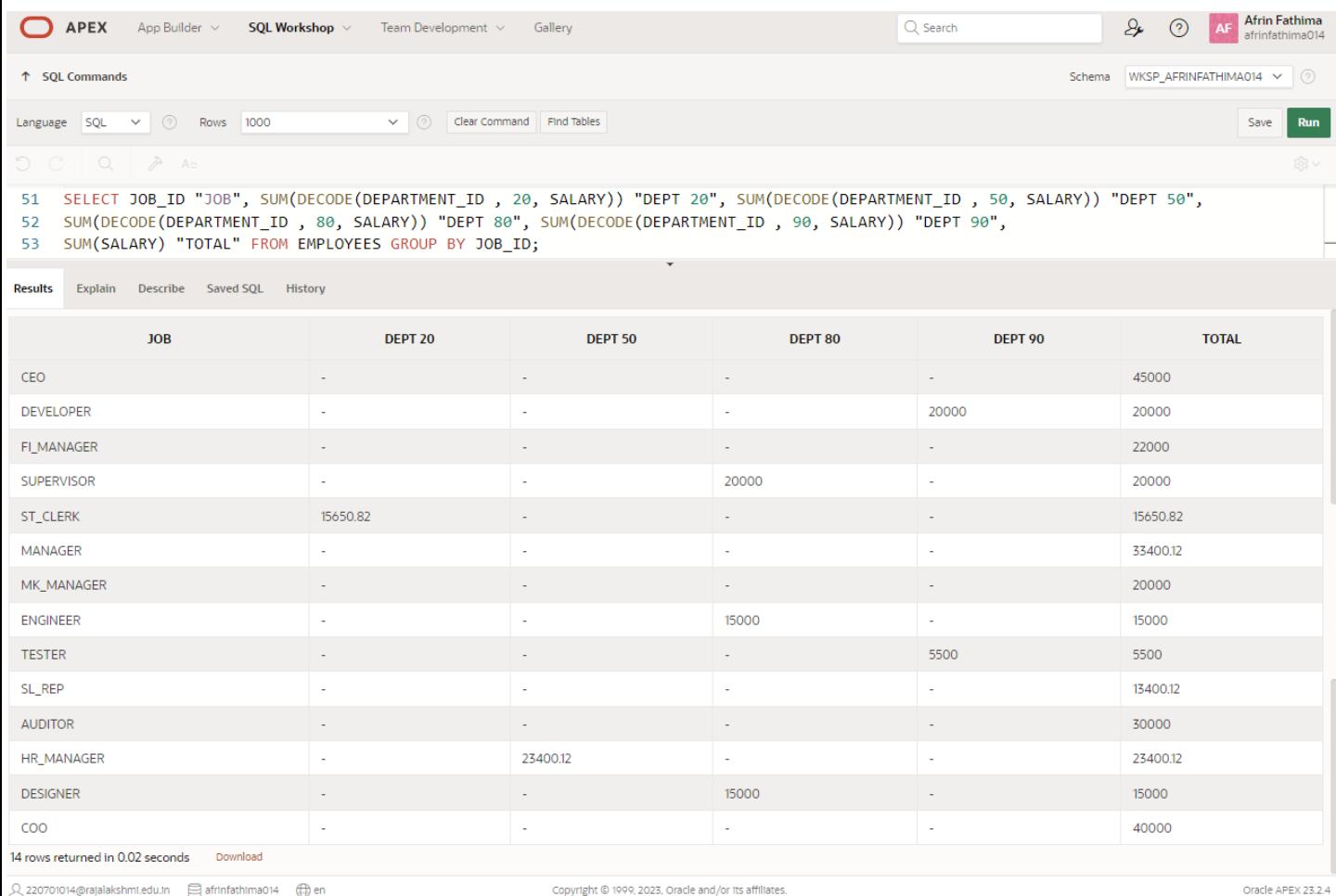
TOTAL	1995	1996	1997	1998
15	1	1	0	3

11. Create a matrix query to display the job, the salary for that job based on department number, and the total salary for that job, for departments 20, 50, 80, and 90, giving each column an appropriate heading.

QUERY:

```
SELECT JOB_ID "JOB", SUM(DECODE(DEPARTMENT_ID , 20, SALARY)) "DEPT 20", SUM(DECODE(DEPARTMENT_ID , 50, SALARY)) "DEPT 50",  
SUM(DECODE(DEPARTMENT_ID , 80, SALARY)) "DEPT 80",  
SUM(DECODE(DEPARTMENT_ID , 90, SALARY)) "DEPT 90",  
SUM(SALARY) "TOTAL" FROM EMPLOYEES GROUP BY JOB_ID;
```

OUTPUT:



The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery. The user is identified as 'Afrin Fathima' (afrafathima014). The SQL tab is selected, and the query is displayed in the editor:

```
51  SELECT JOB_ID "JOB", SUM(DECODE(DEPARTMENT_ID , 20, SALARY)) "DEPT 20", SUM(DECODE(DEPARTMENT_ID , 50, SALARY)) "DEPT 50",  
52  SUM(DECODE(DEPARTMENT_ID , 80, SALARY)) "DEPT 80", SUM(DECODE(DEPARTMENT_ID , 90, SALARY)) "DEPT 90",  
53  SUM(SALARY) "TOTAL" FROM EMPLOYEES GROUP BY JOB_ID;
```

The results are presented in a matrix format:

JOB	DEPT 20	DEPT 50	DEPT 80	DEPT 90	TOTAL
CEO	-	-	-	-	45000
DEVELOPER	-	-	-	20000	20000
FI_MANAGER	-	-	-	-	22000
SUPERVISOR	-	-	20000	-	20000
ST_CLERK	15650.82	-	-	-	15650.82
MANAGER	-	-	-	-	33400.12
MK_MANAGER	-	-	-	-	20000
ENGINEER	-	-	15000	-	15000
TESTER	-	-	-	5500	5500
SL_REP	-	-	-	-	13400.12
AUDITOR	-	-	-	-	30000
HR_MANAGER	-	23400.12	-	-	23400.12
DESIGNER	-	-	15000	-	15000
COO	-	-	-	-	40000

14 rows returned in 0.02 seconds [Download](#)

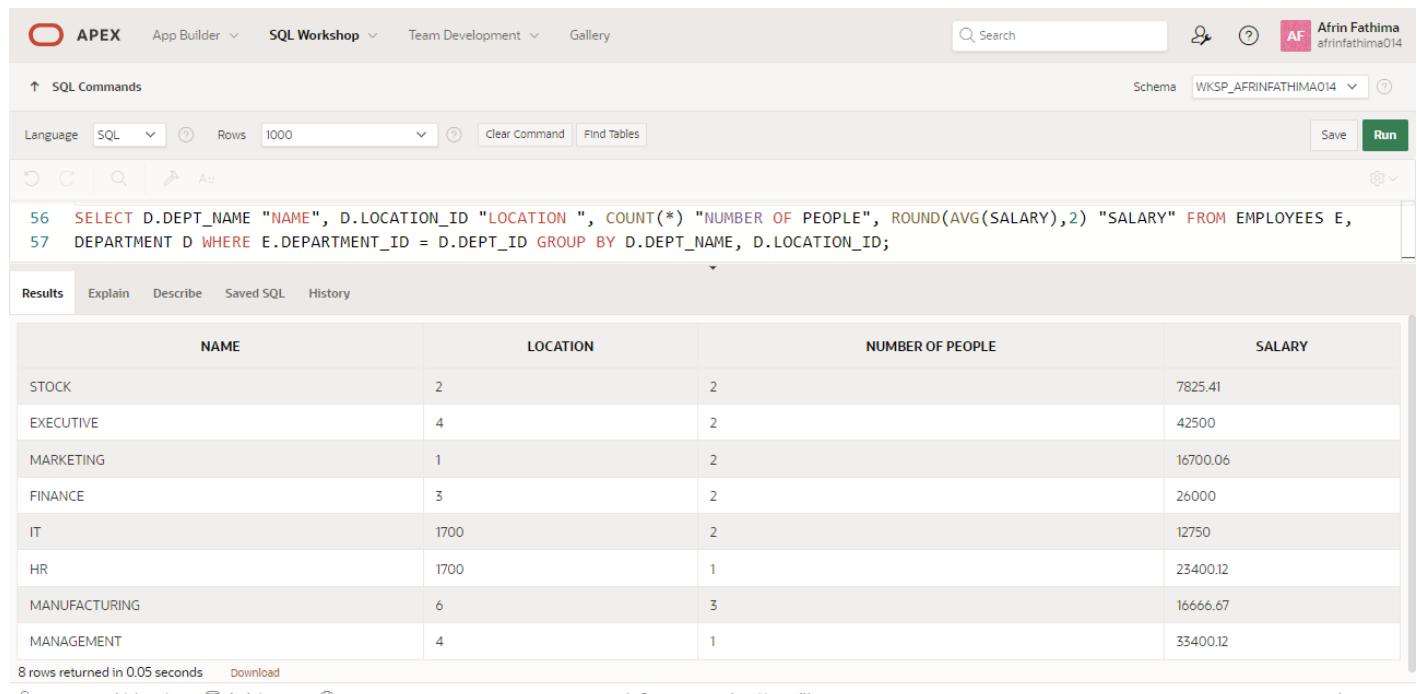
220701014@rajalakshmi.edu.in afrinfathima014 en Copyright © 1999, 2023, Oracle and/or its affiliates. Oracle APEX 23.2.4

12. Write a query to display each department's name, location, number of employees, and the average salary for all the employees in that department. Label the column name-Location, Number of people, and salary respectively. Round the average salary to two decimal places.

QUERY:

```
SELECT D.DEPT_NAME "NAME", D.LOCATION_ID "LOCATION ", COUNT(*)  
"NUMBER OF PEOPLE", ROUND(AVG(SALARY),2) "SALARY" FROM EMPLOYEES  
E, DEPARTMENT D WHERE E.DEPARTMENT_ID = D.DEPT_ID GROUP BY  
D.DEPT_NAME, D.LOCATION_ID;
```

OUTPUT:



The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery. The right side shows a user profile for 'Afrin Fathima' (afrinfathima014). The main area is titled 'SQL Commands' with a search bar and a schema dropdown set to 'WKSP_AFRINFATHIMA014'. Below the toolbar are buttons for Language (SQL), Rows (1000), Clear Command, Find Tables, Save, and Run. The SQL editor contains the following code:

```
56  SELECT D.DEPT_NAME "NAME", D.LOCATION_ID "LOCATION ", COUNT(*) "NUMBER OF PEOPLE", ROUND(AVG(SALARY),2) "SALARY" FROM EMPLOYEES E,  
57  DEPARTMENT D WHERE E.DEPARTMENT_ID = D.DEPT_ID GROUP BY D.DEPT_NAME, D.LOCATION_ID;
```

The results tab is selected, displaying the query results in a grid format:

NAME	LOCATION	NUMBER OF PEOPLE	SALARY
STOCK	2	2	7825.41
EXECUTIVE	4	2	42500
MARKETING	1	2	16700.06
FINANCE	3	2	26000
IT	1700	2	12750
HR	1700	1	23400.12
MANUFACTURING	6	3	16666.67
MANAGEMENT	4	1	33400.12

Below the results, it says '8 rows returned in 0.05 seconds' and provides download options. The bottom footer includes copyright information for Oracle and the APEX version.

Evaluation Procedure	Marks awarded
Query(5)	
Execution (5)	
Viva(5)	
Total (15)	
Faculty Signature	

RESULT :

SUB-QUERIES

EX-NO : 9

DATE:

1. The HR department needs a query that prompts the user for an employee last name. The query then displays the last name and hire date of any employee in the same department as the employee whose name they supply (excluding that employee). For example, if the user enters Zlotkey, find all employees who work with Zlotkey (excluding Zlotkey).

QUERY:

```
SELECT LAST_NAME, TO_CHAR(HIRE_DATE,'DD-MON-YYYY') AS "HIRE_DATE"
FROM EMPLOYEES A JOIN (SELECT DEPARTMENT_ID FROM EMPLOYEES
WHERE LAST_NAME = :SURNAME) B
ON A.DEPARTMENT_ID = B.DEPARTMENT_ID AND LAST_NAME <> :SURNAME;
```

OUTPUT:



The screenshot shows the Oracle SQL Workshop interface. The SQL command window contains the following code:

```
7
8  SELECT LAST_NAME, TO_CHAR(HIRE_DATE,'DD-MON-YYYY') AS "HIRE_DATE" FROM EMPLOYEES A JOIN (SELECT DEPARTMENT_ID FROM EMPLOYEES WHERE LAST_NAME = :SURNAME)
9  B ON A.DEPARTMENT_ID = B.DEPARTMENT_ID AND LAST_NAME <> :SURNAME;
10
```

The results window shows the output of the query:

LAST_NAME	HIRE_DATE
DEV	12-OCT-2004
JAM	04-DEC-2000

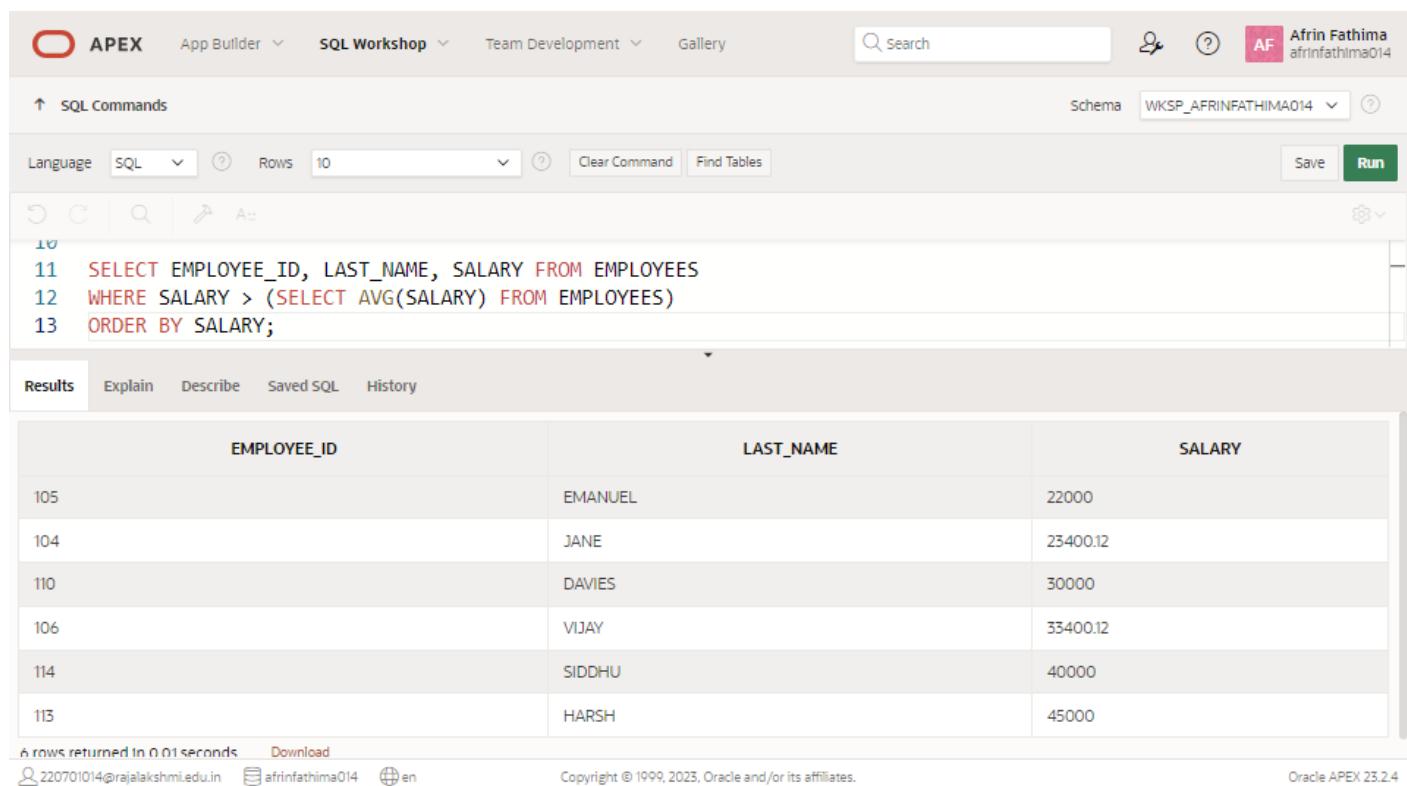
2 rows returned in 0.01 seconds [Download](#)

2. Create a report that displays the employee number, last name, and salary of all employees who earn more than the average salary. Sort the results in order of ascending salary.

QUERY:

```
SELECT EMPLOYEE_ID, LAST_NAME, SALARY FROM EMPLOYEES  
WHERE SALARY > (SELECT AVG(SALARY) FROM EMPLOYEES)  
ORDER BY SALARY;
```

OUTPUT:



The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop (selected), Team Development, and Gallery. A search bar and user profile are also present. The main area shows the SQL command entered:

```
11  SELECT EMPLOYEE_ID, LAST_NAME, SALARY FROM EMPLOYEES  
12  WHERE SALARY > (SELECT AVG(SALARY) FROM EMPLOYEES)  
13  ORDER BY SALARY;
```

The Results tab is selected, displaying the query results in a grid:

EMPLOYEE_ID	LAST_NAME	SALARY
105	EMANUEL	22000
104	JANE	23400.12
110	DAVIES	30000
106	VIJAY	33400.12
114	SIDDHU	40000
113	HARSH	45000

Below the results, it says "6 rows returned in 0.01 seconds". The bottom footer includes copyright information and the version "Oracle APEX 23.2.4".

3. Write a query that displays the employee number and last name of all employees who work in a department with any employee whose last name contains a u.

QUERY:

```
SELECT EMPLOYEE_ID, LAST_NAME FROM EMPLOYEES A JOIN (SELECT DEPARTMENT_ID FROM EMPLOYEES WHERE LAST_NAME LIKE '%U%') B ON A.DEPARTMENT_ID = B.DEPARTMENT_ID;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop (selected), Team Development, and Gallery. A search bar and user profile are also present. The main workspace shows the SQL command being run:

```
15  SELECT EMPLOYEE_ID, LAST_NAME FROM EMPLOYEES A JOIN (SELECT DEPARTMENT_ID FROM EMPLOYEES
16  WHERE LAST_NAME LIKE '%U%') B
17  ON A.DEPARTMENT_ID = B.DEPARTMENT_ID;
```

The results tab is selected, displaying the output of the query:

EMPLOYEE_ID	LAST_NAME
102	UMA
103	PARTHI
113	HARSH
114	SIDDHU
105	EMANUEL
110	DAVIES

Below the table, it says "6 rows returned In 0.00 seconds". The bottom of the page includes user information (email, session ID, language) and copyright information.

4. The HR department needs a report that displays the last name, department number, and job ID of all employees whose department location ID is 1700.

QUERY:

```
SELECT LAST_NAME, DEPARTMENT_ID, JOB_ID FROM EMPLOYEES A  
JOIN (SELECT DEPT_ID FROM DEPARTMENT WHERE LOCATION_ID=1700) B  
ON A.DEPARTMENT_ID=B.DEPT_ID;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop (which is selected), Team Development, and Gallery. There is also a search bar and a user profile for 'Afrin Fathima'.

The main workspace shows the SQL command being run:

```
23  
24  SELECT LAST_NAME, DEPARTMENT_ID, JOB_ID FROM EMPLOYEES A  
25  JOIN (SELECT DEPT_ID FROM DEPARTMENT WHERE LOCATION_ID=1700) B  
26  ON A.DEPARTMENT_ID=B.DEPT_ID;  
27
```

The 'Results' tab is selected, displaying the query results in a grid:

LAST_NAME	DEPARTMENT_ID	JOB_ID
JOBS	90	DEVELOPER
RAVI	90	TESTER
JANE	50	HR_MANAGER

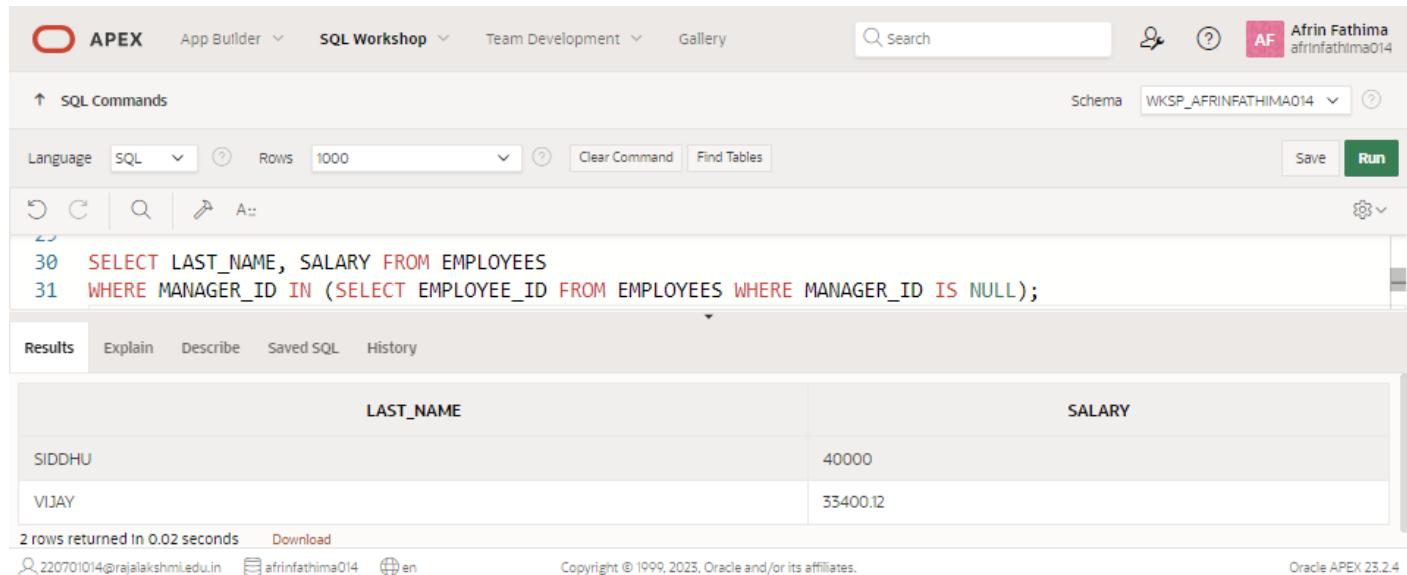
At the bottom, there are footer links for user information (220701014@rajalakshmi.edu.in, afrinfathima014, en), copyright notice (Copyright © 1999, 2023, Oracle and/or its affiliates.), and the software version (Oracle APEX 23.2.4).

5. Create a report for HR that displays the last name and salary of every employee who reports to King.

QUERY:

```
SELECT LAST_NAME, SALARY FROM EMPLOYEES  
WHERE MANAGER_ID IN (SELECT EMPLOYEE_ID FROM EMPLOYEES WHERE  
MANAGER_ID IS NULL);
```

OUTPUT:



The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for App Builder, SQL Workshop (selected), Team Development, and Gallery. A search bar and user profile 'Afrin Fathima' are also present. The main area is titled 'SQL Commands' and contains the following SQL code:

```
30: SELECT LAST_NAME, SALARY FROM EMPLOYEES  
31: WHERE MANAGER_ID IN (SELECT EMPLOYEE_ID FROM EMPLOYEES WHERE MANAGER_ID IS NULL);
```

The 'Results' tab is selected, displaying the output in a grid format:

LAST_NAME	SALARY
SIDDHU	40000
VIJAY	53400.12

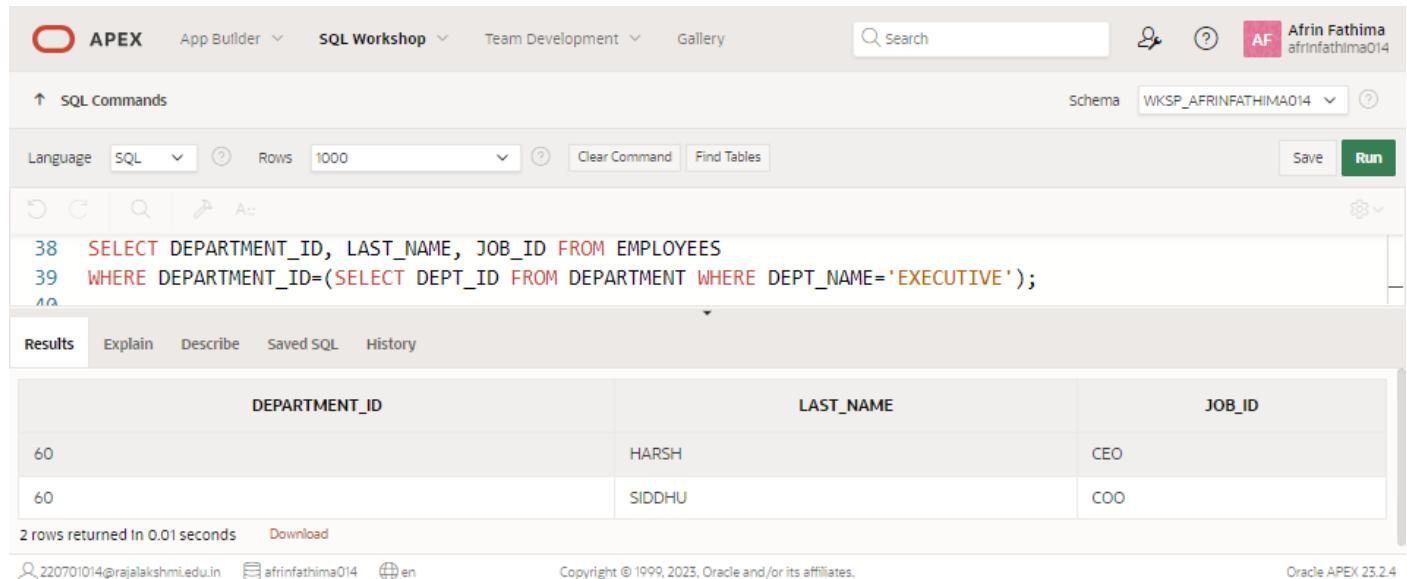
Below the results, a message indicates '2 rows returned in 0.02 seconds'. The bottom of the page shows copyright information for Oracle and the APEX version 'Oracle APEX 23.2.4'.

6. Create a report for HR that displays the department number, last name, and job ID for every employee in the Executive department.

QUERY:

```
SELECT DEPARTMENT_ID, LAST_NAME, JOB_ID FROM EMPLOYEES  
WHERE DEPARTMENT_ID=(SELECT DEPT_ID FROM DEPARTMENT WHERE  
DEPT_NAME='EXECUTIVE');
```

OUTPUT:



The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop (which is selected), Team Development, and Gallery. There is also a search bar and a user profile for 'Afrin Fathima'. The main workspace is titled 'SQL Commands' and contains the following SQL code:

```
38: SELECT DEPARTMENT_ID, LAST_NAME, JOB_ID FROM EMPLOYEES  
39: WHERE DEPARTMENT_ID=(SELECT DEPT_ID FROM DEPARTMENT WHERE DEPT_NAME='EXECUTIVE');
```

The 'Results' tab is selected, displaying the query results in a grid format:

DEPARTMENT_ID	LAST_NAME	JOB_ID
60	HARSH	CEO
60	SIDDHU	COO

Below the results, it says '2 rows returned in 0.01 seconds'. At the bottom, there are footer links for email, user profile, and language selection, along with copyright information and the version 'Oracle APEX 23.2.4'.

7. Modify the query 3 to display the employee number, last name, and salary of all employees who earn more than the average salary and who work in a department with any employee whose last name contains a u.

QUERY:

```
SELECT EMPLOYEE_ID, LAST_NAME, SALARY FROM EMPLOYEES  
WHERE SALARY > (SELECT AVG(SALARY) FROM EMPLOYEES)  
AND DEPARTMENT_ID IN (SELECT DEPARTMENT_ID FROM EMPLOYEES  
WHERE LAST_NAME LIKE '%U%');
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery. A search bar and user profile information ('Afrin Fathima' and 'WKSP_AFRINFATHIMA014') are also present. The main workspace is titled 'SQL Commands' and contains the following SQL code:

```
42  SELECT EMPLOYEE_ID, LAST_NAME, SALARY FROM EMPLOYEES  
43  WHERE SALARY > (SELECT AVG(SALARY) FROM EMPLOYEES)  
44  AND DEPARTMENT_ID IN (SELECT DEPARTMENT_ID FROM EMPLOYEES WHERE LAST_NAME LIKE '%U%');  
45  
46
```

Below the code, the 'Results' tab is selected, displaying the query results in a table:

EMPLOYEE_ID	LAST_NAME	SALARY
113	HARSH	45000
114	SIDDHU	40000
105	EMANUEL	22000
110	DAVIES	30000

At the bottom of the results pane, it says '4 rows returned In 0.00 seconds' and provides a 'Download' link. The footer of the page includes copyright information for Oracle and links for user support and documentation.

Evaluation Procedure	Marks awarded
Query(5)	
Execution (5)	
Viva(5)	
Total (15)	
Faculty Signature	

RESULT :

USING THE SET OPERATORS

EX-NO : 10

DATE:

1. The HR department needs a list of department IDs for departments that do not contain the job ID ST_CLERK. Use set operators to create this report.

QUERY:

```
SELECT DEPT_ID FROM DEPARTMENT MINUS  
SELECT DEPARTMENT_ID FROM EMPLOYEES WHERE JOB_ID = 'ST_CLERK';
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery, along with a search bar and user profile information for Afrin Fathima (afirinfathima014).

The SQL Workshop tab is selected. The schema dropdown shows WKSP_AFRINFATHIMA014. The SQL editor contains the following code:

```
1 SELECT DEPT_ID FROM DEPARTMENT MINUS  
2 SELECT DEPARTMENT_ID FROM EMPLOYEES WHERE JOB_ID = 'ST_CLERK';
```

The results tab is active, displaying the output of the query:

DEPT_ID
10
30
40
50
60
80
90

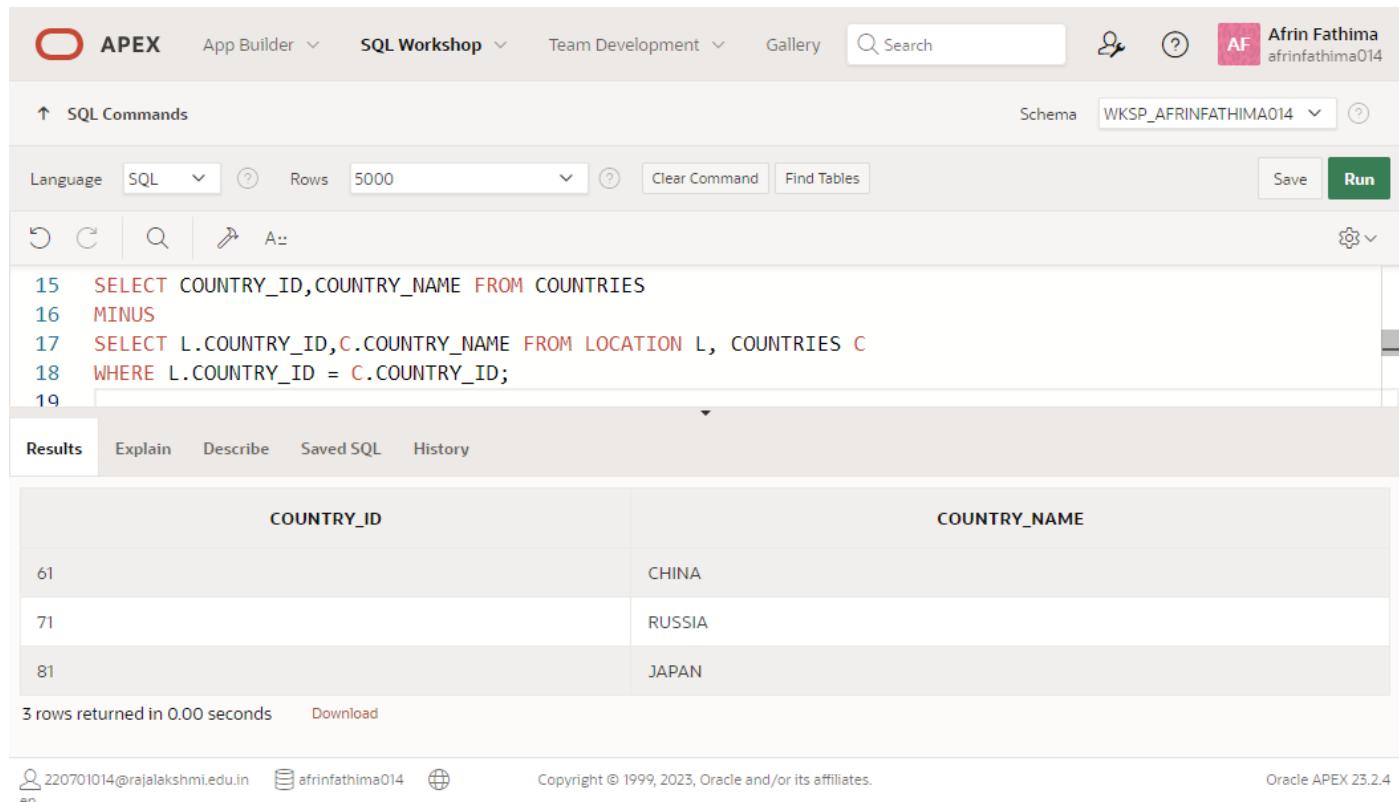
At the bottom of the page, there are footer links for email (220701014@rajalakshmi.edu.in), user profile (afirinfathima014), and copyright information (Copyright © 1999, 2023, Oracle and/or its affiliates). The page also indicates it is running on Oracle APEX 23.2.4.

2. The HR department needs a list of countries that have no departments located in them. Display the country ID and the name of the countries. Use set operators to create this report.

QUERY:

```
SELECT COUNTRY_ID,COUNTRY_NAME FROM COUNTRIES  
MINUS  
SELECT L.COUNTRY_ID,C.COUNTRY_NAME FROM LOCATION L, COUNTRIES C  
WHERE L.COUNTRY_ID = C.COUNTRY_ID;
```

OUTPUT:



The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, Gallery, and a search bar. The user is logged in as Afrin Fathima (afrafathima014). The schema selected is WKSP_AFRINFATHIMA014. The SQL command window displays the query code, and the results window shows the output of the executed query.

SQL Commands:

```
15  SELECT COUNTRY_ID,COUNTRY_NAME FROM COUNTRIES  
16  MINUS  
17  SELECT L.COUNTRY_ID,C.COUNTRY_NAME FROM LOCATION L, COUNTRIES C  
18  WHERE L.COUNTRY_ID = C.COUNTRY_ID;  
19
```

Results:

COUNTRY_ID	COUNTRY_NAME
61	CHINA
71	RUSSIA
81	JAPAN

3 rows returned in 0.00 seconds Download

en 220701014@rajalakshmi.edu.in afrafathima014 Copyright © 1999, 2023, Oracle and/or its affiliates. Oracle APEX 23.2.4

3. Produce a list of jobs for departments 10, 50, and 20, in that order. Display job ID and department ID using set operators.

QUERY:

```
SELECT DISTINCT JOB_ID, DEPARTMENT_ID FROM EMPLOYEES WHERE  
DEPARTMENT_ID = 10  
UNION ALL  
SELECT DISTINCT JOB_ID, DEPARTMENT_ID FROM EMPLOYEES WHERE  
DEPARTMENT_ID = 50  
UNION ALL  
SELECT DISTINCT JOB_ID, DEPARTMENT_ID FROM EMPLOYEES WHERE  
DEPARTMENT_ID = 20;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery, along with a search bar and user information for 'Afrin Fathima'.

In the SQL Workshop area, the schema is set to 'WKSP_AFRINFATHIMA014'. The code entered is:

```
22 SELECT DISTINCT JOB_ID, DEPARTMENT_ID FROM EMPLOYEES WHERE DEPARTMENT_ID = 10  
23 UNION ALL  
24 SELECT DISTINCT JOB_ID, DEPARTMENT_ID FROM EMPLOYEES WHERE DEPARTMENT_ID = 50  
25 UNION ALL  
26 SELECT DISTINCT JOB_ID, DEPARTMENT_ID FROM EMPLOYEES WHERE DEPARTMENT_ID = 20;
```

The 'Results' tab is selected, displaying the output:

JOB_ID	DEPARTMENT_ID
SL_REP	10
MK_MANAGER	10
HR_MANAGER	50
ST_CLERK	20

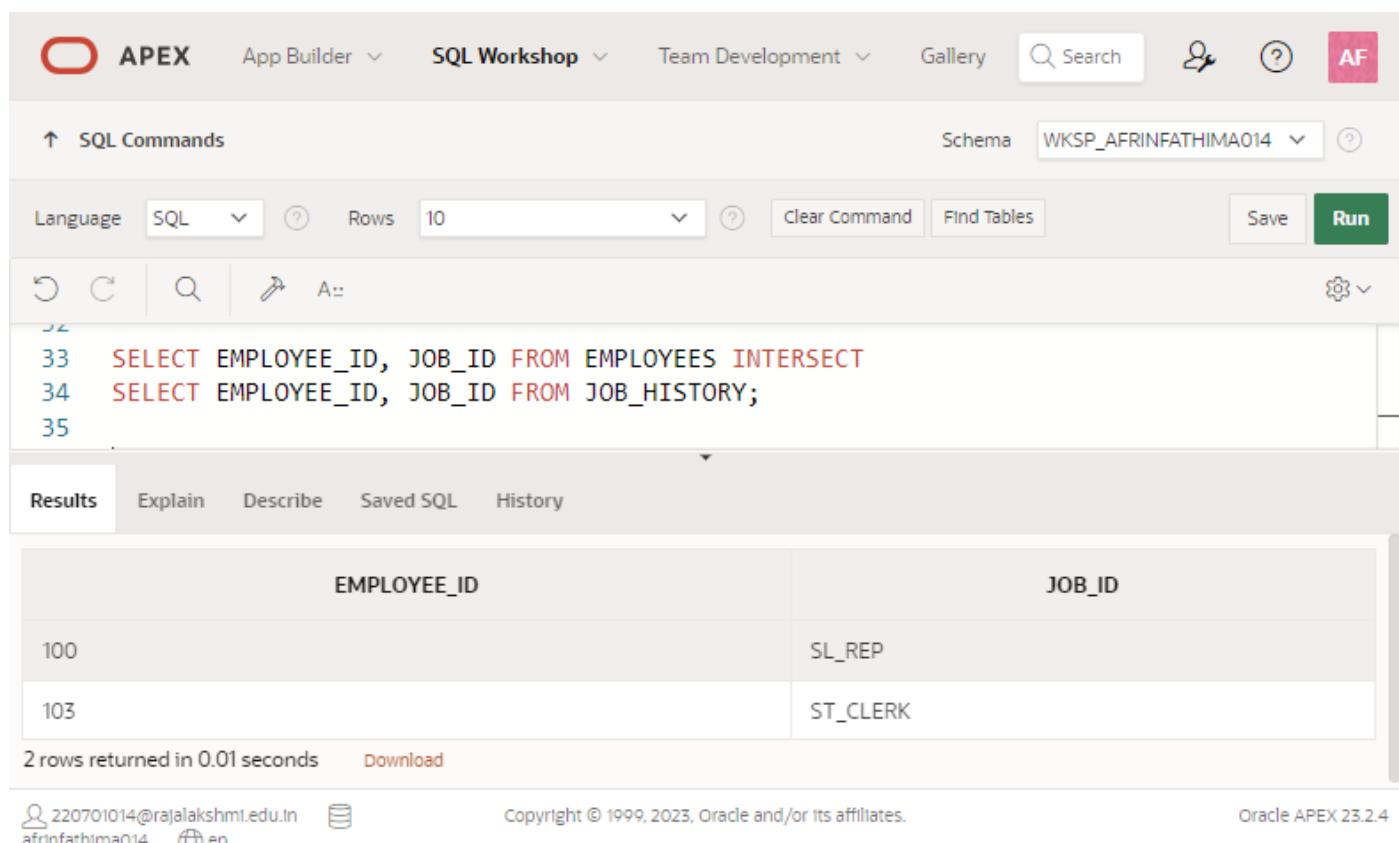
At the bottom, there are footer links for user information, copyright notice (Copyright © 1999, 2023, Oracle and/or its affiliates.), and the version 'Oracle APEX 23.2.4'.

4. Create a report that lists the employee IDs and job IDs of those employees who currently have a job title that is the same as their job title when they were initially hired by the company (that is, they changed jobs but have now gone back to doing their original job).

QUERY:

```
SELECT EMPLOYEE_ID, JOB_ID FROM EMPLOYEES INTERSECT  
SELECT EMPLOYEE_ID, JOB_ID FROM JOB_HISTORY;
```

OUTPUT:



The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, Gallery, Search, and user profile. The SQL Workshop tab is active. The schema dropdown is set to WKSP_AFRINFATHIMA014. The query editor window displays the following SQL code:

```
33: SELECT EMPLOYEE_ID, JOB_ID FROM EMPLOYEES INTERSECT  
34: SELECT EMPLOYEE_ID, JOB_ID FROM JOB_HISTORY;  
35:
```

The results section shows a table with two rows:

EMPLOYEE_ID	JOB_ID
100	SL_REP
103	ST_CLERK

Below the table, it says "2 rows returned in 0.01 seconds" and there is a "Download" link.

The bottom footer includes the user information "220701014@rajalakshmi.edu.in afriinfathima014 en", copyright notice "Copyright © 1999, 2023, Oracle and/or its affiliates.", and the version "Oracle APEX 23.2.4".

5. The HR department needs a report with the following specifications:

- Last name and department ID of all the employees from the EMPLOYEES table, regardless of whether or not they belong to a department.
- Department ID and department name of all the departments from the DEPARTMENTS table, regardless of whether or not they have employees working in them Write a compound query to accomplish this.

QUERY:

```
SELECT LAST_NAME,DEPARTMENT_ID,TO_CHAR(NULL) FROM EMPLOYEES  
UNION SELECT TO_CHAR(NULL),DEPT_ID,DEPT_NAME FROM DEPARTMENT;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop (selected), Team Development, and Gallery. On the right, there's a search bar, user profile (Afrin Fathima), and session information (WKSP_AFRINFATHIMA014). The main workspace displays the SQL command being run:

```
38  SELECT LAST_NAME,DEPARTMENT_ID,TO_CHAR(NULL) FROM EMPLOYEES UNION SELECT TO_CHAR(NULL),DEPT_ID,DEPT_NAME FROM DEPARTMENT;
```

The results tab is selected, showing the output of the query:

LAST_NAME	DEPARTMENT_ID	TO_CHAR(NULL)
DAVIES	30	-
DEV	80	-
EMANUEL	30	-
HARSH	60	-
JAM	80	-
JANE	50	-
JAY	10	-
JOBS	90	-
KOHLI	80	-
PARTHI	20	-
RAVI	10	-
RAVI	90	-
SIDDHU	60	-
UMA	20	-
VIJAY	40	-
-	10	MARKETING
-	20	STOCK
-	30	FINANCE
-	40	MANAGEMENT
-	50	HR
-	60	EXECUTIVE
-	80	MANUFACTURING
-	90	IT

At the bottom, it says "23 rows returned in 0.01 seconds" and provides download and refresh options. The footer includes copyright information and the version "Oracle APEX 23.2.4".

Evaluation Procedure	Marks awarded
Query(5)	
Execution (5)	
Viva(5)	
Total (15)	
Faculty Signature	

RESULT :

CREATING VIEWS

EX-NO : 11

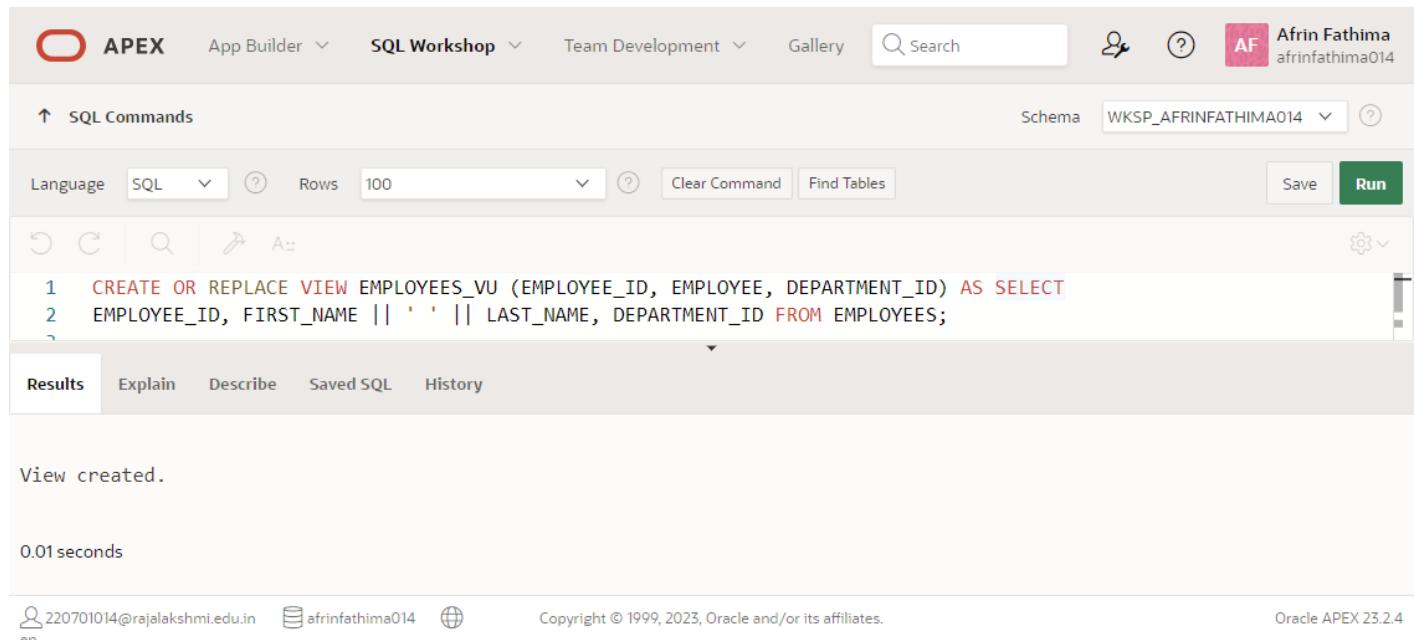
DATE:

1. Create a view called EMPLOYEE_VU based on the employee numbers, employee names and department numbers from the EMPLOYEES table. Change the heading for the employee name to EMPLOYEE.

QUERY:

```
CREATE OR REPLACE VIEW EMPLOYEES_VU (EMPLOYEE_ID, EMPLOYEE,  
DEPARTMENT_ID) AS SELECT  
EMPLOYEE_ID, FIRST_NAME || ' ' || LAST_NAME, DEPARTMENT_ID FROM  
EMPLOYEES;
```

OUTPUT:



The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, Gallery, and a search bar. On the right, there's a user profile for 'Afrin Fathima' (afrinfathima014). The main workspace is titled 'SQL Commands' and shows the following SQL code:

```
1 CREATE OR REPLACE VIEW EMPLOYEES_VU (EMPLOYEE_ID, EMPLOYEE, DEPARTMENT_ID) AS SELECT  
2 EMPLOYEE_ID, FIRST_NAME || ' ' || LAST_NAME, DEPARTMENT_ID FROM EMPLOYEES;
```

Below the code, there are tabs for 'Results', 'Explain', 'Describe', 'Saved SQL', and 'History'. The 'Results' tab is selected. The output area displays the message 'View created.' and '0.01 seconds'. At the bottom, it shows the user's email (220701014@rajalakshmi.edu.in), session ID (afrinfathima014), and copyright information (Copyright © 1999, 2023, Oracle and/or its affiliates).

2. Display the contents of the EMPLOYEES_VU view.

QUERY:

```
SELECT*FROM EMPLOYEES_VU;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for App Builder, SQL Workshop (selected), Team Development, and Gallery. A search bar and user profile 'Afrin Fathima' are also present. The main workspace displays the SQL command 'SELECT*FROM EMPLOYEES_VU;' and its results. The results are presented in a table with three columns: EMPLOYEE_ID, EMPLOYEE, and DEPARTMENT_ID. The data consists of 16 rows, each containing an employee's ID, name, and department ID. The table has a light gray background with alternating row colors. The bottom of the screen shows the message '16 rows returned in 0.01 seconds' and a 'Download' link. The footer contains copyright information and the version 'Oracle APEX 23.2.4'.

EMPLOYEE_ID	EMPLOYEE	DEPARTMENT_ID
109	AKAY KOHLI	80
111	STEVE JOBS	90
101	ASHA RAVI	10
102	TARA UMA	20
105	SANJ PARTHI	20
108	MIKA DEV	80
112	RITU RAVI	90
100	SAHANA JAY	10
113	KHAN HARSH	60
115	JOHN MATOS	80
114	VJ SIDDHU	60
104	MARY JANE	50
105	SANA EMANUEL	30
106	JOE VIJAY	40
110	BEN DAVIES	30
107	JIM JAM	80

16 rows returned in 0.01 seconds Download

220701014@rajalakshmi.edu.in afrinfathima014 en Copyright © 1999, 2023, Oracle and/or its affiliates. Oracle APEX 23.2.4

3. Select the view name and text from the USER_VIEWS data dictionary views.

QUERY:

```
SELECT VIEW_NAME,TEXT FROM USER_VIEWS;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for App Builder, SQL Workshop, Team Development, and Gallery, along with a search bar and user profile information for 'Afrin Fathima' (afrinfathima014). The main workspace is titled 'SQL Commands' and shows the following content:

Language: SQL | Rows: 100 | Clear Command | Find Tables | Save | Run

SQL command entered:

```
8  SELECT VIEW_NAME,TEXT FROM USER_VIEWS;
q
```

The results section displays the output of the query:

VIEW_NAME	TEXT
EMPLOYEES_VU	SELECT EMPLOYEE_ID,FIRST_NAME '' LAST_NAME,DEPARTMENT_ID FROM EMPLOYEES

1 rows returned in 0.05 seconds | Download

At the bottom, there are footer links for user information (220701014@rajalakshmi.edu.in, afrinfathima014), a globe icon, and copyright information (Copyright © 1999, 2023, Oracle and/or its affiliates). The page is identified as Oracle APEX 23.2.4.

4. Using your EMPLOYEES_VU view, enter a query to display all employees names and department.

QUERY:

```
SELECT EMPLOYEE,DEPARTMENT_ID FROM EMPLOYEES_VU;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop (selected), Team Development, Gallery, and a search bar. The right side of the header shows the user profile 'Afrin Fathima' and the schema 'WKSP_AFRINFATHIMA014'. Below the header, the SQL Commands section displays the query: '11 SELECT EMPLOYEE,DEPARTMENT_ID FROM EMPLOYEES_VU;'. The Results tab is selected, showing a table with two columns: 'EMPLOYEE' and 'DEPARTMENT_ID'. The table contains 16 rows of data. At the bottom of the results table, it says '16 rows returned in 0.03 seconds' and has a 'Download' link. The footer includes copyright information for Oracle and the APEX version 'Oracle APEX 23.2.4'.

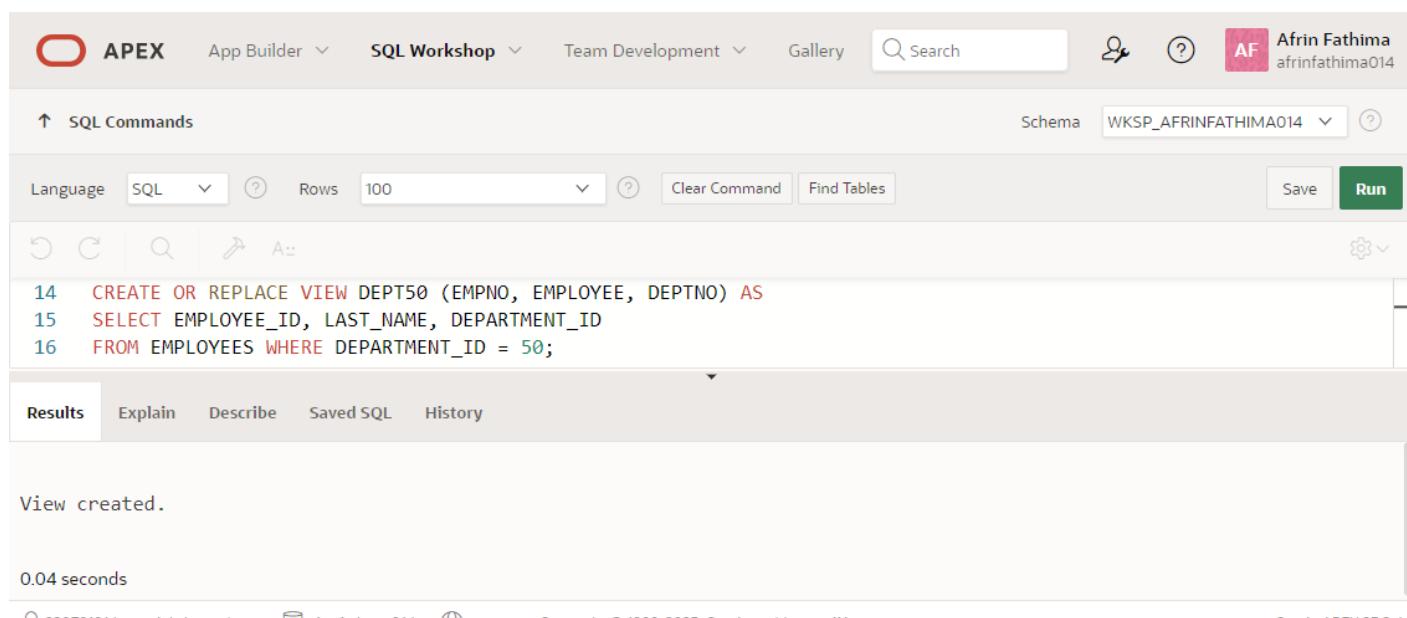
EMPLOYEE	DEPARTMENT_ID
AKAY KOHLI	80
STEVE JOBS	90
ASHA RAVI	10
TARA UMA	20
SANJ PARTHI	20
MIKA DEV	80
RITU RAVI	90
SAHANA JAY	10
KHAN HARSH	60
JOHN MATOS	80
VJ SIDDHU	60
MARY JANE	50
SANA EMANUEL	30
JOE VIJAY	40
BEN DAVIES	30
JIM JAM	80

5. Create a view named DEPT50 that contains the employee number, employee last names and department numbers for all employees in department 50. Label the view columns EMPNO, EMPLOYEE and DEPTNO. Do not allow an employee to be reassigned to another department through the view.

QUERY:

```
CREATE OR REPLACE VIEW DEPT50 (EMPNO, EMPLOYEE, DEPTNO) AS  
SELECT EMPLOYEE_ID, LAST_NAME, DEPARTMENT_ID  
FROM EMPLOYEES WHERE DEPARTMENT_ID = 50;
```

OUTPUT:



The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery, along with a search bar and user profile information for 'Afrin Fathima'.

In the SQL Commands section, the schema is set to 'WKSP_AFRINFATHIMA014'. The code entered is:

```
14 CREATE OR REPLACE VIEW DEPT50 (EMPNO, EMPLOYEE, DEPTNO) AS  
15 SELECT EMPLOYEE_ID, LAST_NAME, DEPARTMENT_ID  
16 FROM EMPLOYEES WHERE DEPARTMENT_ID = 50;
```

The Results tab shows the message "View created." and a execution time of "0.04 seconds".

At the bottom, the footer displays user details (220701014@rajalakshmi.edu.in, afrinfathima014), copyright information (Copyright © 1999, 2023, Oracle and/or its affiliates.), and the version "Oracle APEX 23.2.4".

6. Display the structure and contents of the DEPT50 view.

QUERY:

```
DESCRIBE DEPT50;  
SELECT*FROM DEPT50;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for App Builder, SQL Workshop, Team Development, and Gallery, along with a search bar and user profile information for 'Afrin Fathima'.

In the main workspace, the 'SQL Commands' tab is selected. The query entered is '19 DESCRIBE DEPT50;'. Below the query, there are tabs for Results, Explain, Describe (which is selected), Saved SQL, and History.

The results section displays the structure of the DEPT50 view:

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
DEPT50	EMPNO	NUMBER	-	6	0	-	-	-	-
	EMPLOYEE	VARCHAR2	25	-	-	-	-	-	-
	DEPTNO	NUMBER	-	4	0	-	✓	-	-

At the bottom of the page, there are footer links for 220701014@rajalakshmi.edu.in, afrinfathima014, and Oracle APEX 23.2.4.

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for App Builder, SQL Workshop, Team Development, and Gallery, along with a search bar and user profile information for 'Afrin Fathima'.

In the main workspace, the 'SQL Commands' tab is selected. The query entered is '20 SELECT*FROM DEPT50;'. Below the query, there are tabs for Results (which is selected), Explain, Describe, Saved SQL, and History.

The results section displays the contents of the DEPT50 view:

EMPNO	EMPLOYEE	DEPTNO
104	JANE	50
115	MATOS	50

Below the table, it says '2 rows returned in 0.01 seconds' and there is a 'Download' link. At the bottom, there are footer links for 220701014@rajalakshmi.edu.in, afrinfathima014, and Oracle APEX 23.2.4.

7. Attempt to reassign Matos to department 80.

QUERY:

```
UPDATE DEPT50 SET DEPTNO = 80  
WHERE EMPLOYEE = 'MATOS';
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for App Builder, SQL Workshop, Team Development, and Gallery, along with a search bar and user profile information for 'Afrin Fathima' (afrafathima014). The main workspace is titled 'SQL Commands' and contains the following SQL code:

```
24 UPDATE DEPT50 SET DEPTNO = 80  
25 WHERE EMPLOYEE = 'MATOS';
```

Below the code, the 'Results' tab is selected, showing the output: '1 row(s) updated.' The execution time is listed as '0.03 seconds'. At the bottom, footer information includes the user's email (220701014@rajalakshmi.edu.in), session ID (afrafathima014), and copyright notice (Copyright © 1999, 2023, Oracle and/or its affiliates). The Oracle APEX version is also mentioned as 23.2.4.

8. Create a view called SALARY_VU based on the employee last names, department names, salaries, and salary grades for all employees. Use the Employees, DEPARTMENTS and JOB_GRADE tables. Label the column Employee, Department, salary, and Grade respectively.

QUERY:

```
CREATE OR REPLACE VIEW SALARY_VU AS
SELECT E.LAST_NAME "EMPLOYEE", D.DEPT_NAME "DEPARTMENT",
E.SALARY "SALARY", J.GRADE_LEVEL "GRADES"
FROM EMPLOYEES E, DEPARTMENT D, JOB_GRADES J
WHERE E.DEPARTMENT_ID = D.DEPT_ID
AND E.SALARY BETWEEN J.LOWEST_SAL AND J.HIGHEST_SAL;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes 'APEX', 'App Builder', 'SQL Workshop' (selected), 'Team Development', 'Gallery', a search bar, and a user profile for 'Afrin Fathima'. The main workspace is titled 'SQL Commands' and shows the SQL code for creating the 'SALARY_VU' view. The code is as follows:

```
27 CREATE OR REPLACE VIEW SALARY_VU AS
28 SELECT E.LAST_NAME "EMPLOYEE", D.DEPT_NAME "DEPARTMENT",
29 E.SALARY "SALARY", J.GRADE_LEVEL "GRADES"
30 FROM EMPLOYEES E, DEPARTMENT D, JOB_GRADES J
31 WHERE E.DEPARTMENT_ID = D.DEPT_ID
32 AND E.SALARY BETWEEN J.LOWEST_SAL AND J.HIGHEST_SAL;
```

Below the code, there are tabs for 'Results', 'Explain', 'Describe', 'Saved SQL', and 'History'. The 'Results' tab is selected, displaying the message 'View created.' and a execution time of '0.03 seconds'. The bottom of the screen shows the database connection information ('en 220701014@rajalakshmi.edu.in', 'afrinfathima014') and the copyright notice ('Copyright © 1999, 2023, Oracle and/or its affiliates').

Evaluation Procedure	Marks awarded
Query(5)	
Execution (5)	
Viva(5)	
Total (15)	
Faculty Signature	

RESULT :

INTRO TO CONSTRAINTS

EX-NO : 12

DATE:

NOT NULL AND UNIQUE CONSTRAINTS

Global Fast Foods has been very successful this past year and has opened several new stores. They need to add a table to their database to store information about each of their store's locations. The owners want to make sure that all entries have an identification number, date opened, address, and city and that no other entry in the table can have the same email address. Based on this information, answer the following questions about the global_locations table. Use the table for your answers.

Global Fast Foods global_locations Table						
NAME	TYPE	LENGTH	PRECISION	SCALE	NULLABLE	DEFAULT
Id						
name						
date_opened						
address						
city						
zip/postal code						
phone						
email						
manager_id						
Emergency contact						

1. What is a “constraint” as it relates to data integrity?

Ans:

Database can be as reliable as the data in it, and database rules are implemented as Constraint to maintain data integrity.

2. What are the limitations of constraints that may be applied at the column level and at the table level?

Ans:

- Constraints referring to more than one column are defined at Table Level.
- NOT NULL constraint must be defined at column level as per ANSI/ISO SQL standard.

3. Why is it important to give meaningful names to constraints?

Ans:

- If a constraint is violated in a SQL statement execution, it is easy to identify the cause with user-named constraints.
- It is easy to alter names/drop constraint.

4. Based on the information provided by the owners, choose a datatype for each column. Indicate the length, precision, and scale for each NUMBER datatype.

Ans:

Global Fast Foods global_locations Table						
NAME	TYPE	DataType	LENGTH	PRECISION	SCALE	NULLABLE
id	pk	NUMBER	6	0		No
name		VARCHAR2	50			
date_opened		DATE				No
address		VARCHAR2	50			No
city		VARCHAR2	30			No
zip_postal_code		VARCHAR2	12			
phone		VARCHAR2	20			
email	uk	VARCHAR2	75			
manager_id		NUMBER	6	0		
emergency_contact		VARCHAR2	20			

5. Use “(nullable)” to indicate those columns that can have null values.

Ans:

Global Fast Foods global_locations Table						
NAME	TYPE	DataType	LENGTH	PRECISION	SCALE	NULLABLE
id	pk	NUMBER	6	0		No
name		VARCHAR2	50			Yes
date_opened		DATE				No
address		VARCHAR2	50			No
city		VARCHAR2	30			No
zip_postal_code		VARCHAR2	12			Yes
phone		VARCHAR2	20			Yes
email	uk	VARCHAR2	75			Yes
manager_id		NUMBER	6	0		Yes
emergency_contact		VARCHAR2	20			Yes

6. Write the CREATE TABLE statement for the Global Fast Foods locations table to define the constraints at the column level.

Ans:

```
CREATE TABLE f_global_locations
( id NUMBER(6,0) CONSTRAINT f_gln_id_pk PRIMARY KEY ,
  name VARCHAR2(50),
  date_opened DATE CONSTRAINT f_gln_dt_opened_nn NOT NULL ENABLE,
  address VARCHAR2(50) CONSTRAINT f_gln_add_nn NOT NULL ENABLE,
  city VARCHAR2(30) CONSTRAINT f_gln_city_nn NOT NULL ENABLE,
  zip_postal_code VARCHAR2(12),
  phone VARCHAR2(20),
  email VARCHAR2(75) CONSTRAINT f_gln_email_uk UNIQUE,
  manager_id NUMBER(6,0),
  emergency_contact VARCHAR2(20)
);
```

7. Execute the CREATE TABLE statement in Oracle Application Express.

Ans:

Table Created.

8. Execute a DESCRIBE command to view the Table Summary information.

Ans:

DESCRIBE f_global_locations;

9. Rewrite the CREATE TABLE statement for the Global Fast Foods locations table to define the UNIQUE constraints at the table level. Do not execute this statement.

NAME	TYPE	LENGTH	PRECISION	SCALE	NULLABLE	DEFAULT
id	number	4				
loc_name	varchar2	20			X	
	date					
address	varchar2	30				
city	varchar2	20				
zip_postal	varchar2	20			X	
phone	varchar2	15			X	
email	varchar2	80			X	
manager_id	number	4			X	
contact	varchar2	40			X	

Ans:

```
CREATE TABLE f_global_locations
( id NUMBER(6,0) CONSTRAINT f_gln_id_pk PRIMARY KEY ,
name VARCHAR2(50),
date_opened DATE CONSTRAINT f_gln_dt_opened_nn NOT NULL ENABLE,
address VARCHAR2(50) CONSTRAINT f_gln_add_nn NOT NULL ENABLE,
city VARCHAR2(30) CONSTRAINT f_gln_city_nn NOT NULL ENABLE,
zip_postal_code VARCHAR2(12),
phone VARCHAR2(20), email VARCHAR2(75) ,
manager_id NUMBER(6,0),
emergency_contact VARCHAR2(20),
CONSTRAINT f_gln_email_uk UNIQUE(email)
);
```

PRIMARY KEY, FOREIGN KEY, AND CHECK CONSTRAINTS

1. What is the purpose of a

- PRIMARY KEY
- FOREIGN KEY
- CHECK CONSTRAINT

Ans:

a. PRIMARY KEY

Uniquely identify each row in table.

b. FOREIGN KEY

Referential integrity constraint links back parent table's primary/unique key to child table's column.

c. CHECK CONSTRAINT

Explicitly define condition to be met by each row's fields. This condition must be returned as true or unknown.

2. Using the column information for the animals table below, name constraints where applicable at the table level, otherwise name them at the column level. Define the primary key (animal_id). The license_tag_number must be unique. The admit_date and vaccination_date columns cannot contain null values.

Ans:

animal_id NUMBER(6) - **PRIMARY KEY**

name VARCHAR2(25)

license_tag_number NUMBER(10) - **UNIQUE**

admit_date DATE -**NOT NULL**

adoption_id NUMBER(5),

vaccination_date DATE -**NOT NULL**

3. Create the animals table. Write the syntax you will use to create the table.

Ans:

```
CREATE TABLE animals
( animal_id NUMBER(6,0) CONSTRAINT anl_anl_id_pk PRIMARY KEY ,
name VARCHAR2(25),
license_tag_number NUMBER(10,0) CONSTRAINT anl_l_tag_num_uk
UNIQUE,
admit_date DATE CONSTRAINT anl_adt_dat_nn NOT NULL ENABLE,
adoption_id NUMBER(5,0),
vaccination_date DATE CONSTRAINT anl_vcc_dat_nn NOT NULL ENABLE
);
```

4. Enter one row into the table. Execute a SELECT * statement to verify your input. Refer to the graphic below for input.

ANIMAL_ID	NAME	LICENSE_TAG_NUMBER	ADMIT_DATE	ADOPTION_ID	VACCINATION_DATE
101	Spot	35540	10-Oct-2004	205	12-Oct-2004

Ans:

```
INSERT INTO animals (animal_id, name, license_tag_number, admit_date, adoption_id,
vaccination_date) VALUES( 101, 'Spot', 35540, TO_DATE('10-Oct-2004', 'DD-Mon-
YYYY'), 205, TO_DATE('12-Oct-2004', 'DD-Mon-YYYY'));
```

```
SELECT * FROM animals;
```

5. Write the syntax to create a foreign key (adoption_id) in the animals table that has a corresponding primary-key reference in the adoptions table. Show both the column-level and table-level syntax. Note that because you have not actually created an adoptions table, no adoption_id primary key exists, so the foreign key cannot be added to the animals table.

Ans:

COLUMN LEVEL STATEMENT:

```
ALTER TABLE animals MODIFY ( adoption_id NUMBER(5,0) CONSTRAINT  
anl_adopt_id_fk REFERENCES adoptions(id) ENABLE );
```

TABLE LEVEL STATEMENT:

```
ALTER TABLE animals ADD CONSTRAINT anl_adopt_id_fk FOREIGN KEY  
(adoption_id) REFERENCES adoptions(id) ENABLE;
```

6. What is the effect of setting the foreign key in the ANIMAL table as:

a. ON DELETE CASCADE

```
ALTER TABLE animals ADD CONSTRAINT anl_adopt_id_fk FOREIGN KEY  
(adoption_id) REFERENCES adoptions(id) ON DELETE CASCADE ENABLE ;
```

b. ON DELETE SET NULL

```
ALTER TABLE animals ADD CONSTRAINT anl_adopt_id_fk FOREIGN KEY  
(adoption_id) REFERENCES adoptions(id) ON DELETE SET NULL ENABLE ;
```

7. What are the restrictions on defining a CHECK constraint?

Ans:

- I cannot specify check constraint for a view however in this case I could use WITH CHECK OPTION clause
- I am restricted to columns from self table and fields in self row.
- I cannot use subqueries and scalar subquery expressions.

Evaluation Procedure	Marks awarded
Query(5)	
Execution (5)	
Viva(5)	
Total (15)	
Faculty Signature	

RESULT :

CREATING VIEWS

EX-NO : 13

DATE:

1. What are three uses for a view from a DBA's perspective?

Ans:

- Restrict access and display selective columns
- Reduce complexity of queries from other internal systems. So, providing a way to view same data in a different manner.
- Let the app code rely on views and allow the internal implementation of tables to be modified later.

2. Create a simple view called view_d_songs that contains the ID, title and artist from the DJs on Demand table for each "New Age" type code. In the subquery, use the alias "Song Title" for the title column.

Ans:

```
CREATE VIEW view_d_songs AS
SELECT d_songs.id, d_songs.title "Song Title", d_songs.artist
from d_songs INNER JOIN d_types ON d_songs.type_code = d_types.code
where d_types.description = 'New Age';
```

3. SELECT * FROM view_d_songs. What was returned?

Ans:

Results	Explain	Describe	Saved SQL	History
ID	Song Title		ARTIST	
47	Hurrah for Today		The Jubilant Trio	
49	Lets Celebrate		The Celebrants	

2 rows returned in 0.00 seconds [Download](#)

4. REPLACE view_d_songs. Add type_code to the column list. Use aliases for all columns.

Or use alias after the CREATE statement as shown.

Ans:

```
CREATE OR REPLACE VIEW view_d_songs AS
```

```
SELECT d_songs.id, d_songs.title "Song Title", d_songs.artist, d_songs.type_code  
from d_songs INNER JOIN d_types ON d_songs.type_code = d_types.code  
where d_types.description = 'New Age';
```

5. Jason Tsang, the disk jockey for DJs on Demand, needs a list of the past events and those planned for the coming months so he can make arrangements for each event's equipment setup. As the company manager, you do not want him to have access to the price that clients paid for their events. Create a view for Jason to use that displays the name of the event, the event date, and the theme description. Use aliases for each column name.

Ans:

```
CREATE OR REPLACE VIEW view_d_events_pkgs AS
```

```
SELECT evt.name "Name of Event", TO_CHAR(evt.event_date, 'dd-Month-yyyy') "Event  
date", thm.description "Theme description"  
FROM d_events evt INNER JOIN d_themes thm ON evt.theme_code = thm.code  
WHERE evt.event_date <= ADD_MONTHS(SYSDATE,1);
```

6. It is company policy that only upper-level management be allowed access to individual employee salaries. The department managers, however, need to know the minimum, maximum, and average salaries, grouped by department. Use the Oracle database to prepare a view that displays the needed information for department managers.

Ans:

```
CREATE OR REPLACE VIEW view_min_max_avg_dpt_salary ("Department Id",  
"Department Name", "Max Salary", "Min Salary", "Average Salary") AS
```

```
SELECT dpt.department_id, dpt.department_name, MAX(NVL(emp.salary,0)),  
MIN(NVL(emp.salary,0)), ROUND(AVG(NVL(emp.salary,0)),2)
```

```
FROM departments dpt LEFT OUTER JOIN employees emp ON dpt.department_id =  
emp.department_id
```

```
GROUP BY (dpt.department_id, dpt.department_name);
```

DML OPERATIONS AND VIEWS

Use the DESCRIBE statement to verify that you have tables named copy_d_songs, copy_d_events, copy_d_cds, and copy_d_clients in your schema. If you don't, write a query to create a copy of each.

1. Query the data dictionary USER_UPDATABLE_COLUMNS to make sure the columns in the base tables will allow UPDATE, INSERT, or DELETE. All table names in the data dictionary are stored in uppercase.

Ans:

```
SELECT owner, table_name, column_name, updatable,insertable, deletable  
FROM user_updatable_columns WHERE LOWER(table_name) = 'copy_d_songs';
```

```
SELECT owner, table_name, column_name, updatable,insertable, deletable  
FROM user_updatable_columns WHERE LOWER(table_name) = 'copy_d_events';
```

```
SELECT owner, table_name, column_name, updatable,insertable, deletable  
FROM user_updatable_columns WHERE LOWER(table_name) = 'copy_d_cds';
```

2. Use the CREATE or REPLACE option to create a view of all the columns in the copy_d_songs table called view_copy_d_songs.

Ans:

```
CREATE OR REPLACE VIEW view_copy_d_songs AS  
SELECT * FROM copy_d_songs;  
SELECT * FROM view_copy_d_songs;
```

3. Use view_copy_d_songs to INSERT the following data into the underlying copy_d_songs table. Execute a SELECT * from copy_d_songs to verify your DML command. See the graphic.

ID	TITLE	DURATION	ARTIST	TYPE_CODE
88	Mello Jello	2	The What	4

Ans:

```
INSERT INTO view_copy_d_songs(id,title,duration,artist,type_code)
VALUES(88,'Mello Jello','2 min','The What',4);
```

4. Create a view based on the DJs on Demand COPY_D_CDS table. Name the view read_copy_d_cds. Select all columns to be included in the view. Add a WHERE clause to restrict the year to 2000. Add the WITH READ ONLY option.

Ans:

```
CREATE OR REPLACE VIEW read_copy_d_cds AS
SELECT *
FROM copy_d_cds
WHERE year = '2000'
WITH READ ONLY ;
```

```
SELECT * FROM read_copy_d_cds;
```

5. Using the read_copy_d_cds view, execute a DELETE FROM read_copy_d_cds WHERE cd_number = 90;

Ans:

```
ORA-42399: cannot perform a DML operation on a read-only view
```

6. Use REPLACE to modify read_copy_d_cds. Replace the READ ONLY option with WITH CHECK OPTION CONSTRAINT ck_read_copy_d_cds. Execute a SELECT * statement to verify that the view exists.

Ans:

```
CREATE OR REPLACE VIEW read_copy_d_cds AS  
SELECT *  
FROM copy_d_cds  
WHERE year = '2000'  
WITH CHECK OPTION CONSTRAINT ck_read_copy_d_cds;
```

7. Use the read_copy_d_cds view to delete any CD of year 2000 from the underlying copy_d_cds.

Ans:

```
DELETE FROM read_copy_d_cds WHERE year = '2000';
```

8. Use the read_copy_d_cds view to delete cd_number 90 from the underlying copy_d_cds table.

Ans:

```
DELETE FROM read_copy_d_cds WHERE cd_number = 90;
```

9. Use the read_copy_d_cds view to delete year 2001 records.

Ans:

```
DELETE FROM read_copy_d_cds WHERE year = '2001';
```

10. Execute a SELECT * statement for the base table copy_d_cds. What rows were deleted?

Ans:

Only the one in problem 7 above, not the one in 8 and 9

11. What are the restrictions on modifying data through a view?

Ans:

DELETE,INSERT,MODIFY restricted if it contains:

Group functions

GROUP BY CLAUSE

DISTINCT

pseudocolumn ROWNUM Keyword

12. What is Moore's Law? Do you consider that it will continue to apply indefinitely?
Support your opinion with research from the internet.

Ans:

It roughly predicted that computing power nearly doubles every year. But Moore also said in 2005 that as per nature of exponential functions, this trend may not continue forever.

13. What is the “singularity” in terms of computing?

Ans:

Singularity is the hypothesis that the invention of artificial superintelligence will abruptly trigger runaway technological growth, resulting in unfathomable changes to human civilization

MANAGING VIEWS

1. Create a view from the copy_d_songs table called view_copy_d_songs that includes only the title and artist. Execute a SELECT * statement to verify that the view exists.

Ans:

```
CREATE OR REPLACE VIEW view_copy_d_songs AS SELECT title, artist FROM copy_d_songs; SELECT * FROM view_copy_d_songs;
```

2. Issue a DROP view_copy_d_songs. Execute a SELECT * statement to verify that the view has been deleted.

Ans:

```
DROP VIEW view_copy_d_songs;  
SELECT * FROM view_copy_d_songs;  
ORA-00942: table or view does not exist
```

3. Create a query that selects the last name and salary from the Oracle database. Rank the salaries from highest to lowest for the top three employees.

Ans:

```
SELECT * FROM (SELECT last_name, salary FROM employees ORDER BY salary DESC) WHERE ROWNUM <= 3;
```

4. Construct an inline view from the Oracle database that lists the last name, salary, department ID, and maximum salary for each department. Hint: One query will need to calculate maximum salary by department ID.

Ans:

```
SELECT empm.last_name, empm.salary, dptmx.department_id FROM (SELECT dpt.department_id, MAX(NVL(emp.salary,0)) max_dpt_sal FROM departments dpt LEFT OUTER JOIN employees emp ON dpt.department_id = emp.department_id GROUP BY dpt.department_id) dptmx LEFT OUTER JOIN employees empm ON dptmx.department_id = empm.department_id WHERE NVL(empm.salary,0) = dptmx.max_dpt_sal;
```

5. Create a query that will return the staff members of Global Fast Foods ranked by salary from lowest to highest.

Ans:

```
SELECT ROWNUM, last_name, salary FROM (SELECT * FROM f_staffs ORDER BY SALARY);
```

INDEXES AND SYNONYMS

1. What is an index and what is it used for?

Ans:

Definition: These are schema objects which make retrieval of rows from table faster.

Purpose: An index provides direct and fast access to row in table. They provide indexed path to locate data quickly, so hereby reduce necessity of heavy disk input/output operations.

2. What is a ROWID, and how is it used?

Ans:

Indexes use ROWID's (base 64 string representation of the row address containing block identifier, row location in the block and the database file identifier) which is the fastest way to access any particular row.

3. When will an index be created automatically?

Ans:

Primary key/unique key use already existing unique index but if index is not present already, it is created while applying unique/primary key constraint.

4. Create a nonunique index (foreign key) for the DJs on Demand column (cd_number) in the D_TRACK_LISTINGS table. Use the Oracle Application Express SQL Workshop Data Browser to confirm that the index was created.

Ans:

```
CREATE INDEX d_tlg_cd_number_fk_i ON d_track_listings (cd_number);
```

5. Use the join statement to display the indexes and uniqueness that exist in the data dictionary for the DJs on Demand D_SONGS table.

Ans:

```
SELECT ucm.index_name, ucm.column_name, ucm.column_position, uix.uniqueness FROM user_indexes uix INNER JOIN user_ind_columns ucm ON uix.index_name = ucm.index_name WHERE ucm.table_name = 'D_SONGS';
```

6. Use a SELECT statement to display the index_name, table_name, and uniqueness from the data dictionary USER_INDEXES for the DJs on Demand D_EVENTS table.

Ans:

```
SELECT index_name, table_name,uniqueness FROM user_indexes where table_name = 'D_EVENTS';
```

7. Write a query to create a synonym called dj_tracks for the DJs on Demand d_track_listings table.

Ans:

```
CREATE SYNONYM dj_tracks FOR d_track_listings;
```

8. Create a function-based index for the last_name column in DJs on Demand D_PARTNERS table that makes it possible not to have to capitalize the table name for searches. Write a SELECT statement that would use this index.

Ans:

```
CREATE INDEX d_ptr_last_name_idx ON d_partners(LOWER(last_name));
```

9. Create a synonym for the D_TRACK_LISTINGS table. Confirm that it has been created by querying the data dictionary.

Ans:

```
CREATE SYNONYM dj_tracks2 FOR d_track_listings;
```

```
SELECT * FROM user_synonyms WHERE table_NAME = UPPER('d_track_listings');
```

10. Drop the synonym that you created in question

Ans:

```
DROP SYNONYM dj_tracks2;
```

Evaluation Procedure	Marks awarded
Query(5)	
Execution (5)	
Viva(5)	
Total (15)	
Faculty Signature	

RESULT :

OTHER DATABASE OBJECTS

EX-NO : 14

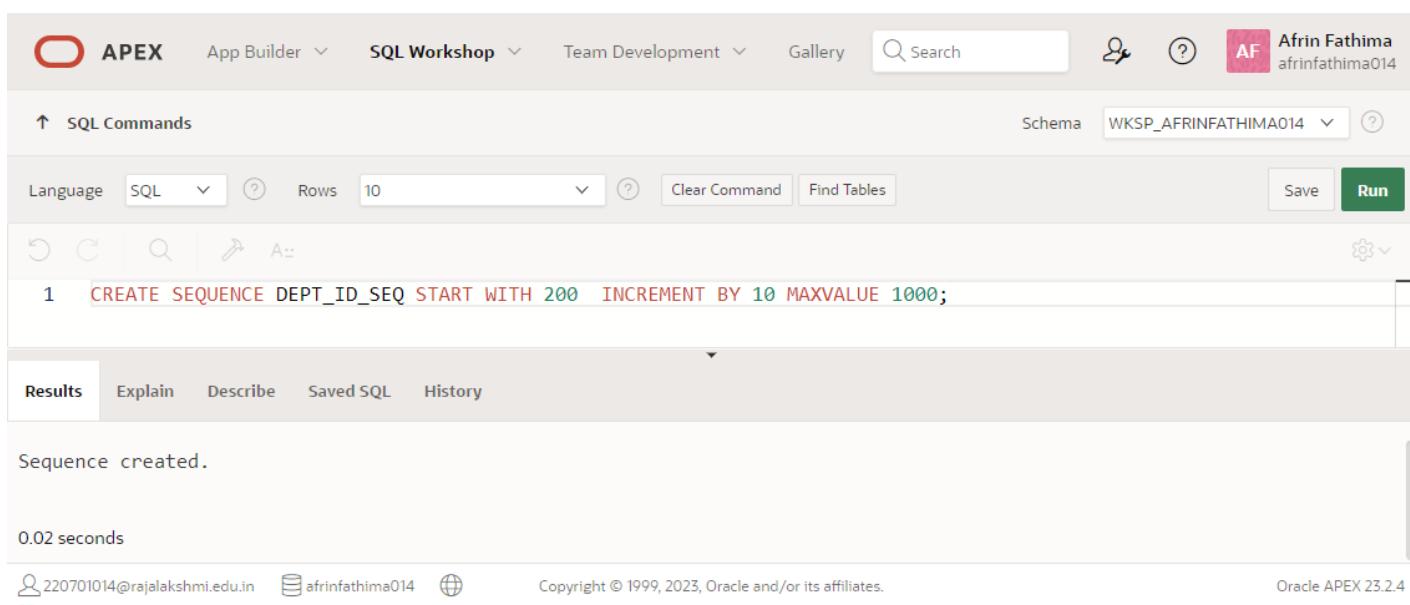
DATE:

1. Create a sequence to be used with the primary key column of the DEPT table. The sequence should start at 200 and have a maximum value of 1000. Have your sequence increment by ten numbers. Name the sequence DEPT_ID_SEQ.

QUERY:

```
CREATE SEQUENCE DEPT_ID_SEQ START WITH 200 INCREMENT BY 10  
MAXVALUE 1000;
```

OUTPUT:



The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery, along with a search bar and user profile information for 'Afrin Fathima' (afrafathima014). The main workspace is titled 'SQL Commands' and contains a command line with the following SQL statement:

```
1 CREATE SEQUENCE DEPT_ID_SEQ START WITH 200 INCREMENT BY 10 MAXVALUE 1000;
```

Below the command line, the 'Results' tab is selected, displaying the output: "Sequence created.". The bottom status bar shows the execution time as "0.02 seconds" and the database connection details: "220701014@rajalakshmi.edu.in" and "afrafathima014". It also includes copyright information for Oracle and the APEX version "Oracle APEX 23.2.4".

2. Write a query in a script to display the following information about your sequences: sequence name, maximum value, increment size, and last number.

QUERY:

```
SELECT SEQUENCE_NAME, MAX_VALUE, INCREMENT_BY, LAST_NUMBER  
FROM USER_SEQUENCES;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, Gallery, and a search bar. The right side shows the user profile of Afrin Fathima (afirinfathima014). The main workspace is titled "SQL Commands". The schema is set to "WKSP_AFRINFATHIMA014". The SQL editor contains the query: "4 SELECT SEQUENCE_NAME, MAX_VALUE, INCREMENT_BY, LAST_NUMBER FROM USER_SEQUENCES;". The results tab is selected, displaying a table with four columns: SEQUENCE_NAME, MAX_VALUE, INCREMENT_BY, and LAST_NUMBER. The data returned is as follows:

SEQUENCE_NAME	MAX_VALUE	INCREMENT_BY	LAST_NUMBER
DEPT_ID_SEQ	1000	10	200
DEPT_SEQ	99999999999999999999999999999999	1	50
EMP_SEQ	99999999999999999999999999999999	1	8000
ISEQ\$\$_270491045	99999999999999999999999999999999	1	1

Below the table, it says "4 rows returned in 0.01 seconds" and there is a "Download" link. The bottom footer includes links for user 220701014@rajalakshmi.edu.in, session afrinfathima014, and copyright information from Oracle.

3. Write a script to insert two rows into the DEPT table. Name your script lab12_3.sql. Be sure to use the sequence that you created for the ID column. Add two departments named Education and Administration. Confirm your additions. Run the commands in your script.

QUERY:

```
INSERT INTO DEPT VALUES (DEPT_ID_SEQ.NEXTVAL, 'EDUCATION');
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. At the top, there are navigation tabs: APEX, App Builder, SQL Workshop (selected), Team Development, and Gallery. A search bar and user profile information for 'Afrin Fathima' are also at the top. Below the header, the title 'SQL Commands' is displayed, along with schema selection 'WKSP_AFRINFATHIMA014'. The main area contains an SQL command line with the following content:

```
7  INSERT INTO DEPT VALUES (DEPT_ID_SEQ.NEXTVAL, 'EDUCATION');
```

Below the command line, the results tab is selected, showing the output: '1 row(s) inserted.' The bottom of the screen displays execution statistics '0.04 seconds' and copyright information 'Copyright © 1999, 2023, Oracle and/or its affiliates.'

4. Create a nonunique index on the foreign key column (DEPT_ID) in the EMP table.

QUERY:

```
CREATE INDEX EMP_DEPT_ID_IDX ON EMPLOYEES (DEPARTMENT_ID);
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for App Builder, SQL Workshop, Team Development, and Gallery, along with a search bar and user profile information for 'Afrin Fathima' (afrinfathima014). The main workspace is titled 'SQL Commands'. The command entered is 'CREATE INDEX EMP_DEPT_ID_IDX ON EMPLOYEES (DEPARTMENT_ID);'. The results section shows the message 'Index created.' and a execution time of '0.03 seconds'. The bottom footer displays copyright information for Oracle and the APEX version 'Oracle APEX 23.2.4'.

```
CREATE INDEX EMP_DEPT_ID_IDX ON EMPLOYEES (DEPARTMENT_ID);
```

Index created.
0.03 seconds

5. Display the indexes and uniqueness that exist in the data dictionary for the EMP table.

QUERY:

```
SELECT INDEX_NAME, TABLE_NAME,UNIQUENESS FROM USER_INDEXES  
WHERE TABLE_NAME='EMPLOYEES';
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for App Builder, SQL Workshop, Team Development, and Gallery, along with a search bar and user profile information for 'Afrin Fathima' (afrinfathima014). The main workspace is titled 'SQL Commands' and contains the following SQL query:

```
13  SELECT INDEX_NAME, TABLE_NAME,UNIQUENESS FROM USER_INDEXES WHERE TABLE_NAME='EMPLOYEES';
```

The results section displays the output of the query:

INDEX_NAME	TABLE_NAME	UNIQUENESS
EMP_DEPT_ID_IDX	EMPLOYEES	NONUNIQUE

Below the results, a message indicates '1 rows returned in 0.06 seconds' and provides a 'Download' link. The bottom of the page shows copyright information for Oracle and the APEX version (23.2.4), along with user details (220701014@rajalakshmi.edu.in, afrinfathima014).

Evaluation Procedure	Marks awarded
Query(5)	
Execution (5)	
Viva(5)	
Total (15)	
Faculty Signature	

RESULT :

CONTROLLING USER ACCESS

EX_NO:15

DATE:

1. What privilege should a user be given to log on to the Oracle Server? Is this a system or an object privilege?

The CREATE SESSION system privilege

2. What privilege should a user be given to create tables?

The CREATE TABLE privilege

3. If you create a table, who can pass along privileges to other users on your table?

You can, or anyone you have given those privileges to by using the WITH GRANT OPTION.

4. You are the DBA. You are creating many users who require the same system privileges. What should you use to make your job easier?

Create a role containing the system privileges and grant the role to the users

5. What command do you use to change your password?

The ALTER USER statement

6. Grant another user access to your DEPARTMENTS table. Have the user grant you query access to his or her DEPARTMENTS table.

Team 2 executes the GRANT statement. GRANT select ON departments TO <user1>;

Team 1 executes the GRANT statement. GRANT select ON departments TO <user2>;

7. Query all the rows in your DEPARTMENTS table.

SELECT * FROM departments;

8. Add a new row to your DEPARTMENTS table. Team 1 should add Education as department number 500. Team 2 should add Human Resources department number 510. Query the other team's table.

Team 1 executes this INSERT statement. INSERT INTO departments(department_id, department_name) VALUES (500, 'Education'); COMMIT;

Team 2 executes this INSERT statement. INSERT INTO departments(department_id, department_name) VALUES (510, 'Administration'); COMMIT;

9. Query the USER_TABLES data dictionary to see information about the tables that you own.

SELECT table_name FROM user_tables;

10. Revoke the SELECT privilege on your table from the other team.

Team 1 revokes the privilege.

```
REVOKE select  
ON departments  
FROM user2;
```

Team 2 revokes the privilege.

```
REVOKE select  
ON departments  
FROM user1;
```

11. Remove the row you inserted into the DEPARTMENTS table in step 8 and save the changes.

Team 1 executes this INSERT statement.

```
DELETE FROM departments  
WHERE department_id = 500;  
COMMIT;
```

Team 2 executes this INSERT statement.

```
DELETE FROM departments  
WHERE department_id = 510;  
COMMIT;
```

<u>Evaluation Procedure</u>	<u>Marks awarded</u>
<u>Practice Evaluation (5)</u>	
<u>Viva(5)</u>	
<u>Total (10)</u>	
<u>Faculty Signature</u>	

RESULT:

PL/SQL CONTROL STRUCTURES

EX-NO : 16

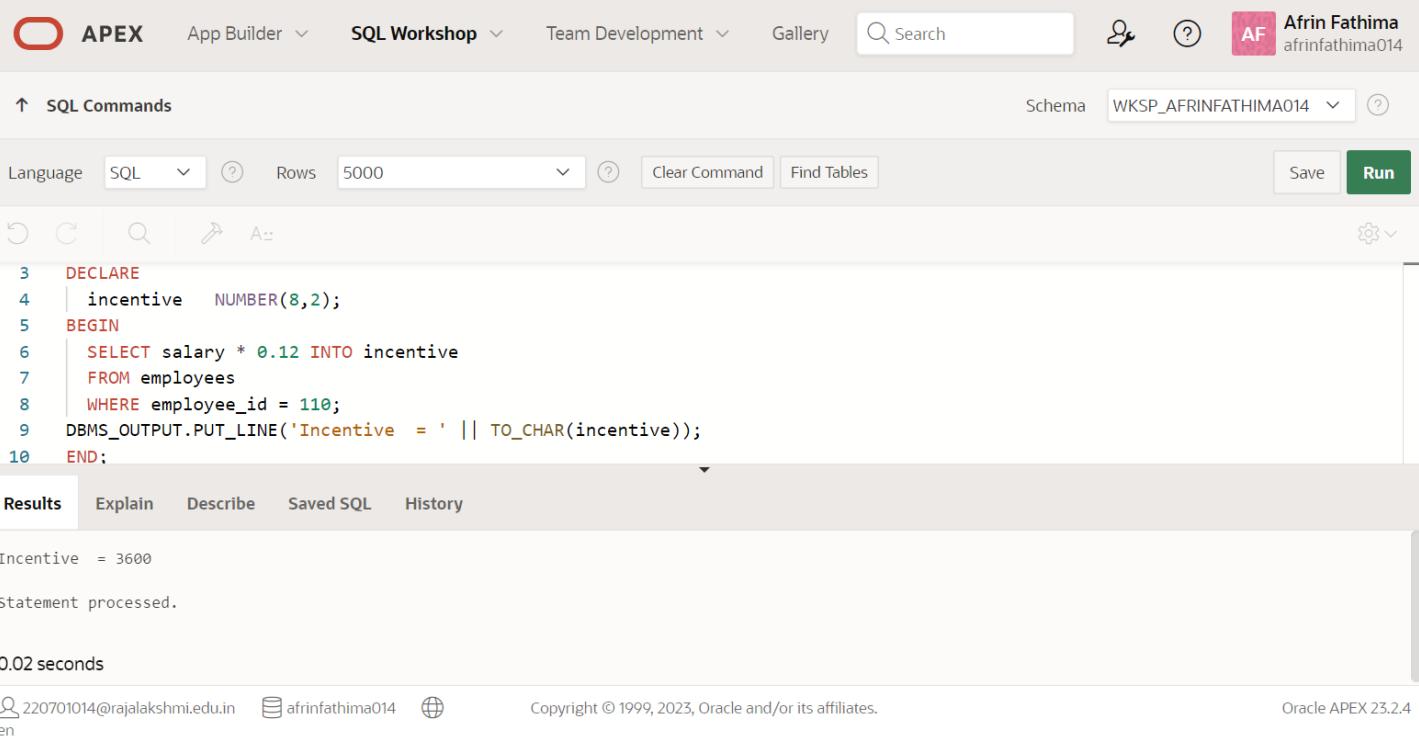
DATE:

1. Write a PL/SQL block to calculate the incentive of an employee whose ID is 110.

QUERY:

```
DECLARE
incentive NUMBER(8,2);
BEGIN
SELECT salary * 0.12 INTO incentive
FROM employees
WHERE employee_id = 110;
DBMS_OUTPUT.PUT_LINE('Incentive = ' || TO_CHAR(incentive));
END;
```

OUTPUT:



The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, Gallery, and a search bar. The user is identified as Afrin Fathima (afrafathima014). The main workspace is titled "SQL Commands". The SQL editor contains the provided PL/SQL code. The results tab shows the output of the executed query: "Incentive = 3600" and "Statement processed.". The bottom footer includes copyright information for Oracle and the APEX version.

```
3  DECLARE
4  |  incentive  NUMBER(8,2);
5  BEGIN
6  |  SELECT salary * 0.12 INTO incentive
7  |  FROM employees
8  |  WHERE employee_id = 110;
9  DBMS_OUTPUT.PUT_LINE('Incentive = ' || TO_CHAR(incentive));
10 END;
```

Incentive = 3600
Statement processed.
0.02 seconds

en 220701014@rajalakshmi.edu.in afrinfathima014 Copyright © 1999, 2023, Oracle and/or its affiliates. Oracle APEX 23.2.4

2. Write a PL/SQL block to show an invalid case-insensitive reference to a quoted and without quoted user-defined identifier.

QUERY:

```
DECLARE
  "WELCOME" varchar2(10) := 'welcome'; -- identifier with quotation
BEGIN
  DBMS_Output.Put_Line(WELCOME); --reference to the identifier without quotation
END;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery, along with a search bar and user profile information for 'Afrin Fathima'.

In the SQL Commands section, the schema is set to 'WKSP_AFRINFATHIMA014'. The code entered is:

```
15  DECLARE
16  |  "WELCOME" varchar2(10) := 'welcome'; -- identifier with quotation
17  BEGIN
18  |  DBMS_Output.Put_Line(WELCOME); --reference to the identifier without quotation
19  END;
```

The 'Results' tab is selected, showing the output:

```
welcome
statement processed.

0.00 seconds
```

At the bottom, there are footer links for 'en', '220701014@rajalakshmi.edu.in', 'afrinfathima014', and 'Copyright © 1999, 2023, Oracle and/or its affiliates.', along with the text 'Oracle APEX 23.2.4'.

3. Write a PL/SQL block to adjust the salary of the employee whose ID 122.

QUERY:

DECLARE

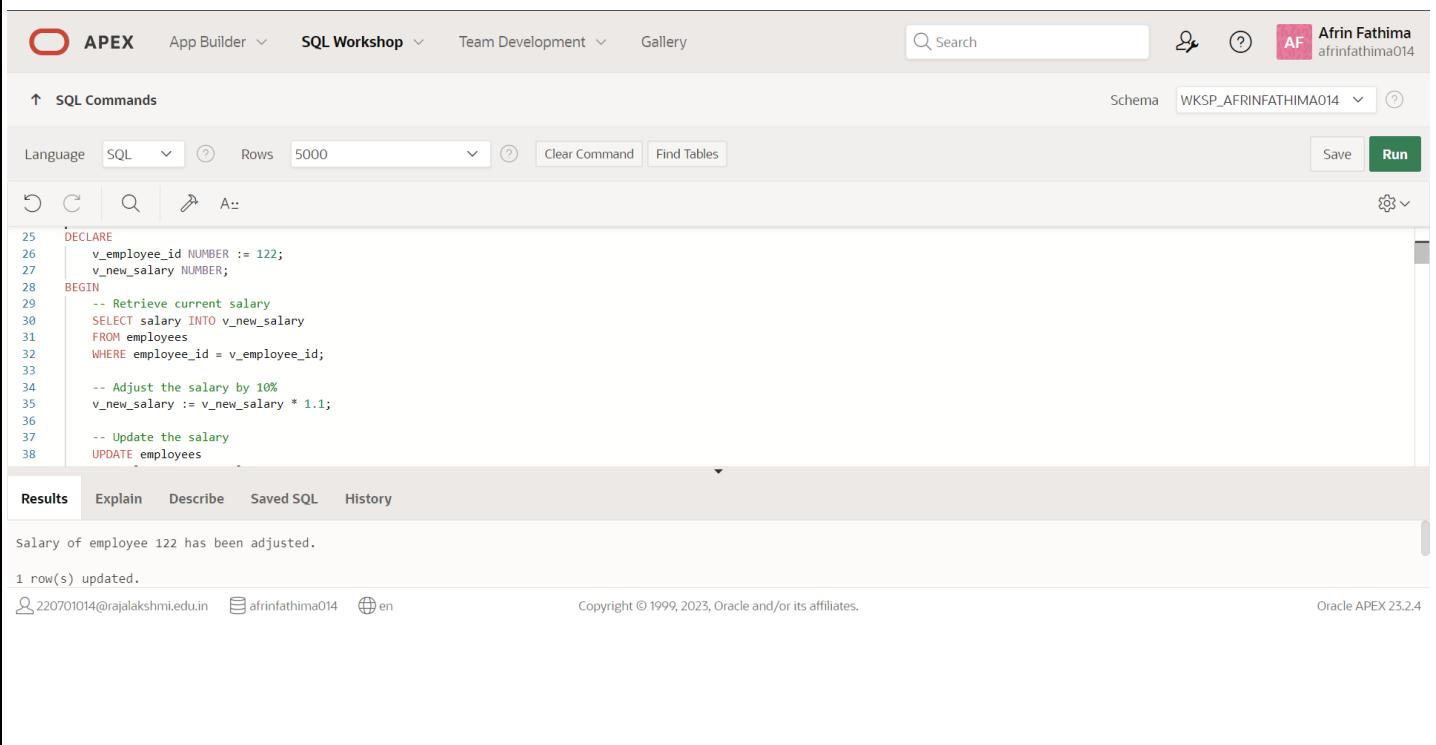
```
v_employee_id NUMBER := 122;
v_new_salary NUMBER;

BEGIN
    SELECT salary INTO v_new_salary
    FROM employees
    WHERE employee_id = v_employee_id;
    v_new_salary := v_new_salary * 1.1;
    UPDATE employees
    SET salary = v_new_salary
    WHERE employee_id = v_employee_id;
    COMMIT;

    DBMS_OUTPUT.PUT_LINE('Salary of employee ' || v_employee_id || ' has been
adjusted.');

EXCEPTION
    WHEN NO_DATA_FOUND THEN
        DBMS_OUTPUT.PUT_LINE('Employee with ID ' || v_employee_id || ' not found.');
    WHEN OTHERS THEN
        DBMS_OUTPUT.PUT_LINE('An error occurred: ' || SQLERRM);
        ROLLBACK;
END;
```

OUTPUT:



The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery. The user is identified as 'Afrin Fathima' (afrafathima014). The main workspace displays the PL/SQL code from the previous block. The code declares variables, retrieves the current salary for employee ID 122, calculates a 10% increase, and updates the salary. It also includes exception handling for no data found and other errors, along with DBMS_OUTPUT statements to log the process. The bottom section shows the results of the execution, indicating that 1 row was updated and confirming that the salary has been adjusted.

```
25 DECLARE
26     v_employee_id NUMBER := 122;
27     v_new_salary NUMBER;
28 BEGIN
29     -- Retrieve current salary
30     SELECT salary INTO v_new_salary
31     FROM employees
32     WHERE employee_id = v_employee_id;
33
34     -- Adjust the salary by 10%
35     v_new_salary := v_new_salary * 1.1;
36
37     -- Update the salary
38     UPDATE employees
```

Results Explain Describe Saved SQL History

Salary of employee 122 has been adjusted.

1 row(s) updated.

220701014@rajalakshmi.edu.in afrafathima014 en Copyright © 1999, 2023, Oracle and/or its affiliates. Oracle APEX 23.2.4

4. Write a PL/SQL block to create a procedure using the "IS [NOT] NULL Operator" and show AND operator returns TRUE if and only if both operands are TRUE.

QUERY:

```
CREATE OR REPLACE PROCEDURE pri_bool(
    boo_name  VARCHAR2,
    boo_val   BOOLEAN
) IS
BEGIN
    IF boo_val IS NULL THEN
        DBMS_OUTPUT.PUT_LINE( boo_name || ' = NULL');
    ELSIF boo_val = TRUE THEN
        DBMS_OUTPUT.PUT_LINE( boo_name || ' = TRUE');
    ELSE
        DBMS_OUTPUT.PUT_LINE( boo_name || ' = FALSE');
    END IF;
END;
/
DECLARE
    PROCEDURE pri_not_m (
        m BOOLEAN
    ) IS
    BEGIN
        pri_bool ('m', m);
        pri_bool ('NOT m', NOT m);
    END pri_not_m;
    BEGIN
        DBMS_OUTPUT.PUT_LINE('----- FOR m TRUE -');
        pri_not_m (TRUE);
        DBMS_OUTPUT.PUT_LINE('----- FOR m FALSE');
        pri_not_m (FALSE);
    END;
```

END,
OUTPUT:

The screenshot shows the Oracle APEX interface with the SQL Workshop module selected. The code editor displays a PL/SQL procedure named `pri_bool`. The procedure takes three parameters: `boo_name` (VARCHAR2), `boo_val` (BOOLEAN), and `m` (NUMBER). It uses `DBMS_OUTPUT.PUT_LINE` to print the value of `boo_name` followed by a condition based on `boo_val`. The results tab shows two executions of the procedure: one for `m = TRUE` which prints "m = TRUE", and one for `m = FALSE` which prints "m = FALSE". The bottom navigation bar includes tabs for Results, Explain, Describe, Saved SQL, and History.

```
CREATE OR REPLACE PROCEDURE pri_bool(
  boo_name  VARCHAR2,
  boo_val   BOOLEAN,
  m         NUMBER
) IS
BEGIN
  IF boo_val IS NULL THEN
    DBMS_OUTPUT.PUT_LINE( boo_name || ' = NULL');
  ELSIF boo_val = TRUE THEN
    DBMS_OUTPUT.PUT_LINE( boo_name || ' = TRUE');
  ELSE
    DBMS_OUTPUT.PUT_LINE( boo_name || ' = FALSE');
  END IF;
END;
/
```

Results Explain Describe Saved SQL History

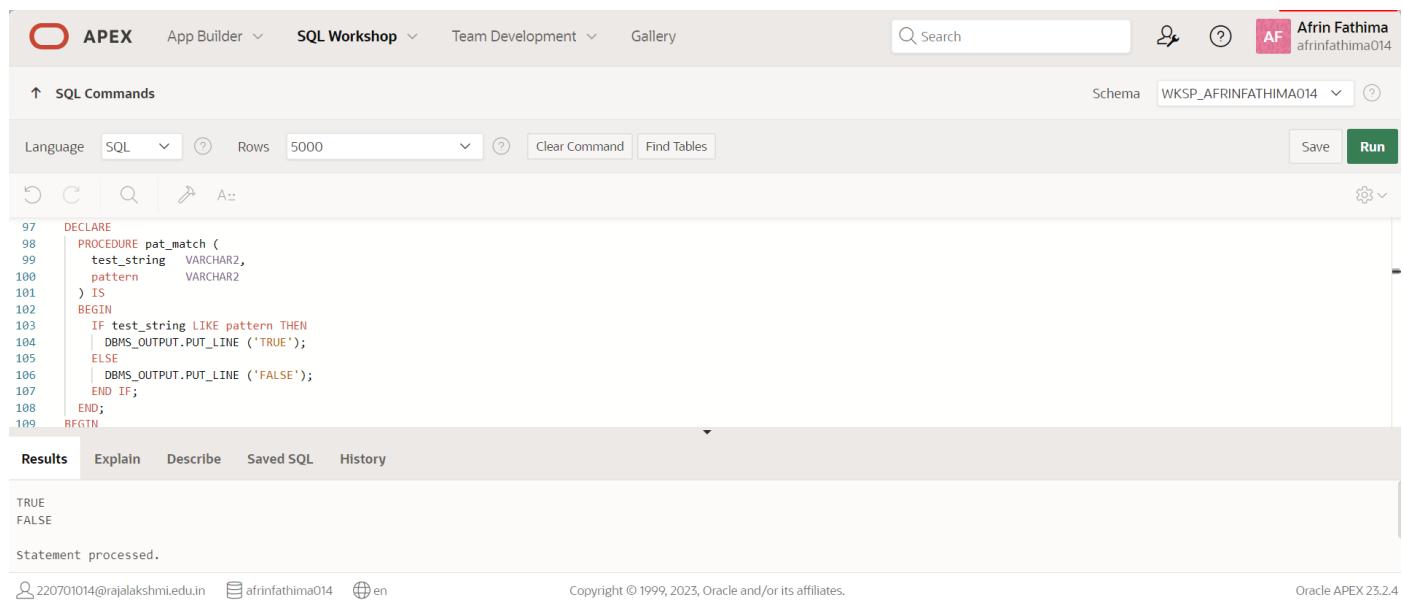
----- FOR m TRUE -----
m = TRUE
NOT m = FALSE
----- FOR m FALSE -----
m = FALSE

5. Write a PL/SQL block to describe the usage of LIKE operator including wildcard characters and escape character.

QUERY:

```
DECLARE
  PROCEDURE pat_match (
    test_string  VARCHAR2,
    pattern      VARCHAR2
  ) IS
BEGIN
  IF test_string LIKE pattern THEN
    DBMS_OUTPUT.PUT_LINE ('TRUE');
  ELSE
    DBMS_OUTPUT.PUT_LINE ('FALSE');
  END IF;
END;
BEGIN
  pat_match('Blweate', 'B%a_e');
  pat_match('Blweate', 'B%A_E');
END;
```

OUTPUT:



The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery. The user's profile 'Afrin Fathima' is visible on the right. The main workspace is titled 'SQL Commands'. The code area contains the provided PL/SQL block. The results section shows the output of the execution: 'TRUE' and 'FALSE'. The bottom status bar indicates the statement was processed successfully.

```
97  DECLARE
98    PROCEDURE pat_match (
99      test_string  VARCHAR2,
100     pattern      VARCHAR2
101   ) IS
102 BEGIN
103   IF test_string LIKE pattern THEN
104     DBMS_OUTPUT.PUT_LINE ('TRUE');
105   ELSE
106     DBMS_OUTPUT.PUT_LINE ('FALSE');
107   END IF;
108 END;
109 RFGTN
```

Results Explain Describe Saved SQL History

TRUE
FALSE

Statement processed.

220701014@rajalakshmi.edu.in afrafathima014 en Copyright © 1999, 2023, Oracle and/or its affiliates. Oracle APEX 23.2.4

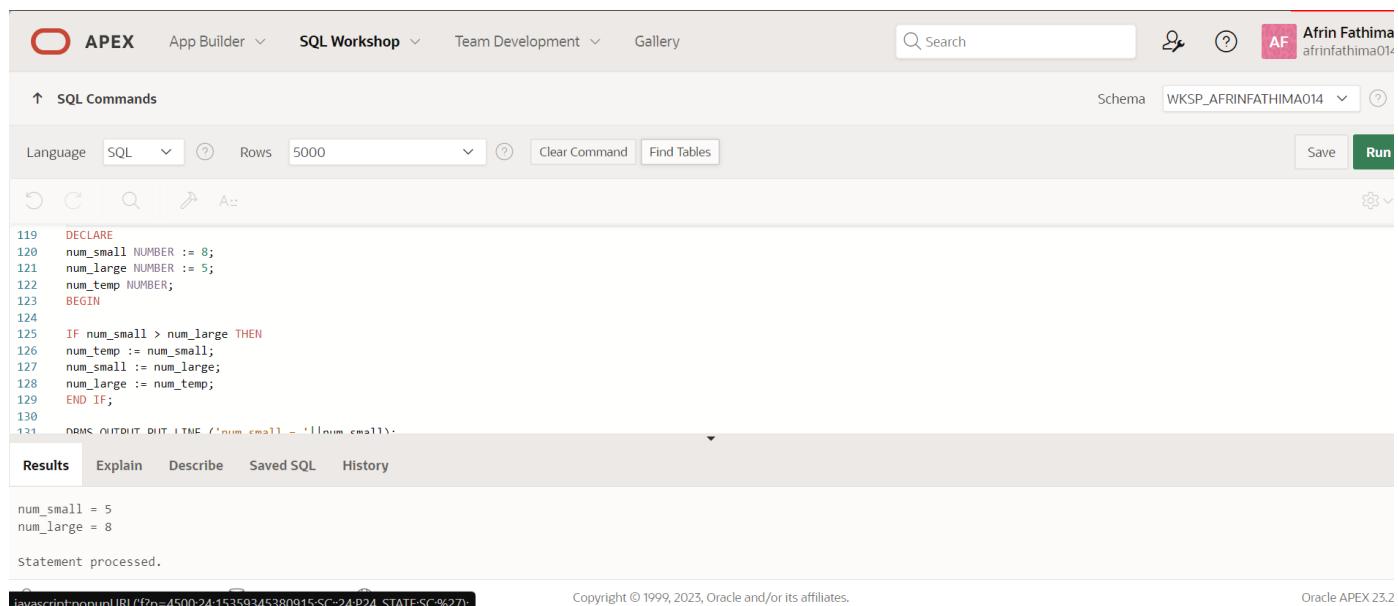
6. Write a PL/SQL program to arrange the number of two variable in such a way that the small number will store in num_small variable and large number will store in num_large variable.

QUERY:

```
DECLARE
num_small NUMBER := 8;
num_large NUMBER := 5;
num_temp NUMBER;
BEGIN
IF num_small > num_large THEN
num_temp := num_small;
num_small := num_large;
num_large := num_temp;
END IF;

DBMS_OUTPUT.PUT_LINE ('num_small ='||num_small);
DBMS_OUTPUT.PUT_LINE ('num_large ='||num_large);
END;
```

OUTPUT:



The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery. On the right, there's a search bar, a user profile for 'Afrin Fathima' (afrinfathima014), and a 'Run' button. The main workspace is titled 'SQL Commands'. It contains a code editor with the following PL/SQL block:

```
119  DECLARE
120    num_small NUMBER := 8;
121    num_large NUMBER := 5;
122    num_temp NUMBER;
123    BEGIN
124      IF num_small > num_large THEN
125        num_temp := num_small;
126        num_small := num_large;
127        num_large := num_temp;
128      END IF;
129
130      DBMS_OUTPUT.PUT_LINE ('num_small ='||num_small);
131      DBMS_OUTPUT.PUT_LINE ('num_large ='||num_large);
132
133  END;
```

Below the code editor, there are tabs for 'Results', 'Explain', 'Describe', 'Saved SQL', and 'History'. The 'Results' tab is selected, showing the output of the executed code:

```
num_small = 5
num_large = 8

Statement processed.
```

At the bottom of the page, there are copyright notices: 'Copyright © 1999, 2023, Oracle and/or its affiliates.' and 'Oracle APEX 23.2'.

7. Write a PL/SQL procedure to calculate the incentive on a target achieved and display the message either the record updated or not.

QUERY:

```
DECLARE
  PROCEDURE test1 (sal_achieve NUMBER, target_qty NUMBER, emp_id NUMBER )
  IS
    incentive NUMBER := 0;
    updated VARCHAR2(3) := 'No';
  BEGIN
    IF sal_achieve > (target_qty + 200) THEN
      incentive := (sal_achieve - target_qty)/4;

      UPDATE employees
      SET salary = salary + incentive
      WHERE employee_id = emp_id;

      updated := 'Yes';
    END IF;
    DBMS_OUTPUT.PUT_LINE (
      'Table updated? ' || updated || ',' ||
      'incentive = ' || incentive || !
    );
  END test1;
BEGIN
  test1(2300, 2000, 144);
  test1(3600, 3000, 145);
END;
OUTPUT:
```

OUTPUT:

APEX App Builder SQL Workshop Team Development Gallery

SQL Commands Schema WKSP_AFRINFATHIMA014

Language SQL Rows 5000 Clear Command Find Tables Save Run

DECLARE
PROCEDURE test1 (
 sal_achieve NUMBER,
 target_qty NUMBER,
 emp_id NUMBER
)
IS
 incentive NUMBER := 0;
 updated VARCHAR2(3) := 'No';
BEGIN
 IF sal_achieve > (target_qty + 200) THEN
 incentive := (sal_achieve - target_qty)/4;
 END IF;
 updated := 'Yes';
 UPDATE sal_achieve SET incentive = incentive, updated = updated WHERE emp_id = emp_id;
END test1;

Results Explain Describe Saved SQL History

Table updated? Yes, incentive = 75.
Table updated? Yes, incentive = 150.

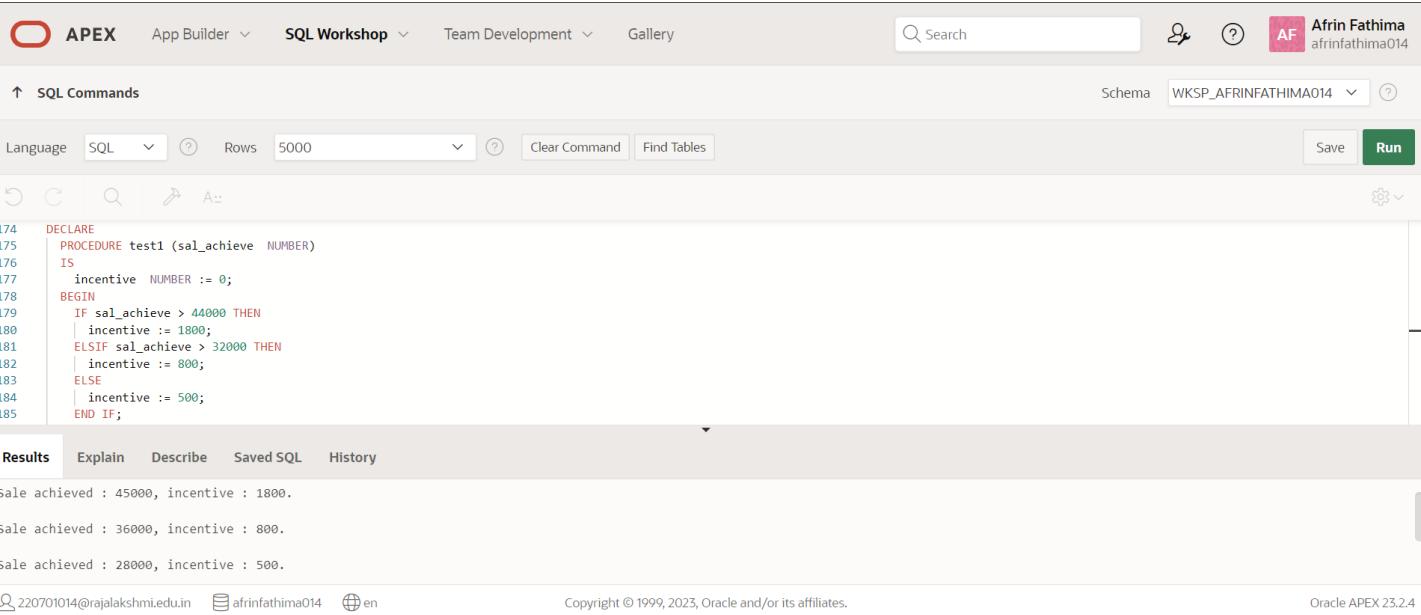
1 row(s) updated.

8. Write a PL/SQL procedure to calculate incentive achieved according to the specific sale limit.

QUERY:

```
DECLARE
PROCEDURE test1 (sal_achieve NUMBER)
IS
    incentive NUMBER := 0;
BEGIN
    IF sal_achieve > 44000 THEN
        incentive := 1800;
    ELSIF sal_achieve > 32000 THEN
        incentive := 800;
    ELSE
        incentive := 500;
    END IF;
    DBMS_OUTPUT.NEW_LINE;
    DBMS_OUTPUT.PUT_LINE (
        'Sale achieved : ' || sal_achieve || ', incentive : ' || incentive || ')';
END test1;
BEGIN
    test1(45000);
    test1(36000);
    test1(28000);
END;
```

OUTPUT:



The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop (which is selected), Team Development, and Gallery. A search bar and user profile 'Afrin Fathima' are also present. The main workspace displays the PL/SQL code for 'test1'. The code uses an IF-ELSIF-ELSE structure to calculate an incentive based on a salary achievement. It includes DBMS_OUTPUT statements to print the results. The code is numbered from 174 to 185. Below the code, there are tabs for Results, Explain, Describe, Saved SQL, and History. The Results tab is active, showing the three execution outputs:

```
174 DECLARE
175 PROCEDURE test1 (sal_achieve NUMBER)
176 IS
177     incentive NUMBER := 0;
178 BEGIN
179     IF sal_achieve > 44000 THEN
180         | incentive := 1800;
181     ELSIF sal_achieve > 32000 THEN
182         | incentive := 800;
183     ELSE
184         | incentive := 500;
185     END IF;

```

Sale achieved : 45000, incentive : 1800.
Sale achieved : 36000, incentive : 800.
Sale achieved : 28000, incentive : 500.

At the bottom, footer information includes the URL '220701014@rajalakshmi.edu.in', the schema name 'WKSP_AFRINFATHIMA014', copyright notice 'Copyright © 1999, 2023, Oracle and/or its affiliates.', and the version 'Oracle APEX 23.2.4'.

9. Write a PL/SQL program to count number of employees in department 50 and check whether this department have any vacancies or not. There are 45 vacancies in this department.

QUERY:

DECLARE

```
v_emp_count NUMBER;
v_vacancies NUMBER := 45;
```

BEGIN

```
-- Count the number of employees in department 50
```

```
SELECT COUNT(*)
```

```
INTO v_emp_count
```

```
FROM employees
```

```
WHERE department_id = 50;
```

```
-- Display the number of employees in department 50
```

```
DBMS_OUTPUT.PUT_LINE('Number of employees in department 50: ' || v_emp_count);
```

```
-- Check if there are any vacancies
```

```
IF v_emp_count < v_vacancies THEN
```

```
    DBMS_OUTPUT.PUT_LINE('There are vacancies in department 50.');
```

```
ELSE
```

```
    DBMS_OUTPUT.PUT_LINE('There are no vacancies in department 50.');
```

```
END IF;
```

```
END;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery. A search bar and user profile 'Afrin Fathima' are also present. The main area is titled 'SQL Commands' and contains the following PL/SQL code:

```
200 DECLARE
201   v_emp_count NUMBER;
202   v_vacancies NUMBER := 45;
203 BEGIN
204   -- Count the number of employees in department 50
205   SELECT COUNT(*)
206   INTO v_emp_count
207   FROM employees
208   WHERE department_id = 50;
209
210   -- Display the number of employees in department 50
211   DBMS_OUTPUT.PUT_LINE('Number of employees in department 50: ' || v_emp_count);
212
213   -- Check if there are any vacancies
214   IF v_emp_count < v_vacancies THEN
215       DBMS_OUTPUT.PUT_LINE('There are vacancies in department 50.');
216   ELSE
217       DBMS_OUTPUT.PUT_LINE('There are no vacancies in department 50.');
218   END IF;
219
220 END;
```

The results tab at the bottom shows the output of the query:

```
Number of employees in department 50: 1
There are vacancies in department 50.
```

At the bottom of the page, footer information includes the URL '220701014@rajalakshmi.edu.in', the schema name 'afrinfathima014', and the copyright notice 'Copyright © 1999, 2023, Oracle and/or its affiliates.' The page is identified as 'Oracle APEX 23.2.4'.

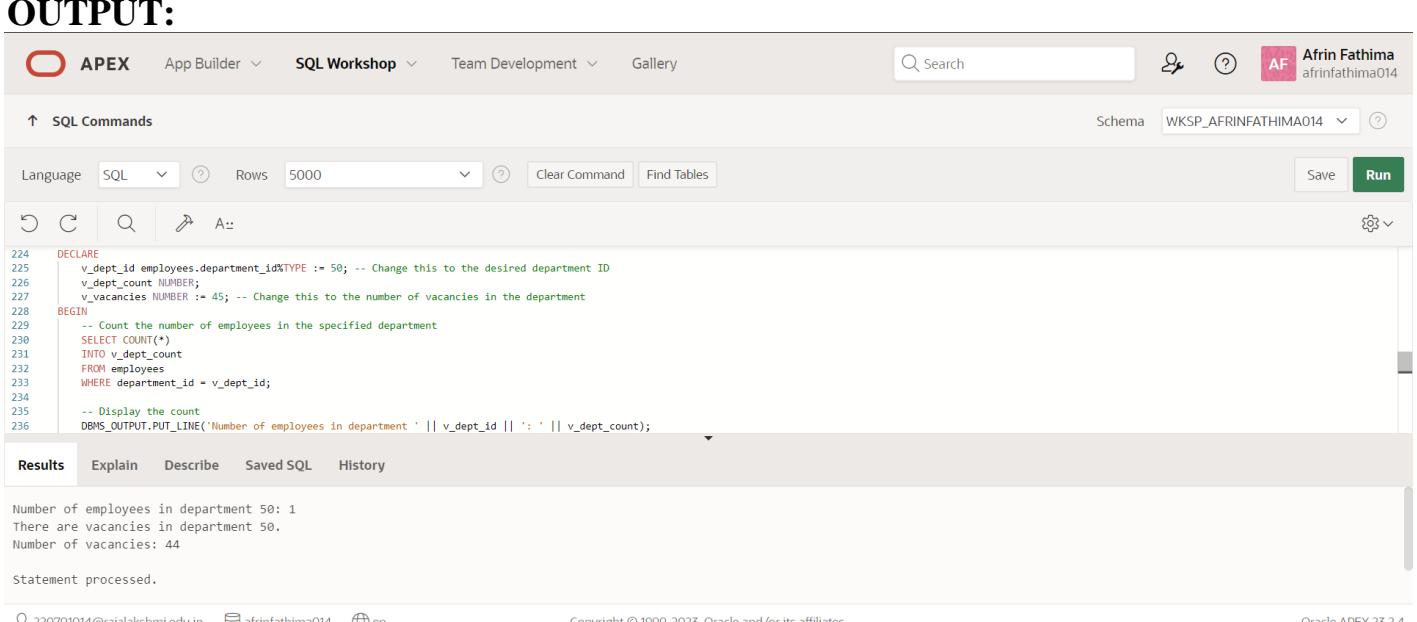
10. Write a PL/SQL program to count number of employees in a specific department and check whether this department have any vacancies or not. If any vacancies, how many vacancies are in that department.

QUERY:

DECLARE

```
v_dept_id employees.department_id%TYPE := 50; -- Change this to the desired department ID
v_dept_count NUMBER;
v_vacancies NUMBER := 45; -- Change this to the number of vacancies in the department
BEGIN
    SELECT COUNT(*)
    INTO v_dept_count
    FROM employees
    WHERE department_id = v_dept_id;
    DBMS_OUTPUT.PUT_LINE('Number of employees in department ' || v_dept_id || ':' || v_dept_count);
    IF v_dept_count < v_vacancies THEN
        DBMS_OUTPUT.PUT_LINE('There are vacancies in department ' || v_dept_id || '.');
        DBMS_OUTPUT.PUT_LINE('Number of vacancies: ' || (v_vacancies - v_dept_count));
    ELSE
        DBMS_OUTPUT.PUT_LINE('There are no vacancies in department ' || v_dept_id || '.');
    END IF;
END;
```

OUTPUT:



The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery. The user is logged in as 'Afrin Fathima' (afrinfathima014). The main workspace is titled 'SQL Commands'. The code area contains the PL/SQL block from above, with line numbers 224 through 236. The 'Run' button at the bottom right of the code editor is highlighted in green. Below the code, the 'Results' tab is selected, displaying the output of the executed query. The output shows the count of employees (1), a message indicating vacancies (There are vacancies in department 50.), and the number of vacancies (44). The status bar at the bottom indicates the statement was processed.

```
224 DECLARE
225     v_dept_id employees.department_id%TYPE := 50; -- Change this to the desired department ID
226     v_dept_count NUMBER;
227     v_vacancies NUMBER := 45; -- Change this to the number of vacancies in the department
228 BEGIN
229     -- Count the number of employees in the specified department
230     SELECT COUNT(*)
231     INTO v_dept_count
232     FROM employees
233     WHERE department_id = v_dept_id;
234
235     -- Display the count
236     DBMS_OUTPUT.PUT_LINE('Number of employees in department ' || v_dept_id || ':' || v_dept_count);
```

Results Explain Describe Saved SQL History

```
Number of employees in department 50: 1
There are vacancies in department 50.
Number of vacancies: 44

Statement processed.
```

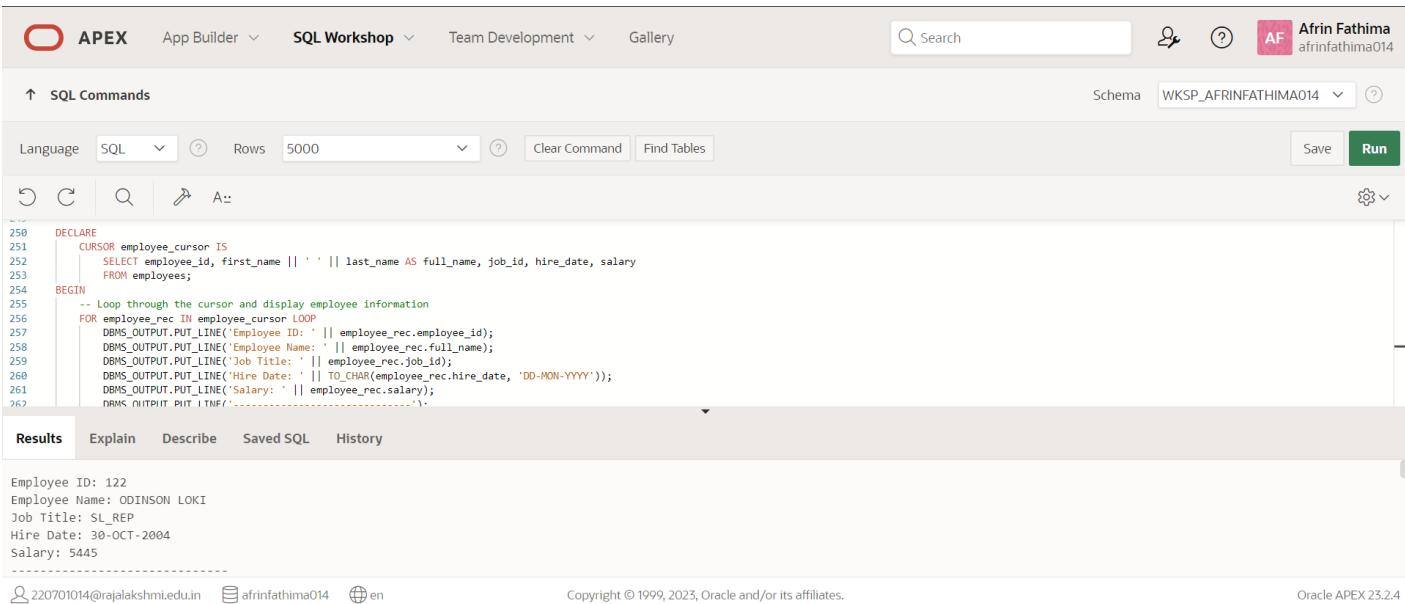
220701014@rajalakshmi.edu.in afrinfathima014 en Copyright © 1999, 2023, Oracle and/or its affiliates. Oracle APEX 23.2.4

11. Write a PL/SQL program to display the employee IDs, names, job titles, hire dates, and salaries of all employees.

QUERY:

```
DECLARE
  CURSOR employee_cursor IS
    SELECT employee_id, first_name || ' ' || last_name AS full_name, job_id, hire_date,
salary
      FROM employees;
BEGIN
  -- Loop through the cursor and display employee information
  FOR employee_rec IN employee_cursor LOOP
    DBMS_OUTPUT.PUT_LINE('Employee ID: ' || employee_rec.employee_id);
    DBMS_OUTPUT.PUT_LINE('Employee Name: ' || employee_rec.full_name);
    DBMS_OUTPUT.PUT_LINE('Job Title: ' || employee_rec.job_id);
    DBMS_OUTPUT.PUT_LINE('Hire Date: ' || TO_CHAR(employee_rec.hire_date,
'DD-MON-YYYY'));
    DBMS_OUTPUT.PUT_LINE('Salary: ' || employee_rec.salary);
    DBMS_OUTPUT.PUT_LINE('-----');
  END LOOP;
END;
```

OUTPUT:



The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery. A search bar and user profile 'Afrin Fathima' are also present. The main workspace is titled 'SQL Commands'. The code area contains a PL/SQL block with line numbers 250 to 262. The code is as follows:

```
250 DECLARE
251   CURSOR employee_cursor IS
252     SELECT employee_id, first_name || ' ' || last_name AS full_name, job_id, hire_date, salary
253       FROM employees;
254 BEGIN
255   -- Loop through the cursor and display employee information
256   FOR employee_rec IN employee_cursor LOOP
257     DBMS_OUTPUT.PUT_LINE('Employee ID: ' || employee_rec.employee_id);
258     DBMS_OUTPUT.PUT_LINE('Employee Name: ' || employee_rec.full_name);
259     DBMS_OUTPUT.PUT_LINE('Job Title: ' || employee_rec.job_id);
260     DBMS_OUTPUT.PUT_LINE('Hire Date: ' || TO_CHAR(employee_rec.hire_date, 'DD-MON-YYYY'));
261     DBMS_OUTPUT.PUT_LINE('Salary: ' || employee_rec.salary);
262     DBMS_OUTPUT.PUT_LINE('-----');
```

The 'Results' tab is selected, displaying the output of the executed code. The output shows the details for employee ID 122:

```
Employee ID: 122
Employee Name: ODINSON LOKI
Job Title: SL REP
Hire Date: 30-OCT-2004
Salary: 5445
```

At the bottom of the interface, there are footer links for '220701014@rajalakshmi.edu.in', 'afrinfathima014', and 'en'. The copyright notice 'Copyright © 1999, 2023, Oracle and/or its affiliates.' and the version 'Oracle APEX 23.2.4' are also visible.

12. Write a PL/SQL program to display the employee IDs, names, and department names of all employees.

QUERY:

DECLARE

 CURSOR employee_cursor IS

```
    SELECT e.employee_id, e.first_name || ' ' || e.last_name AS full_name, d.dept_name
    FROM employees e
    INNER JOIN department d ON e.department_id = d.dept_id;
```

BEGIN

 -- Loop through the cursor and display employee information

 FOR employee_rec IN employee_cursor LOOP

```
    DBMS_OUTPUT.PUT_LINE('Employee ID: ' || employee_rec.employee_id);
    DBMS_OUTPUT.PUT_LINE('Employee Name: ' || employee_rec.full_name);
    DBMS_OUTPUT.PUT_LINE('Department Name: ' || employee_rec.dept_name);
    DBMS_OUTPUT.PUT_LINE('-----');
```

 END LOOP;

END;

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery. The user is logged in as 'Afrin Fathima' (af@frinfathima014). The SQL Commands tab is selected, showing the following PL/SQL code:

```
271 DECLARE
272   CURSOR employee_cursor IS
273     SELECT e.employee_id, e.first_name || ' ' || e.last_name AS full_name, d.dept_name
274     FROM employees e
275     INNER JOIN department d ON e.department_id = d.dept_id;
276 BEGIN
277   -- Loop through the cursor and display employee information
278   FOR employee_rec IN employee_cursor LOOP
279     DBMS_OUTPUT.PUT_LINE('Employee ID: ' || employee_rec.employee_id);
280     DBMS_OUTPUT.PUT_LINE('Employee Name: ' || employee_rec.full_name);
281     DBMS_OUTPUT.PUT_LINE('Department Name: ' || employee_rec.dept_name);
282     DBMS_OUTPUT.PUT_LINE('-----');
283   END LOOP;
```

The results section displays the output of the PL/SQL block:

```
Employee ID: 122
Employee Name: ODINSON LOKI
Department Name: MARKETING
-----
Employee ID: 109
Employee Name: AKAY KOHLI
Department Name: MANUFACTURING
```

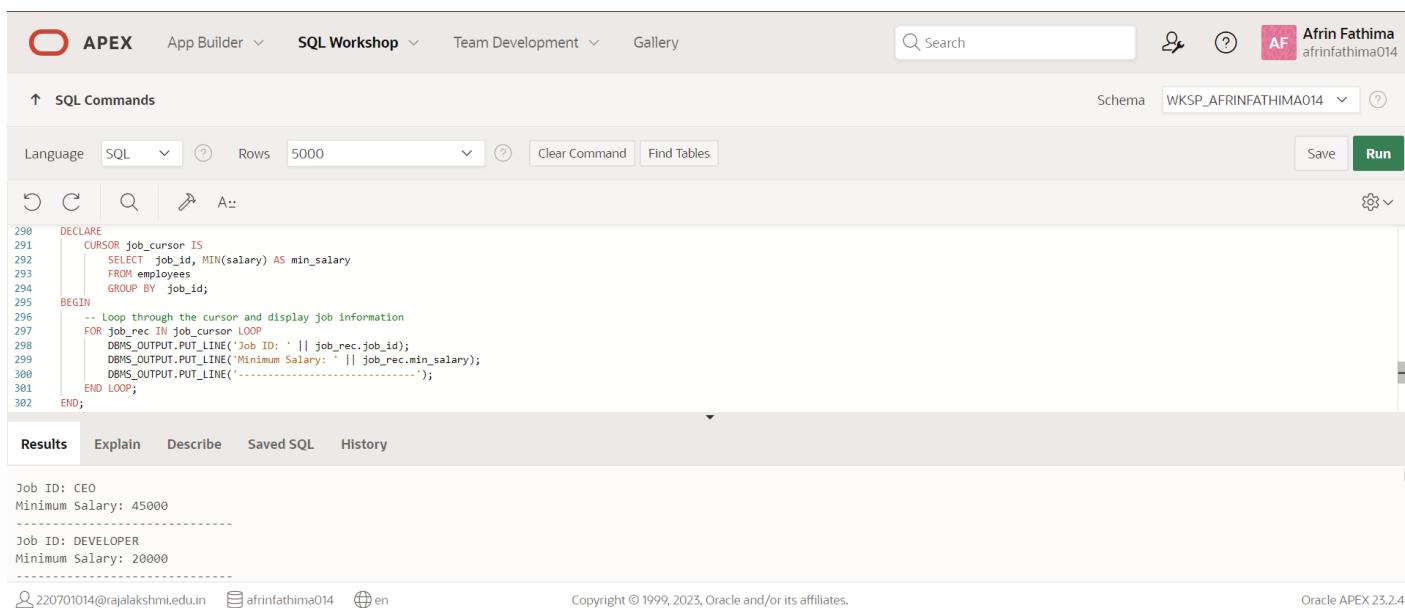
At the bottom, it shows the session details: 220701014@rajalakshmi.edu.in, af@frinfathima014, en, and the copyright notice: Copyright © 1999, 2023, Oracle and/or its affiliates. Oracle APEX 23.2.4.

13. Write a PL/SQL program to display the job IDs, titles, and minimum salaries of all jobs.

QUERY:

```
DECLARE
  CURSOR job_cursor IS
    SELECT job_id, MIN(salary) AS min_salary
    FROM employees
    GROUP BY job_id;
BEGIN
  -- Loop through the cursor and display job information
  FOR job_rec IN job_cursor LOOP
    DBMS_OUTPUT.PUT_LINE('Job ID: ' || job_rec.job_id);
    DBMS_OUTPUT.PUT_LINE('Minimum Salary: ' || job_rec.min_salary);
    DBMS_OUTPUT.PUT_LINE('-----');
  END LOOP;
END;
```

OUTPUT:



The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes APEX, App Builder, SQL Workshop (selected), Team Development, and Gallery. The right side shows the user profile 'Afrin Fathima' (afrafathima014). The main area is titled 'SQL Commands' with tabs for Language (SQL selected), Rows (5000), Clear Command, Find Tables, Save, and Run. The code editor contains the provided PL/SQL block. The results tab shows the output of the execution:

```
Job ID: CEO
Minimum Salary: 45000
-----
Job ID: DEVELOPER
Minimum Salary: 20000
```

14. Write a PL/SQL program to display the employee IDs, names, and job history start dates of all employees.

QUERY:

DECLARE

```
CURSOR employee_cursor IS
  SELECT e.employee_id, e.first_name || ' ' || e.last_name AS full_name, jh.start_date
  FROM employees e
  JOIN job_history jh ON e.employee_id = jh.employee_id;
```

BEGIN

```
-- Loop through the cursor and display employee information
```

```
FOR employee_rec IN employee_cursor LOOP
```

```
  DBMS_OUTPUT.PUT_LINE('Employee ID: ' || employee_rec.employee_id);
```

```
  DBMS_OUTPUT.PUT_LINE('Employee Name: ' || employee_rec.full_name);
```

```
  DBMS_OUTPUT.PUT_LINE('Job History Start Date: ' ||
```

```
    TO_CHAR(employee_rec.start_date, 'DD-MON-YYYY'));
```

```
  DBMS_OUTPUT.PUT_LINE('-----');
```

```
END LOOP;
```

```
END;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes 'APEX', 'App Builder', 'SQL Workshop' (selected), 'Team Development', and 'Gallery'. A search bar and user profile 'Afrin Fathima' are also present. The main workspace displays the PL/SQL code from the previous section. The code uses a cursor to select employee details and then loops through the results to print each employee's ID, name, and start date. The output pane at the bottom shows two rows of results:

Employee ID	Employee Name	Job History Start Date
103	SANJ PARTHI	04-DEC-1998
100	SAHANA JAY	05-JAN-1999

At the bottom of the page, footer links include 'Copyright © 1999, 2023, Oracle and/or its affiliates.', 'Oracle APEX 23.2.4', and the URL '220701014@rajalakshmi.edu.in'.

```
310 DECLARE
311   CURSOR employee_cursor IS
312     SELECT e.employee_id, e.first_name || ' ' || e.last_name AS full_name, jh.start_date
313     FROM employees e
314     JOIN job_history jh ON e.employee_id = jh.employee_id;
315 BEGIN
316   -- Loop through the cursor and display employee information
317   FOR employee_rec IN employee_cursor LOOP
318     DBMS_OUTPUT.PUT_LINE('Employee ID: ' || employee_rec.employee_id);
319     DBMS_OUTPUT.PUT_LINE('Employee Name: ' || employee_rec.full_name);
320     DBMS_OUTPUT.PUT_LINE('Job History Start Date: ' || TO_CHAR(employee_rec.start_date, 'DD-MON-YYYY'));
321     DBMS_OUTPUT.PUT_LINE('-----');
322   END LOOP;

```

Results Explain Describe Saved SQL History

```
Employee ID: 103
Employee Name: SANJ PARTHI
Job History Start Date: 04-DEC-1998
-----
Employee ID: 100
Employee Name: SAHANA JAY
Job History Start Date: 05-JAN-1999
```

15. Write a PL/SQL program to display the employee IDs, names, and job history end dates of all employees.

QUERY:

DECLARE

```
CURSOR employee_cursor IS
  SELECT e.employee_id, e.first_name || ' ' || e.last_name AS full_name, jh.end_date
  FROM employees e
  JOIN job_history jh ON e.employee_id = jh.employee_id;
```

BEGIN

-- Loop through the cursor and display employee information

FOR employee_rec IN employee_cursor LOOP

```
  DBMS_OUTPUT.PUT_LINE('Employee ID: ' || employee_rec.employee_id);
  DBMS_OUTPUT.PUT_LINE('Employee Name: ' || employee_rec.full_name);
```

-- Check if the end date is NULL (meaning the employee is currently in the job)

IF employee_rec.end_date IS NULL THEN

```
  DBMS_OUTPUT.PUT_LINE('Job History End Date: (Still Employed)');
```

ELSE

```
  DBMS_OUTPUT.PUT_LINE('Job History End Date: ' ||
```

TO_CHAR(employee_rec.end_date, 'DD-MON-YYYY'));

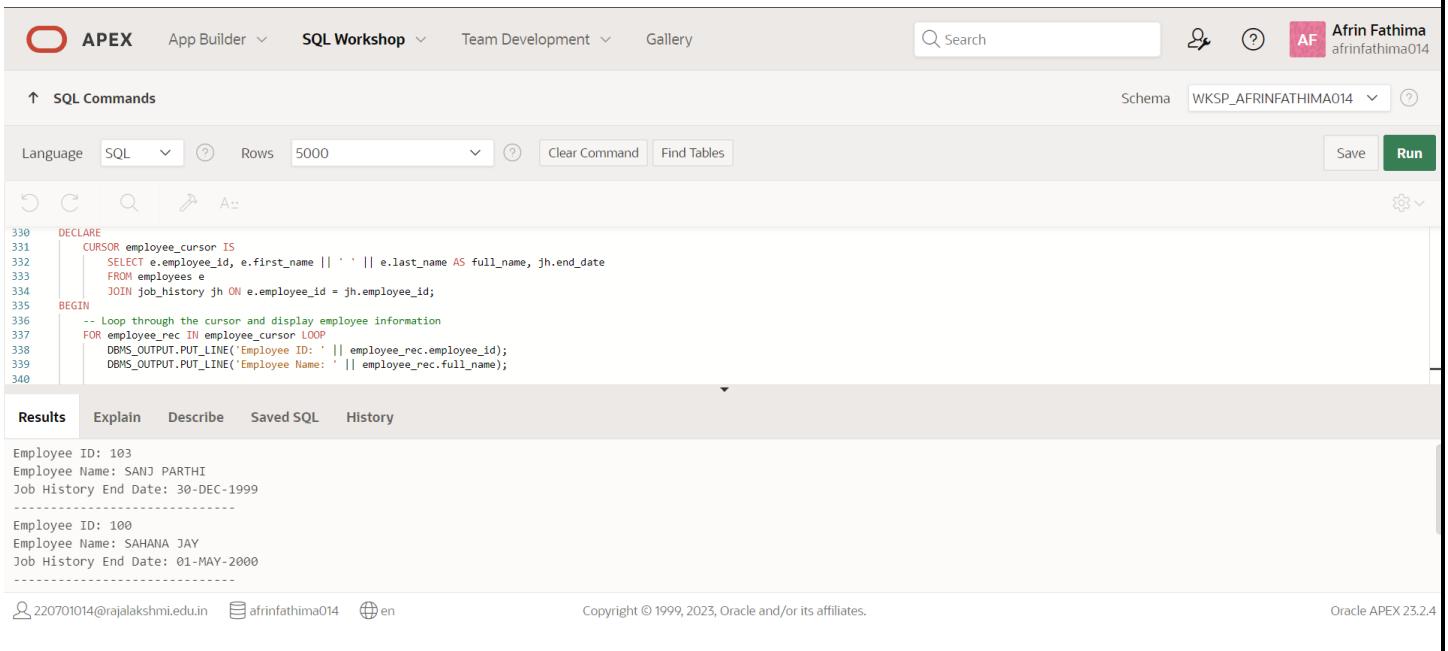
END IF;

```
  DBMS_OUTPUT.PUT_LINE('-----');
```

END LOOP;

END;

OUTPUT:



```
330  DECLARE
331    CURSOR employee_cursor IS
332      SELECT e.employee_id, e.first_name || ' ' || e.last_name AS full_name, jh.end_date
333      FROM employees e
334      JOIN job_history jh ON e.employee_id = jh.employee_id;
335  BEGIN
336    -- Loop through the cursor and display employee information
337    FOR employee_rec IN employee_cursor LOOP
338      DBMS_OUTPUT.PUT_LINE('Employee ID: ' || employee_rec.employee_id);
339      DBMS_OUTPUT.PUT_LINE('Employee Name: ' || employee_rec.full_name);
340
Employee ID: 103
Employee Name: SANJ PARTHI
Job History End Date: 30-DEC-1999
-----
Employee ID: 100
Employee Name: SAHANA JAY
Job History End Date: 01-MAY-2000
```

Evaluation Procedure	Marks awarded
Query(5)	
Execution (5)	
Viva(5)	
Total (15)	
Faculty Signature	

RESULT :

PROCEDURES AND FUNCTIONS

EX. NO: 17

DATE:

1.) Factorial of a number using function.

QUERY:

```
DECLARE
    fac NUMBER := 1;
    n NUMBER := :1;
BEGIN
    WHILE n > 0 LOOP
        fac := n * fac;
        n := n - 1;
    END LOOP;
    DBMS_OUTPUT.PUT_LINE(fac);
END;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop (selected), Team Development, and Gallery. The right side shows the user profile 'Afrin Fathima' and the schema 'WKSP_AFRINFATHIMA014'. The main workspace is titled 'SQL Commands' and contains the following PL/SQL code:

```
1  DECLARE
2      fac NUMBER := 1;
3      n NUMBER := :1;
4  BEGIN
5      WHILE n > 0 LOOP
6          fac := n * fac;
7          n := n - 1;
8      END LOOP;
9      DBMS_OUTPUT.PUT_LINE(fac);
10 END;
```

Below the code, there are tabs for 'Results', 'Explain', 'Describe', 'Saved SQL', and 'History'. The 'Results' tab displays the output:

```
120
Statement processed.
```

At the bottom, footer information includes email addresses (220701014@rajalakshmi.edu.in, afrinfathima014), a language setting (en), copyright notice (Copyright © 1999, 2023, Oracle and/or its affiliates.), and the software version (Oracle APEX 23.2.4).

2.) Write a PL/SQL program using Procedures IN,INOUT,OUT parameters to retrieve the corresponding book information in library.

QUERY:

```
CREATE OR REPLACE PROCEDURE get_book_info (
    p_book_id IN NUMBER,
    p_title IN OUT VARCHAR2,
    p_author OUT VARCHAR2,
    p_year_published OUT NUMBER
)
AS
BEGIN
    SELECT title, author, year_published INTO p_title, p_author, p_year_published
    FROM books
    WHERE book_id = p_book_id;
```

```
    p_title := p_title || ' - Retrieved';
EXCEPTION
    WHEN NO_DATA_FOUND THEN
        p_title := NULL;
        p_author := NULL;
        p_year_published := NULL;
END;
```

```
DECLARE
    v_book_id NUMBER := 1;
    v_title VARCHAR2(100);
    v_author VARCHAR2(100);
    v_year_published NUMBER;
BEGIN
    v_title := 'Initial Title';

    get_book_info(p_book_id => v_book_id, p_title => v_title, p_author => v_author,
    p_year_published => v_year_published);

    DBMS_OUTPUT.PUT_LINE('Title: ' || v_title);
    DBMS_OUTPUT.PUT_LINE('Author: ' || v_author);
    DBMS_OUTPUT.PUT_LINE('Year Published: ' || v_year_published);
END;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. At the top, there are tabs for App Builder, SQL Workshop (selected), Team Development, and Gallery. A search bar and user information for 'Afrin Fathima' are also at the top. The main area is titled 'SQL Commands' and contains the following PL/SQL code:

```
27
28 CREATE OR REPLACE PROCEDURE get_book_info (
29   p_book_id IN NUMBER,
30   p_title IN OUT VARCHAR2,
31   p_author OUT VARCHAR2,
32   p_year_published OUT NUMBER
33 )
34 AS
35 BEGIN
36   SELECT title, author, year_published INTO p_title, p_author, p_year_published
37   FROM books
38   WHERE book_id = p_book_id;
39 
```

Below the code, there are tabs for Results (selected), Explain, Describe, Saved SQL, and History. The results section displays the output of the query:

Title: 1984 - Retrieved
Author: George Orwell
Year Published: 1949
Statement processed.

At the bottom, there are footer links for 220701014@rajalakshmi.edu.in, afrinfathima014, en, Copyright © 1999, 2023, Oracle and/or its affiliates, and Oracle APEX 23.2.4.

Evaluation Procedure	Marks awarded
Query(5)	
Execution (5)	
Viva(5)	
Total (15)	
Faculty Signature	

RESULT:

TRIGGER

EX_NO: 18

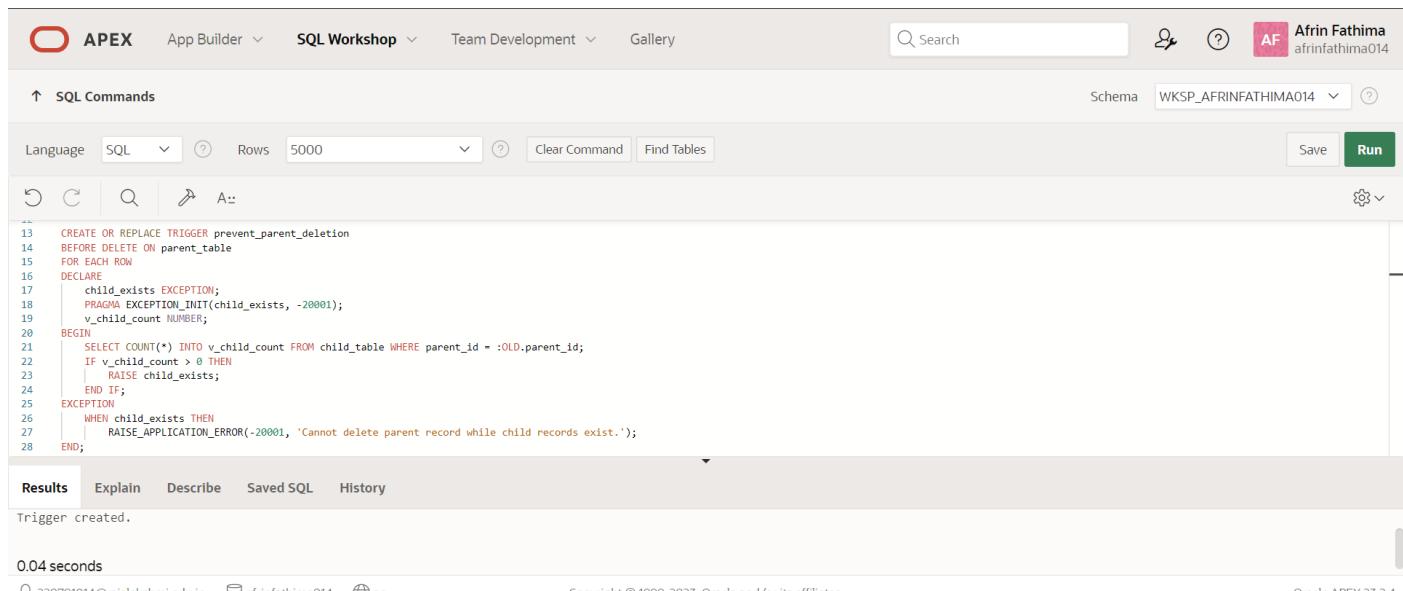
DATE:

1.) Write a code in PL/SQL to develop a trigger that enforces referential integrity by preventing the deletion of a parent record if child records exist

QUERY:

```
CREATE OR REPLACE TRIGGER prevent_parent_deletion
BEFORE DELETE ON parent_table
FOR EACH ROW
DECLARE
    child_exists EXCEPTION;
    PRAGMA EXCEPTION_INIT(child_exists, -20001);
    v_child_count NUMBER;
BEGIN
    SELECT COUNT(*) INTO v_child_count FROM child_table WHERE parent_id =
:OLD.parent_id;
    IF v_child_count > 0 THEN
        RAISE child_exists;
    END IF;
EXCEPTION
    WHEN child_exists THEN
        RAISE_APPLICATION_ERROR(-20001, 'Cannot delete parent record while child
records exist.');
END;
```

OUTPUT:



The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop (selected), Team Development, and Gallery. A search bar and user profile 'Afrin Fathima' are also present. The main workspace is titled 'SQL Commands' and contains the PL/SQL code for the trigger. The code is highlighted in blue and red. Below the code, there are tabs for 'Results', 'Explain', 'Describe', 'Saved SQL', and 'History'. The results section shows the message 'Trigger created.' and a execution time of '0.04 seconds'. The bottom footer includes copyright information for Oracle and the APEX version.

```
13 CREATE OR REPLACE TRIGGER prevent_parent_deletion
14 BEFORE DELETE ON parent_table
15 FOR EACH ROW
16 DECLARE
17     child_exists EXCEPTION;
18     PRAGMA EXCEPTION_INIT(child_exists, -20001);
19     v_child_count NUMBER;
20 BEGIN
21     SELECT COUNT(*) INTO v_child_count FROM child_table WHERE parent_id = :OLD.parent_id;
22     IF v_child_count > 0 THEN
23         RAISE child_exists;
24     END IF;
25 EXCEPTION
26     WHEN child_exists THEN
27         RAISE_APPLICATION_ERROR(-20001, 'Cannot delete parent record while child records exist.');
28 END;
```

2.) Write a code in PL/SQL to create a trigger that checks for duplicate values in a specific column and raises an exception if found

QUERY:

```
CREATE OR REPLACE TRIGGER check_duplicates
BEFORE INSERT OR UPDATE ON unique_values_table
FOR EACH ROW
DECLARE
    duplicate_found EXCEPTION;
    PRAGMA EXCEPTION_INIT(duplicate_found, -20002);
    v_count NUMBER;
BEGIN
    SELECT COUNT(*) INTO v_count FROM unique_values_table
    WHERE unique_col = :NEW.unique_col AND id != :NEW.id;
    IF v_count > 0 THEN
        RAISE duplicate_found;
    END IF;
EXCEPTION
    WHEN duplicate_found THEN
        RAISE_APPLICATION_ERROR(-20002, 'Duplicate value found in unique_col.');
END;
```

OUTPUT:

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes 'APEX', 'App Builder', 'SQL Workshop', 'Team Development', and 'Gallery'. The 'SQL Workshop' tab is active. The right side of the header shows the user 'Afrin Fathima' with the ID 'afarinfathima014'. Below the header, the schema 'WKSP_AFRINFATHIMA014' is selected. The main workspace is titled 'SQL Commands' and contains the PL/SQL code for the trigger. The code is highlighted in red and blue, indicating syntax. The bottom of the workspace shows the results of the command execution, which includes the message 'Trigger created.' and a timing of '0.04 seconds'. The footer of the page includes copyright information for Oracle and the APEX version '23.2.4'.

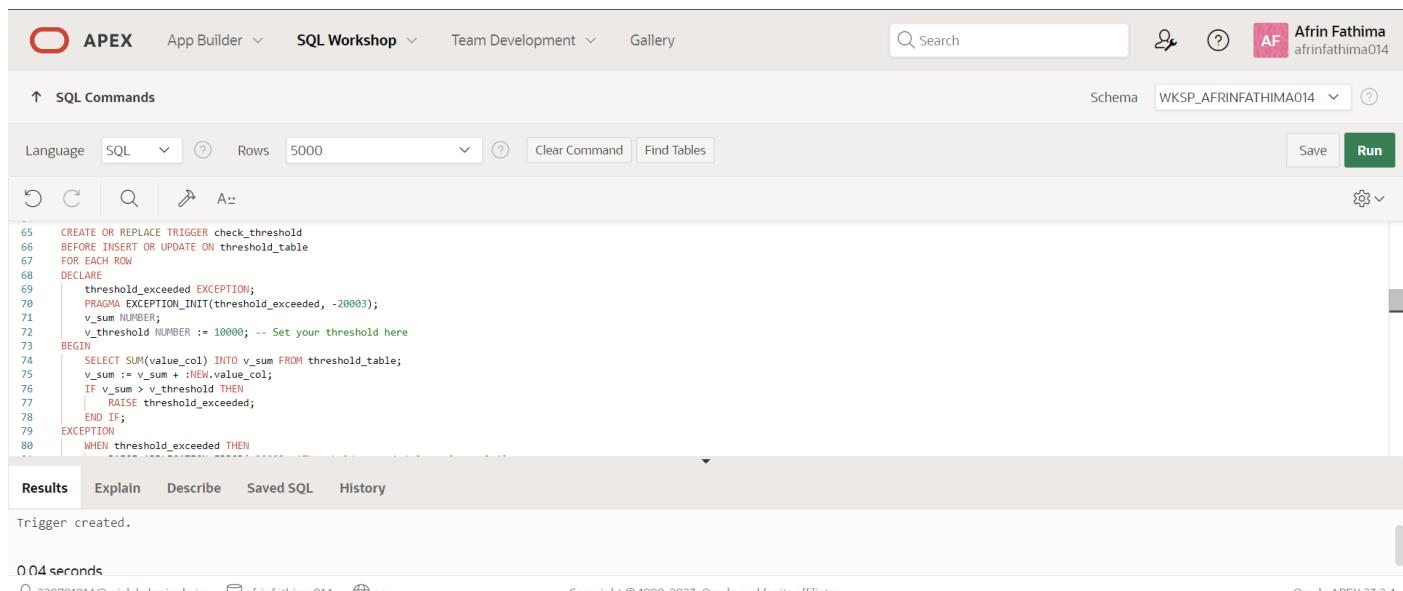
```
40 CREATE OR REPLACE TRIGGER check_duplicates
41 BEFORE INSERT OR UPDATE ON unique_values_table
42 FOR EACH ROW
43 DECLARE
44     duplicate_found EXCEPTION;
45     PRAGMA EXCEPTION_INIT(duplicate_found, -20002);
46     v_count NUMBER;
47 BEGIN
48     SELECT COUNT(*) INTO v_count FROM unique_values_table
49     WHERE unique_col = :NEW.unique_col AND id != :NEW.id;
50     IF v_count > 0 THEN
51         RAISE duplicate_found;
52     END IF;
53 EXCEPTION
54     WHEN duplicate_found THEN
55         RAISE_APPLICATION_ERROR(-20002, 'Duplicate value found in unique_col.');
56 END.
```

3.) Write a code in PL/SQL to create a trigger that restricts the insertion of new rows if the total of a column's values exceeds a certain threshold

QUERY:

```
CREATE OR REPLACE TRIGGER check_threshold
BEFORE INSERT OR UPDATE ON threshold_table
FOR EACH ROW
DECLARE
    threshold_exceeded EXCEPTION;
    PRAGMA EXCEPTION_INIT(threshold_exceeded, -20003);
    v_sum NUMBER;
    v_threshold NUMBER := 10000; -- Set your threshold here
BEGIN
    SELECT SUM(value_col) INTO v_sum FROM threshold_table;
    v_sum := v_sum + :NEW.value_col;
    IF v_sum > v_threshold THEN
        RAISE threshold_exceeded;
    END IF;
EXCEPTION
    WHEN threshold_exceeded THEN
        RAISE_APPLICATION_ERROR(-20003, 'Threshold exceeded for value_col.');
END;
```

OUTPUT:



The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery. A search bar and user profile 'Afrin Fathima' are also present. The main workspace is titled 'SQL Commands' and contains the PL/SQL code for the trigger. The code is numbered from 65 to 88. The 'Run' button is visible at the bottom right of the code area. Below the code, the 'Results' tab is selected, showing the message 'Trigger created.' The status bar at the bottom indicates the operation took '0.04 seconds'. The bottom right corner shows the copyright notice 'Copyright © 1999, 2023, Oracle and/or its affiliates.' and the version 'Oracle APEX 23.2.4'.

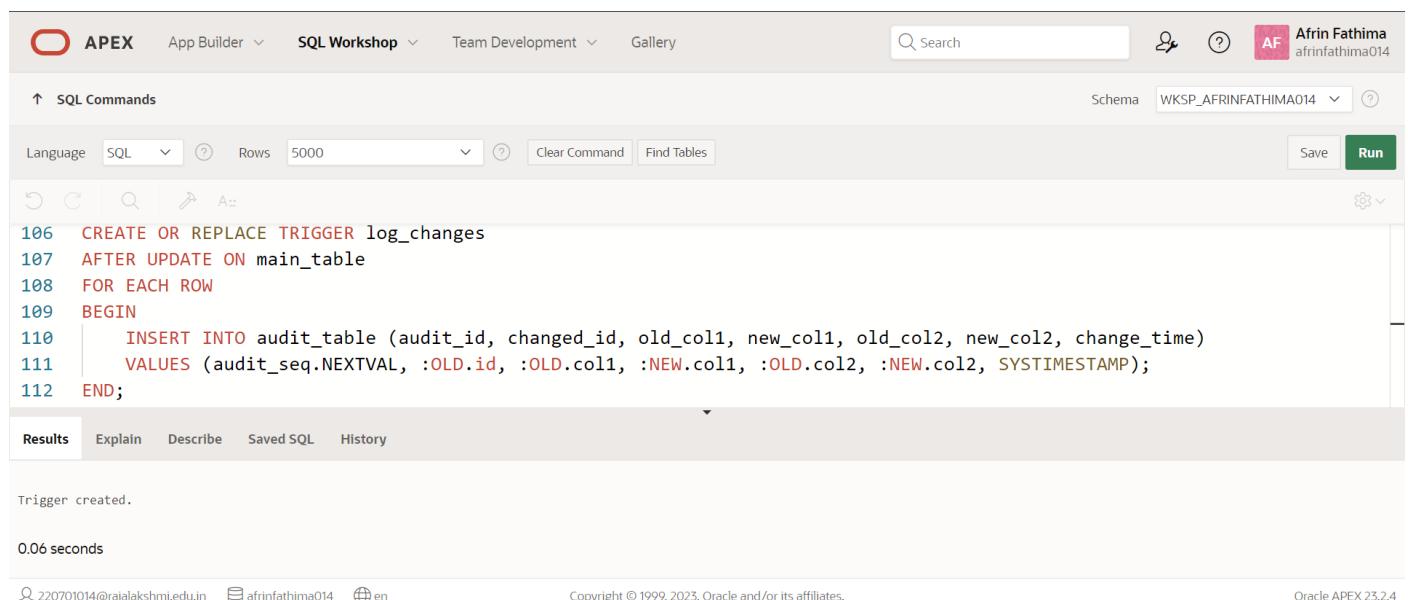
```
65 CREATE OR REPLACE TRIGGER check_threshold
66 BEFORE INSERT OR UPDATE ON threshold_table
67 FOR EACH ROW
68 DECLARE
69     threshold_exceeded EXCEPTION;
70     PRAGMA EXCEPTION_INIT(threshold_exceeded, -20003);
71     v_sum NUMBER;
72     v_threshold NUMBER := 10000; -- Set your threshold here
73 BEGIN
74     SELECT SUM(value_col) INTO v_sum FROM threshold_table;
75     v_sum := v_sum + :NEW.value_col;
76     IF v_sum > v_threshold THEN
77         RAISE threshold_exceeded;
78     END IF;
79 EXCEPTION
80     WHEN threshold_exceeded THEN
```

4.) Write a code in PL/SQL to design a trigger that captures changes made to specific columns and logs them in an audit table.

QUERY:

```
CREATE OR REPLACE TRIGGER log_changes
AFTER UPDATE ON main_table
FOR EACH ROW
BEGIN
    INSERT INTO audit_table (audit_id, changed_id, old_col1, new_col1, old_col2,
new_col2, change_time)
    VALUES (audit_seq.NEXTVAL, :OLD.id, :OLD.col1, :NEW.col1, :OLD.col2,
:NEW.col2, SYSTIMESTAMP);
END;
```

OUTPUT:



The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery. A search bar and user profile 'Afrin Fathima' are also present. The main workspace displays the SQL command for creating the 'log_changes' trigger. The code is as follows:

```
106 CREATE OR REPLACE TRIGGER log_changes
107 AFTER UPDATE ON main_table
108 FOR EACH ROW
109 BEGIN
110     INSERT INTO audit_table (audit_id, changed_id, old_col1, new_col1, old_col2, new_col2, change_time)
111     VALUES (audit_seq.NEXTVAL, :OLD.id, :OLD.col1, :NEW.col1, :OLD.col2, :NEW.col2, SYSTIMESTAMP);
112 END;
```

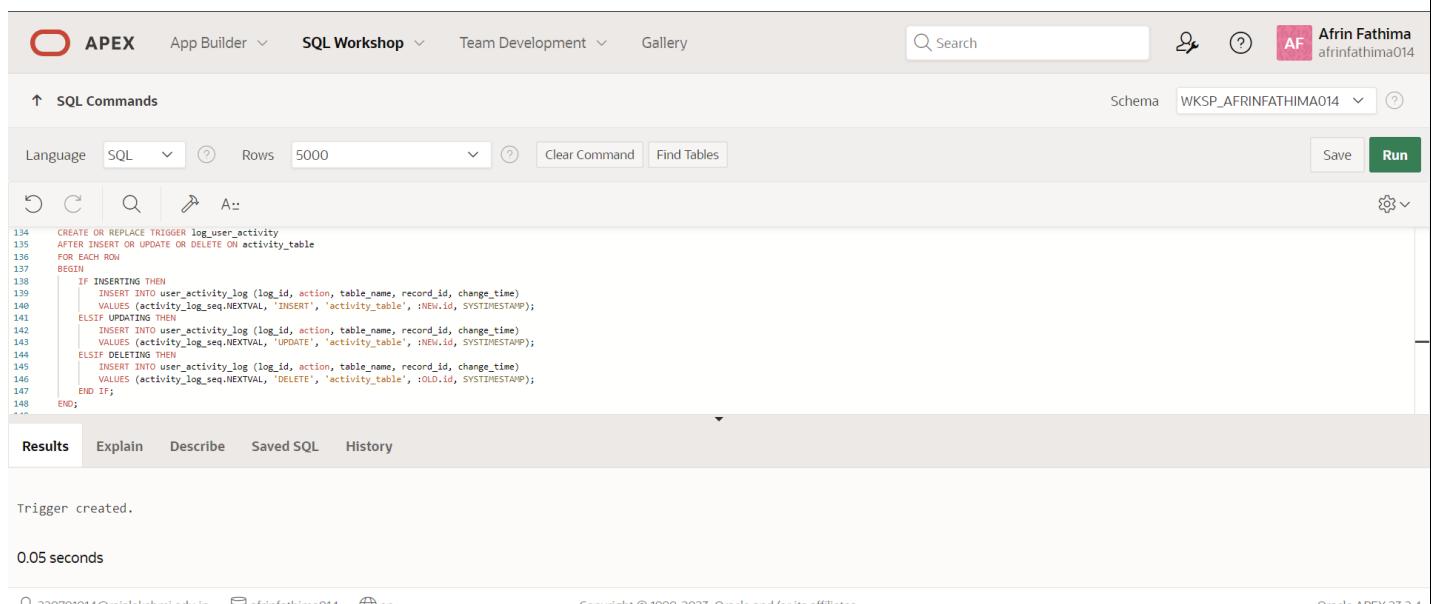
The 'Results' tab is selected at the bottom, showing the message "Trigger created." and a execution time of "0.06 seconds". The footer includes copyright information for Oracle and the APEX version "Oracle APEX 23.2.4".

5.) Write a code in PL/SQL to implement a trigger that records user activity (inserts, updates, deletes) in an audit log for a given set of tables.

QUERY:

```
CREATE OR REPLACE TRIGGER log_user_activity
AFTER INSERT OR UPDATE OR DELETE ON activity_table
FOR EACH ROW
BEGIN
    IF INSERTING THEN
        INSERT INTO user_activity_log (log_id, action, table_name, record_id, change_time)
        VALUES (activity_log_seq.NEXTVAL, 'INSERT', 'activity_table', :NEW.id,
SYSTIMESTAMP);
    ELSIF UPDATING THEN
        INSERT INTO user_activity_log (log_id, action, table_name, record_id, change_time)
        VALUES (activity_log_seq.NEXTVAL, 'UPDATE', 'activity_table', :NEW.id,
SYSTIMESTAMP);
    ELSIF DELETING THEN
        INSERT INTO user_activity_log (log_id, action, table_name, record_id, change_time)
        VALUES (activity_log_seq.NEXTVAL, 'DELETE', 'activity_table', :OLD.id,
SYSTIMESTAMP);
    END IF;
END;
```

OUTPUT:



The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery. A search bar and a user profile for 'Afrin Fathima' are also present. The main workspace is titled 'SQL Commands'. The schema dropdown is set to 'WKSP_AFRINFATHIMA014'. The SQL editor contains the PL/SQL code for the trigger, with line numbers 134 through 148 visible. The code implements the specified logic for inserting, updating, and deleting rows from the 'activity_table'. The bottom section of the interface shows the 'Results' tab, which displays the message 'Trigger created.' and a execution time of '0.05 seconds'. The footer includes copyright information for Oracle and the APEX version '23.2.4'.

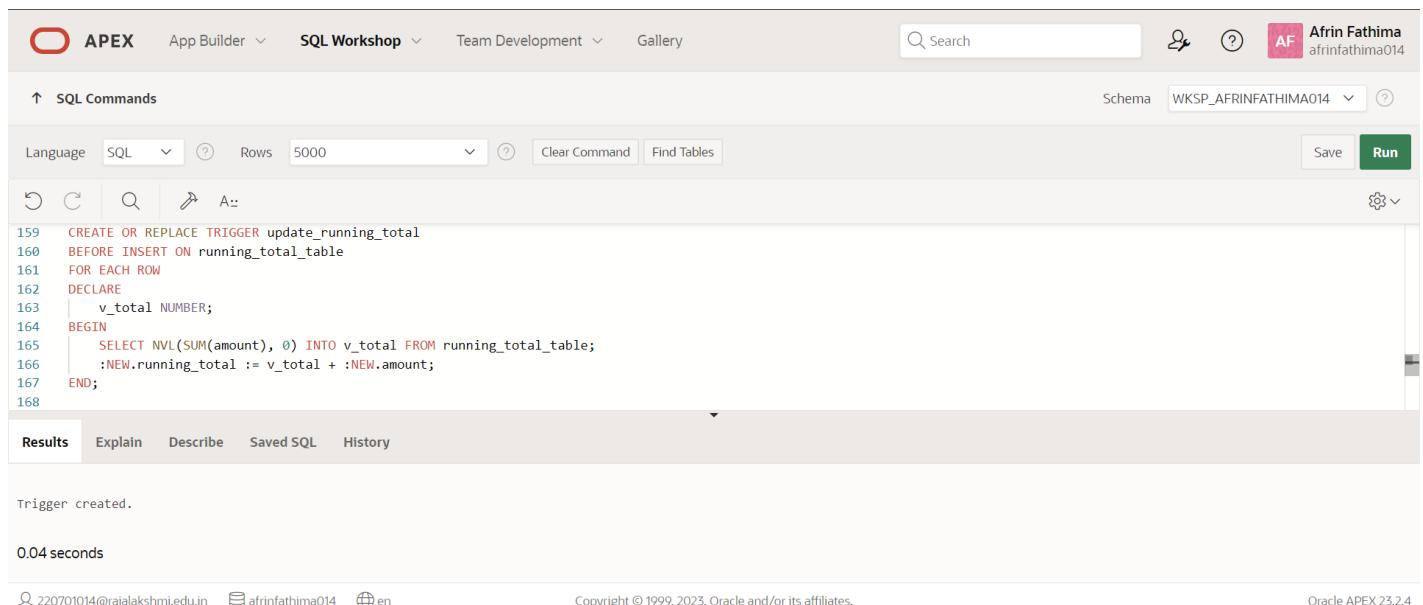
```
134 CREATE OR REPLACE TRIGGER log_user_activity
135     AFTER INSERT OR UPDATE OR DELETE ON activity_table
136     FOR EACH ROW
137 BEGIN
138     IF INSERTING THEN
139         INSERT INTO user_activity_log (log_id, action, table_name, record_id, change_time)
140         VALUES (activity_log_seq.NEXTVAL, 'INSERT', 'activity_table', :NEW.id, SYSTIMESTAMP);
141     ELSIF UPDATING THEN
142         INSERT INTO user_activity_log (log_id, action, table_name, record_id, change_time)
143         VALUES (activity_log_seq.NEXTVAL, 'UPDATE', 'activity_table', :NEW.id, SYSTIMESTAMP);
144     ELSIF DELETING THEN
145         INSERT INTO user_activity_log (log_id, action, table_name, record_id, change_time)
146         VALUES (activity_log_seq.NEXTVAL, 'DELETE', 'activity_table', :OLD.id, SYSTIMESTAMP);
147     END IF;
148 END;
```

6.) Write a code in PL/SQL to implement a trigger that automatically calculates and updates a running total column for a table whenever new rows are inserted

QUERY:

```
CREATE OR REPLACE TRIGGER update_running_total
BEFORE INSERT ON running_total_table
FOR EACH ROW
DECLARE
    v_total NUMBER;
BEGIN
    SELECT NVL(SUM(amount), 0) INTO v_total FROM running_total_table;
    :NEW.running_total := v_total + :NEW.amount;
END;
```

OUTPUT:



The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for APEX, App Builder, SQL Workshop, Team Development, and Gallery. A search bar is at the top right. The main area is titled "SQL Commands". The schema dropdown is set to "WKSP_AFRINFATHIMA014". The SQL editor contains the PL/SQL code for the trigger. The code is numbered from 159 to 168. The "Run" button is visible at the bottom right of the editor. Below the editor, there are tabs for Results, Explain, Describe, Saved SQL, and History. The results tab shows the output: "Trigger created." and "0.04 seconds". The bottom of the screen displays user information and copyright details.

```
159 CREATE OR REPLACE TRIGGER update_running_total
160 BEFORE INSERT ON running_total_table
161 FOR EACH ROW
162 DECLARE
163     v_total NUMBER;
164 BEGIN
165     SELECT NVL(SUM(amount), 0) INTO v_total FROM running_total_table;
166     :NEW.running_total := v_total + :NEW.amount;
167 END;
168
```

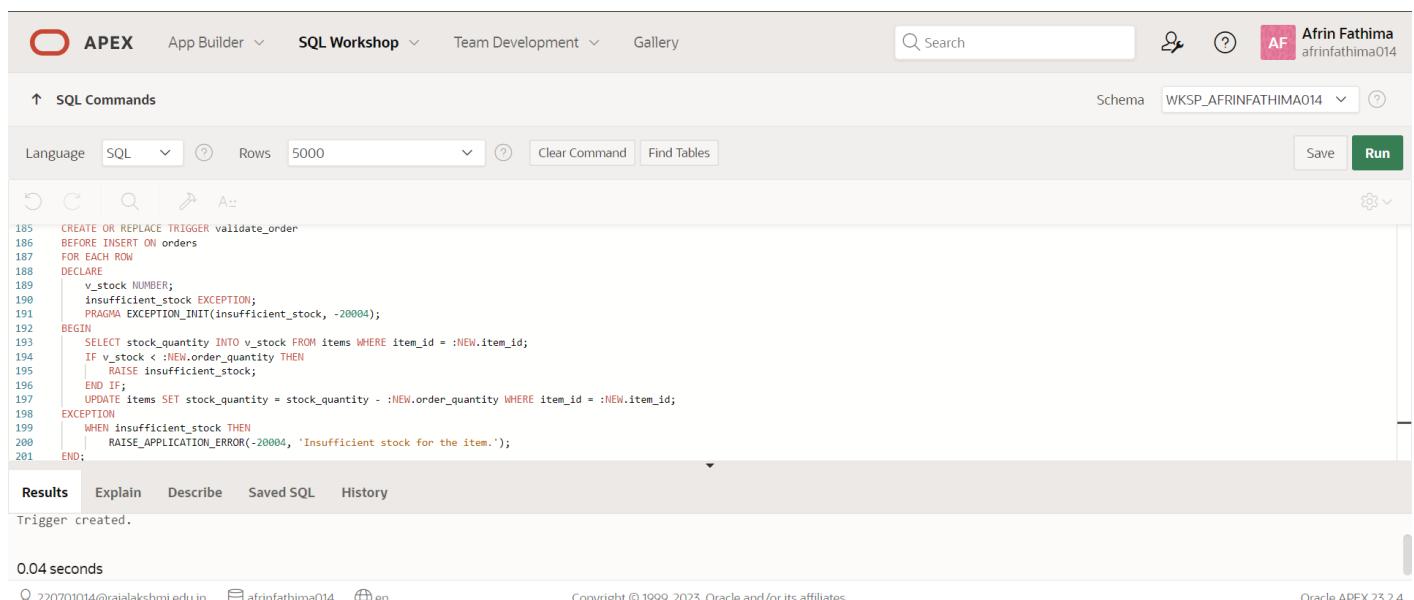
Trigger created.
0.04 seconds

7.) Write a code in PL/SQL to create a trigger that validates the availability of items before allowing an order to be placed, considering stock levels and pending orders

QUERY:

```
CREATE OR REPLACE TRIGGER validate_order
BEFORE INSERT ON orders
FOR EACH ROW
DECLARE
    v_stock NUMBER;
    insufficient_stock EXCEPTION;
    PRAGMA EXCEPTION_INIT(insufficient_stock, -20004);
BEGIN
    SELECT stock_quantity INTO v_stock FROM items WHERE item_id = :NEW.item_id;
    IF v_stock < :NEW.order_quantity THEN
        RAISE insufficient_stock;
    END IF;
    UPDATE items SET stock_quantity = stock_quantity - :NEW.order_quantity WHERE
item_id = :NEW.item_id;
EXCEPTION
    WHEN insufficient_stock THEN
        RAISE_APPLICATION_ERROR(-20004, 'Insufficient stock for the item.');
END;
```

OUTPUT:



The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes 'APEX', 'App Builder', 'SQL Workshop' (selected), 'Team Development', 'Gallery', 'Search', and a user profile 'Afrin Fathima'. The main workspace is titled 'SQL Commands' and contains the PL/SQL code for the trigger. The code is highlighted in red and black, indicating syntax. The bottom section shows the 'Results' tab with the message 'Trigger created.' and a timestamp of '0.04 seconds'. The footer includes copyright information for Oracle and the APEX version '23.2.4'.

```
185 CREATE OR REPLACE TRIGGER validate_order
186 BEFORE INSERT ON orders
187 FOR EACH ROW
188 DECLARE
189     v_stock NUMBER;
190     insufficient_stock EXCEPTION;
191     PRAGMA EXCEPTION_INIT(insufficient_stock, -20004);
192 BEGIN
193     SELECT stock_quantity INTO v_stock FROM items WHERE item_id = :NEW.item_id;
194     IF v_stock < :NEW.order_quantity THEN
195         RAISE insufficient_stock;
196     END IF;
197     UPDATE items SET stock_quantity = stock_quantity - :NEW.order_quantity WHERE
item_id = :NEW.item_id;
198 EXCEPTION
199     WHEN insufficient_stock THEN
200         RAISE_APPLICATION_ERROR(-20004, 'Insufficient stock for the item.');
201 END;
```

Evaluation Procedure	Marks awarded
Query(5)	
Execution (5)	
Viva(5)	
Total (15)	
Faculty Signature	

RESULT:

MONGO DB

EX_NO: 19

DATE:

1.) Write a MongoDB query to find the restaurant Id, name, borough and cuisine for those restaurants which prepared dish except 'American' and 'Chinees' or restaurant's name begins with letter 'Wil'.

QUERY:

```
db.restaurants.find(
{
  $or: [
    { name: /^Wil/ },
    { cuisine: { $nin: ['American', 'Chinese'] } }
  ]
},
{
  restaurant_id: 1,
  name: 1,
  borough: 1,
  cuisine: 1
}
);
```

OUTPUT:

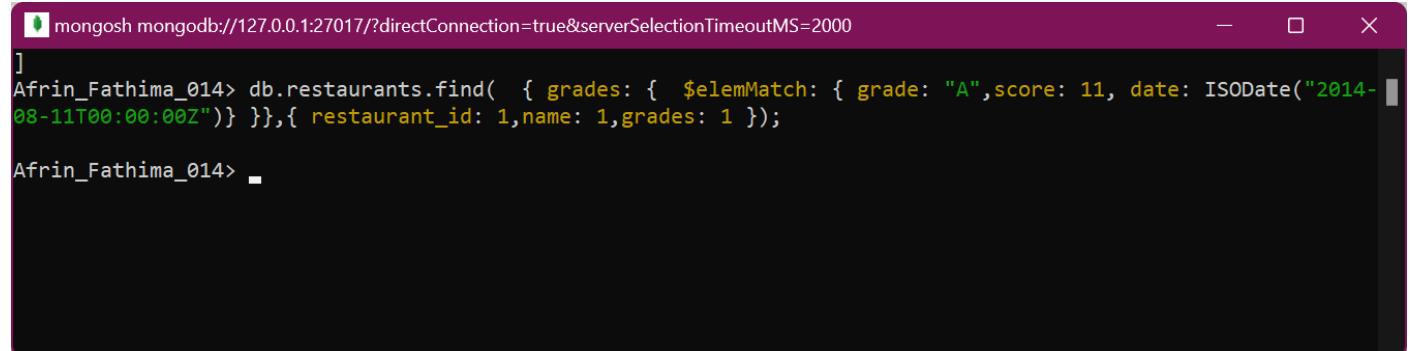
```
Afrin_Fathima_014> db.restaurants.find(
...   {
...     $or: [
...       { name: /^Wil/ },
...       { cuisine: { $nin: ['American', 'Chinese'] } }
...     ]
...   },
...   {
...     restaurant_id: 1,
...     name: 1,
...     borough: 1,
...     cuisine: 1
...   }
... );
[
  {
    _id: ObjectId('564c2d949eb21ad392f1d6de'),
    borough: 'Manhattan',
    cuisine: 'Other',
    name: '',
    restaurant_id: '50017887'
  },
  {
    _id: ObjectId('564c2d949eb21ad392f1d6ec'),
    borough: 'Brooklyn',
    cuisine: 'Other',
    name: '',
    restaurant_id: '50017910'
  },
  {
    _id: ObjectId('564c2d949eb21ad392f1d6ed'),
    borough: 'Manhattan',
    cuisine: 'Other',
    name: '',
    restaurant_id: '50017912'
  }
]
```

2.) Write a MongoDB query to find the restaurant Id, name, and grades for those restaurants which achieved a grade of "A" and scored 11 on an ISODate "2014-08-11T00:00:00Z" among many of survey dates.

QUERY:

```
db.restaurants.find( { grades: { $elemMatch: { grade: "A", score: 11, date: ISODate("2014-08-11T00:00:00Z") } } }, { restaurant_id: 1, name: 1, grades: 1 } );
```

OUTPUT:



A screenshot of a terminal window titled "mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000". The window contains the following text:

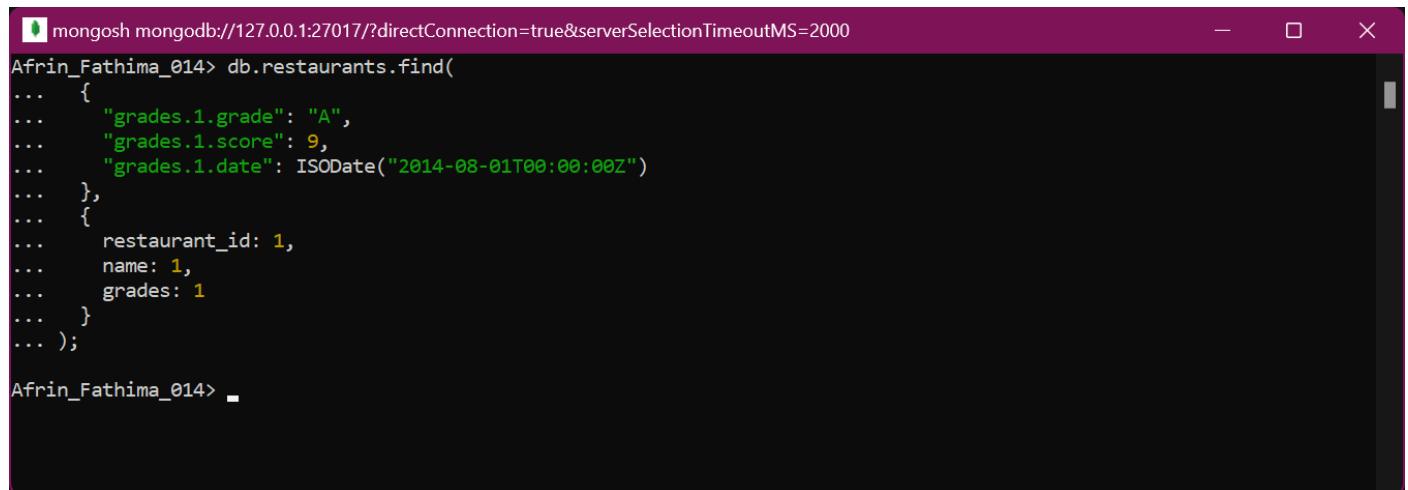
```
[Afrin_Fathima_014] > db.restaurants.find( { grades: { $elemMatch: { grade: "A", score: 11, date: ISODate("2014-08-11T00:00:00Z") } } }, { restaurant_id: 1, name: 1, grades: 1 } );
```

3.) Write a MongoDB query to find the restaurant Id, name and grades for those restaurants where the 2nd element of grades array contains a grade of "A" and score 9 on an ISODate "2014-08-11T00:00:00Z".

QUERY:

```
db.restaurants.find(  
  {  
    "grades.1.grade": "A",  
    "grades.1.score": 9,  
    "grades.1.date": ISODate("2014-08-01T00:00:00Z")  
  },  
  {  
    restaurant_id: 1,  
    name: 1,  
    grades: 1  
  }  
);
```

OUTPUT:



A screenshot of a terminal window titled 'mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000'. The window shows a command being typed in the mongo shell:

```
Afrin_Fathima_014> db.restaurants.find(  
...   {  
...     "grades.1.grade": "A",  
...     "grades.1.score": 9,  
...     "grades.1.date": ISODate("2014-08-01T00:00:00Z")  
...   },  
...   {  
...     restaurant_id: 1,  
...     name: 1,  
...     grades: 1  
...   }  
... );
```

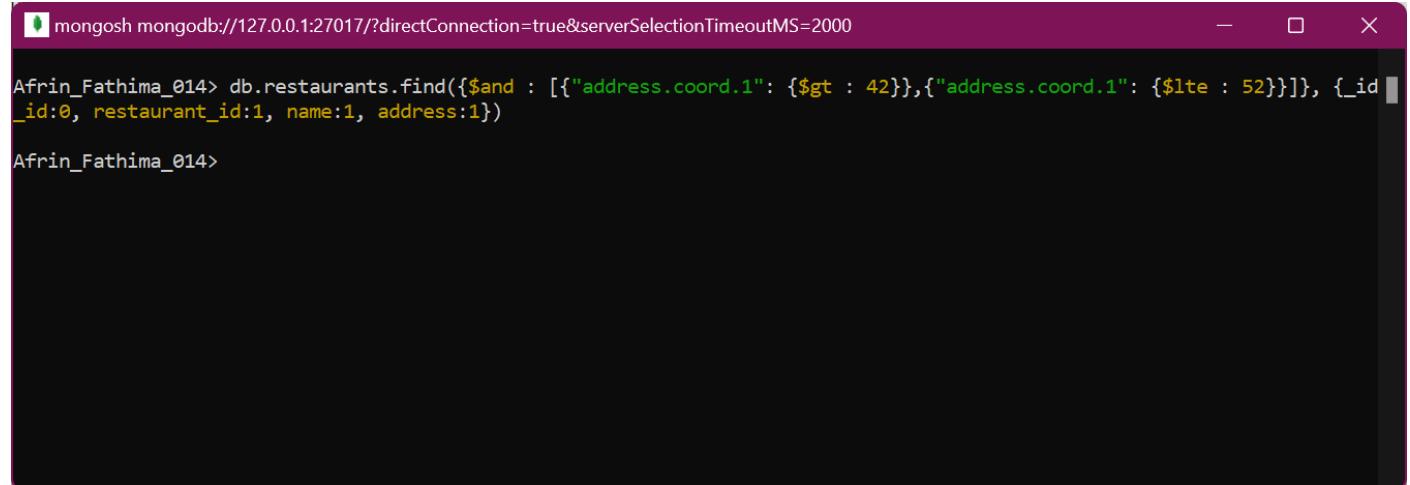
The command is incomplete, ending with a final semicolon.

4.) Write a MongoDB query to find the restaurant Id, name, address and geographical location for those restaurants where 2nd element of coord array contains a value which is more than 42 and upto 52

QUERY:

```
db.restaurants.find({$and : [{"address.coord.1": {$gt : 42}}, {"address.coord.1": {$lte : 52}}]}, {_id:0, restaurant_id:1, name:1, address:1})
```

OUTPUT:



The screenshot shows a terminal window titled "mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000". The command entered is:

```
Afrin_Fathima_014> db.restaurants.find({$and : [{"address.coord.1": {$gt : 42}}, {"address.coord.1": {$lte : 52}}]}, {_id:0, restaurant_id:1, name:1, address:1})
```

The output of the command is shown below the command line.

5.) Write a MongoDB query to arrange the name of the restaurants in ascending order along with all the columns.

QUERY:

```
db.restaurants.find({}, { _id: 0 }).sort({ name: 1 });
```

OUTPUT:

```
[mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
Afrin_Fathima_014> db.restaurants.find({}, { _id: 0 }).sort({ name: 1 });
[
  {
    address: {
      building: '154',
      coord: [ -73.9189064, 40.8654529 ],
      street: 'Post Ave',
      zipcode: '10034'
    },
    borough: 'Manhattan',
    cuisine: 'Other',
    grades: [],
    name: '',
    restaurant_id: '50017887'
  },
  {
    address: {
      building: '508',
      coord: [ -73.999813, 40.683876 ],
      street: 'Henry St',
      zipcode: '11231'
    },
    borough: 'Brooklyn',
    cuisine: 'Other',
    grades: [],
    name: '',
    restaurant_id: '50017910'
  },
  {
    address: {
      building: '15',
      coord: [ -73.9966882, 40.7139264 ],
      street: 'Division St',
      zipcode: '10002'
    },
    borough: 'Manhattan',
    cuisine: 'Other',
    grades: []
  }
]
```

6.) Write a MongoDB query to arrange the name of the restaurants in descending along with all the columns.

QUERY:

```
db.restaurants.find({}, { _id: 0 }).sort({ name: -1 })
```

OUTPUT:

```
Afrin_Fathima_014> db.restaurants.find({}, { _id: 0 }).sort({ name: -1 })
[ { address: {
    building: '508',
    address: [ -73.999813, 40.683876 ],
    building: '154',t',
    coord: [ -73.9189064, 40.8654529 ],
    street: 'Post Ave',
    zipcode: '10034',
  },isine: 'Other',
  borough: 'Manhattan',
  cuisine: 'Other',
  grades: [],id: '50017910'
  name: '',
  restaurant_id: '50017887'
},address: {
  building: '15',
  address: [ -73.9966882, 40.7139264 ],
  building: '508',n St',
  coord: [ -73.999813, 40.683876 ],
  street: 'Henry St',
  zipcode: '11231'n',
},isine: 'Other',
borough: 'Brooklyn',
cuisine: 'Other',
grades: [],id: '50017912'
name: '',
restaurant_id: '50017910'
},address: {
  building: '4704',
  address: [ -74.013391, 40.64943 ],
  building: '15',e',
  coord: [ -73.9966882, 40.7139264 ],
  street: 'Division St',
  zipcode: '10002',
},isine: 'Other',
borough: 'Manhattan',
cuisine: 'Other',
  ... 11 more documents]
```

7.) Write a MongoDB query to arranged the name of the cuisine in ascending order and for that same cuisine borough should be in descending order.

QUERY:

```
db.restaurants.find({}, { _id: 0 }).sort({ cuisine: 1, borough: -1 })
```

OUTPUT:

```
Afrin_Fathima_014> db.restaurants.find({}, { _id: 0 }).sort({ cuisine: 1, borough: -1 })
[
  {
    address: {
      building: '154',
      coord: [ -73.9189064, 40.8654529 ],
      street: 'Post Ave',
      zipcode: '10034'
    },
    borough: 'Manhattan',
    cuisine: 'Other',
    grades: [],
    name: '',
    restaurant_id: '50017887'
  },
  {
    address: {
      building: '15',
      coord: [ -73.9966882, 40.7139264 ],
      street: 'Division St',
      zipcode: '10002'
    },
    borough: 'Manhattan',
    cuisine: 'Other',
    grades: [],
    name: '',
    restaurant_id: '50017912'
  },
  {
    address: {
      building: '508',
      coord: [ -73.999813, 40.683876 ],
      street: 'Henry St',
      zipcode: '11231'
    },
    borough: 'Brooklyn',
    cuisine: 'Other',
    grades: []
  }
]
```

8.) Write a MongoDB query to know whether all the addresses contains the street or not.

QUERY:

```
db.restaurants.find({ "address.street": { $exists: true, $ne: "" } })
```

OUTPUT:

```
Afrin_Fathima_014> db.restaurants.find({ "address.street": { $exists: true, $ne: "" } })
[
  {
    _id: ObjectId('564c2d949eb21ad392f1d6de'),
    address: {
      building: '154',
      coord: [ -73.9189064, 40.8654529 ],
      street: 'Post Ave',
      zipcode: '10034'
    },
    borough: 'Manhattan',
    cuisine: 'Other',
    grades: [],
    name: '',
    restaurant_id: '50017887'
  },
  {
    _id: ObjectId('564c2d949eb21ad392f1d6ec'),
    address: {
      building: '508',
      coord: [ -73.999813, 40.683876 ],
      street: 'Henry St',
      zipcode: '11231'
    },
    borough: 'Brooklyn',
    cuisine: 'Other',
    grades: [],
    name: ''
  }
]
```

9.) Write a MongoDB query which will select all documents in the restaurants collection where the coord field value is Double.

QUERY:

```
db.restaurants.find({ "address.coord": { $elemMatch: { $type: "double" } } })
```

OUTPUT:

```

mongosh mongoDB://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
Afrin_Fathima_014> db.restaurants.find({ "address.coord": { $elemMatch: { $type: "double" } } })
[
  {
    _id: ObjectId('564c2d949eb21ad392f1d6de'),
    address: {
      building: '154',
      coord: [ -73.9189064, 40.8654529 ],
      street: 'Post Ave',
      zipcode: '10034'
    },
    borough: 'Manhattan',
    cuisine: 'Other',
    grades: [],
    name: '',
    restaurant_id: '50017887'
  },
  {
    _id: ObjectId('564c2d949eb21ad392f1d6ec'),
    address: {
      building: '508',
      coord: [ -73.999813, 40.683876 ],
      street: 'Henry St',
      zipcode: '11231'
    },
    borough: 'Brooklyn',
    cuisine: 'Other',
    grades: [],
    name: '',
    restaurant_id: '50017910'
  },
  {
    _id: ObjectId('564c2d949eb21ad392f1d6ed'),
    address: {
      building: '15',
      coord: [ -73.9966882, 40.7139264 ],
      street: 'Division St',
      zipcode: '10002'
    },
  }
]

```

10. Write a MongoDB query which will select the restaurant Id, name and grades for those restaurants which returns 0 as a remainder after dividing the score by 7.

QUERY:

```
db.restaurants.find({ "grades.score": { $mod: [7, 0] } }, { restaurant_id: 1, name: 1, grades: 1 });
```

OUTPUT:

```

mongosh mongoDB://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
Afrin_Fathima_014> db.restaurants.find({ "grades.score": { $mod: [7, 0] } }, { restaurant_id: 1, name: 1, grades: 1 });
Afrin_Fathima_014> -

```

11. Write a MongoDB query to find the restaurant name, borough, longitude and attitude and cuisine for those restaurants which contains 'mon' as three letters somewhere in its name.

QUERY:

```
db.restaurants.find({ name: /mon/i }, { name: 1, borough: 1, "address.coord": 1, cuisine: 1 })
```

OUTPUT:

```

mongosh mongoDB://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
Afrin_Fathima_014> db.restaurants.find({ name: /mon/i }, { name: 1, borough: 1, "address.coord": 1, cuisine: 1 });
Afrin_Fathima_014> -

```

12. Write a MongoDB query to find the restaurant name, borough, longitude and latitude and cuisine for those restaurants which contain 'Mad' as first three letters of its name.

QUERY:

```
db.restaurants.find({ name: /^Mad/i }, { name: 1, borough: 1, "address.coord": 1, cuisine: 1 })
```

OUTPUT:

```
[mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000]
Afrin_Fathima_014> db.restaurants.find({ name: /^Mad/i }, { name: 1, borough: 1, "address.coord": 1, cuisine: 1 })
Afrin_Fathima_014>
```

13. Write a MongoDB query to find the restaurants that have at least one grade with a score of less than 5.

QUERY:

```
db.restaurants.find({ "grades": { $elemMatch: { "score": { $lt: 5 } } } })
```

OUTPUT:

```
[mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000]
Afrin_Fathima_014> db.restaurants.find({ "grades": { $elemMatch: { "score": { $lt: 5 } } } })
Afrin_Fathima_014>
```

14. Write a MongoDB query to find the restaurants that have at least one grade with a score of less than 5 and that are located in the borough of Manhattan.

QUERY:

```
db.restaurants.find({ "grades": { $elemMatch: { "score": { $lt: 5 } } }, "borough": "Manhattan" })
```

OUTPUT:

```
[mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000]
Afrin_Fathima_014> db.restaurants.find({ "grades": { $elemMatch: { "score": { $lt: 5 } } }, "borough": "Manhattan" })
Afrin_Fathima_014> -
```

15. Write a MongoDB query to find the restaurants that have at least one grade with a score of less than 5 and that are located in the borough of Manhattan or Brooklyn.

QUERY:

```
db.restaurants.find({ "grades": { $elemMatch: { "score": { $lt: 5 } } } }, { $or: [{ "borough": "Manhattan" }, { "borough": "Brooklyn" }] })
```

OUTPUT:

```
Afrin_Fathima_014> db.restaurants.find({ "grades": { $elemMatch: { "score": { $lt: 5 } } } }, { $or: [{ "borough": "Manhattan" }, { "borough": "Brooklyn" }] })
```

16. Write a MongoDB query to find the restaurants that have at least one grade with a score of less than 5 and that are located in the borough of Manhattan or Brooklyn, and their cuisine is not American.

QUERY:

```
db.restaurants.find({ "grades": { $elemMatch: { "score": { $lt: 5 } } } }, { $or: [{ "borough": "Manhattan" }, { "borough": "Brooklyn" }], "cuisine": { $ne: "American" } })
```

OUTPUT:

```
Afrin_Fathima_014> db.restaurants.find({ "grades": { $elemMatch: { "score": { $lt: 5 } } } }, { $or: [{ "borough": "Manhattan" }, { "borough": "Brooklyn" }], "cuisine": { $ne: "American" } })
```

17. Write a MongoDB query to find the restaurants that have at least one grade with a score of less than 5 and that are located in the borough of Manhattan or Brooklyn, and their cuisine is not American or Chinese.

QUERY:

```
db.restaurants.find({ "grades": { $elemMatch: { "score": { $lt: 5 } } } }, { $or: [{ "borough": "Manhattan" }, { "borough": "Brooklyn" }], "cuisine": { $nin: ["American", "Chinese"] } })
```

OUTPUT:

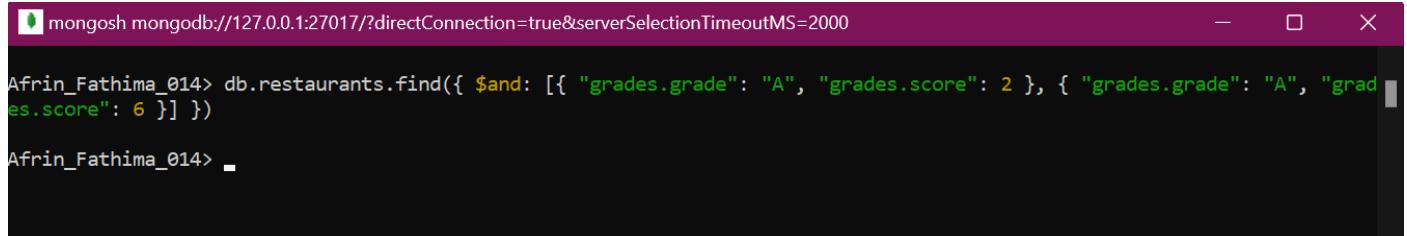
```
Afrin_Fathima_014> db.restaurants.find({ "grades": { $elemMatch: { "score": { $lt: 5 } } } }, { $or: [{ "borough": "Manhattan" }, { "borough": "Brooklyn" }], "cuisine": { $nin: ["American", "Chinese"] } })
```

18. Write a MongoDB query to find the restaurants that have a grade with a score of 2 and a grade with a score of 6.

QUERY:

```
db.restaurants.find({ $and: [{ "grades.grade": "A", "grades.score": 2 }, { "grades.grade": "A", "grades.score": 6 }] })
```

OUTPUT:



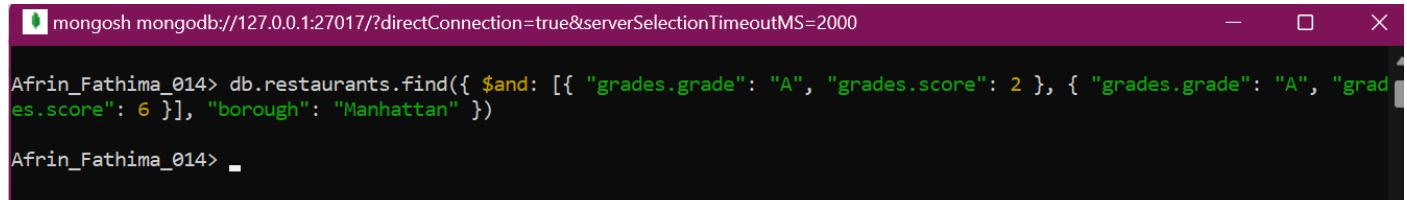
```
Afrin_Fathima_014> db.restaurants.find({ $and: [{ "grades.grade": "A", "grades.score": 2 }, { "grades.grade": "A", "grades.score": 6 }] })
```

19. Write a MongoDB query to find the restaurants that have a grade with a score of 2 and a grade with a score of 6 and are located in the borough of Manhattan.

QUERY:

```
db.restaurants.find({ $and: [{ "grades.grade": "A", "grades.score": 2 }, { "grades.grade": "A", "grades.score": 6 }], "borough": "Manhattan" })
```

OUTPUT:



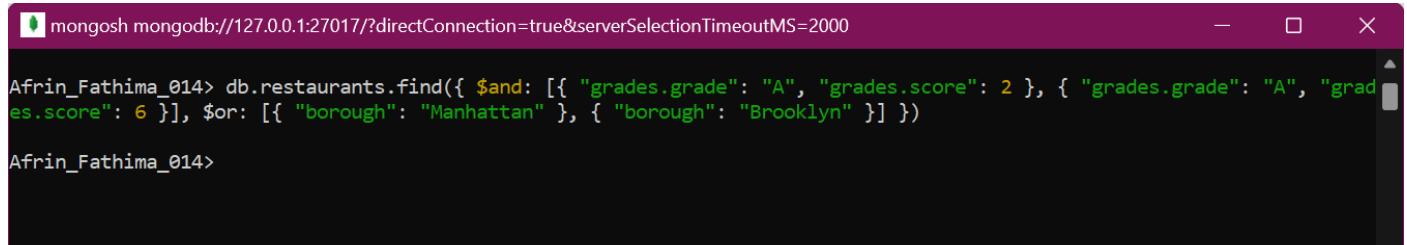
```
Afrin_Fathima_014> db.restaurants.find({ $and: [{ "grades.grade": "A", "grades.score": 2 }, { "grades.grade": "A", "grades.score": 6 }], "borough": "Manhattan" })
```

20. Write a MongoDB query to find the restaurants that have a grade with a score of 2 and a grade with a score of 6 and are located in the borough of Manhattan or Brooklyn.

QUERY:

```
db.restaurants.find({ $and: [{ "grades.grade": "A", "grades.score": 2 }, { "grades.grade": "A", "grades.score": 6 }], $or: [{ "borough": "Manhattan" }, { "borough": "Brooklyn" }] })
```

OUTPUT:



```
Afrin_Fathima_014> db.restaurants.find({ $and: [{ "grades.grade": "A", "grades.score": 2 }, { "grades.grade": "A", "grades.score": 6 }], $or: [{ "borough": "Manhattan" }, { "borough": "Brooklyn" }] })
```

21. Write a MongoDB query to find the restaurants that have a grade with a score of 2 and a grade with a score of 6 and are located in the borough of Manhattan or Brooklyn, and their cuisine is not American.

QUERY:

```
db.restaurants.find({ $and: [{ "grades.grade": "A", "grades.score": 2 }, { "grades.grade": "A", "grades.score": 6 }], $or: [{ "borough": "Manhattan" }, { "borough": "Brooklyn" }], "cuisine": { $ne: "American" } })
```

OUTPUT:

```
Afrin_Fathima_014> db.restaurants.find({ $and: [{ "grades.grade": "A", "grades.score": 2 }, { "grades.grade": "A", "grades.score": 6 }], $or: [{ "borough": "Manhattan" }, { "borough": "Brooklyn" }], "cuisine": { $ne: "American" } })
```

22. Write a MongoDB query to find the restaurants that have a grade with a score of 2 and a grade with a score of 6 and are located in the borough of Manhattan or Brooklyn, and their cuisine is not American or Chinese.

QUERY:

```
db.restaurants.find({ $and: [{ "grades.grade": "A", "grades.score": 2 }, { "grades.grade": "A", "grades.score": 6 }], $or: [{ "borough": "Manhattan" }, { "borough": "Brooklyn" }], "cuisine": { $nin: ["American", "Chinese"] } })
```

OUTPUT:

```
Afrin_Fathima_014> db.restaurants.find({ $and: [{ "grades.grade": "A", "grades.score": 2 }, { "grades.grade": "A", "grades.score": 6 }], $or: [{ "borough": "Manhattan" }, { "borough": "Brooklyn" }], "cuisine": { $nin: ["American", "Chinese"] } })
```

23. Write a MongoDB query to find the restaurants that have a grade with a score of 2 or a grade with a score of 6.

QUERY:

```
db.restaurants.find({ $or: [{ "grades.score": 2 }, { "grades.score": 6 }] })
```

OUTPUT:

```
Afrin_Fathima_014> db.restaurants.find({ $or: [{ "grades.score": 2 }, { "grades.score": 6 }] })
```

Evaluation Procedure	Marks awarded
Query(5)	
Execution (5)	
Viva(5)	
Total (15)	
Faculty Signature	

RESULT:

MONGO DB

EX_NO: 20

DATE:

1.) Find all movies with full information from the 'movies' collection that released in the year 1893.

QUERY:

```
db.movies.find({ year: 1893 })
```

OUTPUT:

```
Afrin_Fathima_014> db.movies.find({ year: 1893 })
[ {
  _id: ObjectId('573a1390f29313caabcd4135'),
  plot: 'Three men hammer on an anvil and pass a bottle of beer around.',
  genres: [ 'Short' ],
  runtime: 1,
  cast: [ 'Charles Kayser', 'John Ott' ],
  num_mflix_comments: 1,
  title: 'Blacksmith Scene',
  fullplot: 'A stationary camera looks at a large anvil with a blacksmith behind it and one on either side. The smith in the middle draws a heated metal rod from the fire, places it on the anvil, and all three begin a rhythmic hammering. After several blows, the metal goes back in the fire. One smith pulls out a bottle of beer, and they each take a swig. Then, out comes the glowing metal and the hammering resumes.',
  countries: [ 'USA' ],
  released: ISODate('1893-05-09T00:00:00.000Z'),
  directors: [ 'William K.L. Dickson' ],
  rated: 'UNRATED',
  awards: { wins: 1, nominations: 0, text: '1 win.' },
  lastupdated: '2015-08-26 00:03:50.13300000',
  year: 1893,
  imdb: { rating: 6.2, votes: 1189, id: 5 },
  type: 'movie',
  tomatoes: {
    viewer: { rating: 3, numReviews: 184, meter: 32 },
    lastUpdated: ISODate('2015-06-28T18:34:09.000Z')
  }
}
```

2.) Find all movies with full information from the 'movies' collection that have a runtime greater than 120 minutes.

QUERY:

```
db.movies.find({ runtime: { $gt: 120 } })
```

OUTPUT:

```
Afrin_Fathima_014> db.movies.find({ runtime: { $gt: 120 } })
[
  {
    _id: ObjectId('6650ef78e55a843f15cdcdf7'),
    id: ObjectId('573a1390f22313caabcd5967'),
    plot: 'An intrepid reporter and his loyal friend battle a bizarre secret society of criminals known as The Vampires.',
    genres: [ 'Action', 'Adventure', 'Crime' ],
    runtime: 399,
    rated: 'NOT RATED',
    cast: [ 'Musidora', 'édouardMathè', 'Marcel Lèvesque', 'Jean Aymè' ],
    poster: 'https://m.media-amazon.com/images/M/MV5BMTc1NTY3NDIzN15BMl5BanBnXkFtZTgwNTIyODg5MTE@._V1_SY1000_SX677_AL.jpg',
    title: 'Les vampires',
    fullplot: 'An intrepid reporter and his loyal friend battle a bizarre secret society of criminals known as The Vampires.',
    languages: [ 'French' ],
    released: ISODate('1916-11-23T00:00:00.000Z'),
    directors: [ 'Louis Feuillade' ],
    writers: [ 'Louis Feuillade' ],
    awards: { wins: 0, nominations: 1, text: '1 nomination.' },
    lastupdated: '2015-09-02 00:24:27.333000000',
    year: 1915,
    imdb: { rating: 6.8, votes: 2878, id: 6206 },
    countries: [ 'France' ],
    type: 'movie',
    tomatoes: {
      viewer: { rating: 3.8, numReviews: 2118, meter: 82 },
      dvd: ISODate('2000-05-16T00:00:00.000Z'),
      critic: { rating: 8.8, numReviews: 13, meter: 100 },
      lastUpdated: ISODate('2015-09-15T17:02:33.000Z'),
      rotten: 0,
      fresh: 13
    }
  }
]
```

3.) Find all movies with full information from the 'movies' collection that have "Short" genre.

QUERY:

```
db.movies.find({ genres: 'Short' })
```

OUTPUT:

```
Afrin_Fathima_014> db.movies.find({ genres: 'Short' })
[
  {
    _id: ObjectId('573a1390f29313caabcd4135'),
    plot: 'Three men hammer on an anvil and pass a bottle of beer around.',
    genres: [ 'Short' ],
    runtime: 1,
    cast: [ 'Charles Kayser', 'John Ott' ],
    num_mflix_comments: 1,
    title: 'Blacksmith Scene',
    fullplot: 'A stationary camera looks at a large anvil with a blacksmith behind it and one on either side. The smith in the middle draws a heated metal rod from the fire, places it on the anvil, and all three begin a rhythmic hammering. After several blows, the metal goes back in the fire. One smith pulls out a bottle of beer, and they each take a swig. Then, out comes the glowing metal and the hammering resumes.',
    countries: [ 'USA' ],
    released: ISODate('1893-05-09T00:00:00.000Z'),
    directors: [ 'William K.L. Dickson' ],
    rated: 'UNRATED',
    awards: { wins: 1, nominations: 0, text: '1 win.' },
    lastupdated: '2015-08-26 00:03:50.133000000',
    year: 1893,
    imdb: { rating: 6.2, votes: 1189, id: 5 },
    type: 'movie',
    tomatoes: {
      viewer: { rating: 3, numReviews: 184, meter: 32 },
      lastUpdated: ISODate('2015-06-28T18:34:09.000Z')
    }
  }
]
```

4.) Retrieve all movies from the 'movies' collection that were directed by "William K.L. Dickson" and include complete information for each movie.

QUERY:

```
db.movies.find({ directors: 'William K.L. Dickson' })
```

OUTPUT:

```
Afrin_Fathima_014> db.movies.find({ directors: 'William K.L. Dickson' })
[
  {
    _id: ObjectId('573a1390f29313caabcd4135'),
    plot: 'Three men hammer on an anvil and pass a bottle of beer around.',
    genres: [ 'Short' ],
    runtime: 1,
    cast: [ 'Charles Kayser', 'John Ott' ],
    num_mflix_comments: 1,
    title: 'Blacksmith Scene',
    fulplot: 'A stationary camera looks at a large anvil with a blacksmith behind it and one on either side. The smith in the middle draws a heated metal rod from the fire, places it on the anvil, and all three begin a rhythmic hammering. After several blows, the metal goes back in the fire. One smith pulls out a bottle of beer, and they each take a swig. Then, out comes the glowing metal and the hammering resumes.',
    countries: [ 'USA' ],
    released: ISODate('1893-05-09T00:00:00.000Z'),
    directors: [ 'William K.L. Dickson' ],
    rated: 'UNRATED',
    awards: { wins: 1, nominations: 0, text: '1 win.' },
    lastupdated: '2015-08-26 00:03:50.133000000',
    year: 1893,
    imdb: { rating: 6.2, votes: 1189, id: 5 },
    type: 'movie',
    tomatoes: {
      viewer: { rating: 3, numReviews: 184, meter: 32 },
      lastUpdated: ISODate('2015-06-28T18:34:09.000Z')
    }
  }
]
Afrin_Fathima_014> -
```

5.) Retrieve all movies from the 'movies' collection that were released in the USA and include complete information for each movie.

QUERY:

```
db.movies.find({ countries: 'USA' })
```

OUTPUT:

```
Afrin_Fathima_014> db.movies.find({ countries: 'USA' })
[
  {
    _id: ObjectId('573a1390f29313caabcd4135'),
    plot: 'Three men hammer on an anvil and pass a bottle of beer around.',
    genres: [ 'Short' ],
    runtime: 1,
    cast: [ 'Charles Kayser', 'John Ott' ],
    num_mflix_comments: 1,
    title: 'Blacksmith Scene',
    fulplot: 'A stationary camera looks at a large anvil with a blacksmith behind it and one on either side. The smith in the middle draws a heated metal rod from the fire, places it on the anvil, and all three begin a rhythmic hammering. After several blows, the metal goes back in the fire. One smith pulls out a bottle of beer, and they each take a swig. Then, out comes the glowing metal and the hammering resumes.',
    countries: [ 'USA' ],
    released: ISODate('1893-05-09T00:00:00.000Z'),
    directors: [ 'William K.L. Dickson' ],
    rated: 'UNRATED',
    awards: { wins: 1, nominations: 0, text: '1 win.' },
    lastupdated: '2015-08-26 00:03:50.133000000',
    year: 1893,
    imdb: { rating: 6.2, votes: 1189, id: 5 },
    type: 'movie',
    tomatoes: {
      viewer: { rating: 3, numReviews: 184, meter: 32 },
      lastUpdated: ISODate('2015-06-28T18:34:09.000Z')
    }
  }
]
Afrin_Fathima_014> -
```

6.) Retrieve all movies from the 'movies' collection that have complete information and are rated as "UNRATED".

QUERY:

```
db.movies.find({ rated: 'UNRATED' })
```

OUTPUT:

```
Afrin_Fathima_014> db.movies.find({ rated: 'UNRATED' })
[
  {
    _id: ObjectId('573a1390f29313caabcd4135'),
    plot: 'Three men hammer on an anvil and pass a bottle of beer around.',
    genres: [ 'Short' ],
    runtime: 1,
    cast: [ 'Charles Kayser', 'John Ott' ],
    num_mflix_comments: 1,
    title: 'Blacksmith Scene',
    fullplot: 'A stationary camera looks at a large anvil with a blacksmith behind it and one on either side. The smith in the middle draws a heated metal rod from the fire, places it on the anvil, and all three begin a rhythmic hammering. After several blows, the metal goes back in the fire. One smith pulls out a bottle of beer, and they all take a swig. Then, out comes the glowing metal and the hammering resumes.',
    countries: [ 'USA' ],
    released: ISODate('1893-05-09T00:00:00.000Z'),
    directors: [ 'William K.L. Dickson' ],
    rated: 'UNRATED',
    awards: { wins: 1, nominations: 0, text: '1 win.' },
    lastupdated: '2015-08-26 00:03:50.133000000',
    year: 1893,
    imdb: { rating: 6.2, votes: 1189, id: 5 },
    type: 'movie',
    tomatoes: {
      viewer: { rating: 3, numReviews: 184, meter: 32 },
      lastUpdated: ISODate('2015-06-28T18:34:09.000Z')
    }
  }
]
```

7.) Retrieve all movies from the 'movies' collection that have complete information and have received more than 1000 votes on IMDb.

QUERY:

```
db.movies.find({ 'imdb.votes': { $gt: 1000 } })
```

OUTPUT:

```
Afrin_Fathima_014> db.movies.find({ 'imdb.votes': { $gt: 1000 } })
[
  {
    _id: ObjectId('6650ef78e55a843f15cdcdf7'),
    id: ObjectId('573a1390f22313caabcd5967'),
    plot: 'An intrepid reporter and his loyal friend battle a bizarre secret society of criminals known as The Vampires.',
    genres: [ 'Action', 'Adventure', 'Crime' ],
    runtime: 399,
    rated: 'NOT RATED',
    cast: [ 'Musidora', 'èdouardMathè', 'Marcel Lèvesque', 'Jean Aymè' ],
    poster: 'https://m.media-amazon.com/images/M/MV5BMTc1NTY3NDIzMjNlZBM15BanBnXkFtZTgwNTIyODg5MTE@._V1_SY1000_SX677_AL.jpg',
    title: 'Les vampires',
    fullplot: 'An intrepid reporter and his loyal friend battle a bizarre secret society of criminals known as The Vampires.',
    languages: [ 'French' ],
    released: ISODate('1916-11-23T00:00:00.000Z'),
    directors: [ 'Louis Feuillade' ],
    writers: [ 'Louis Feuillade' ],
    awards: { wins: 0, nominations: 1, text: '1 nomination.' },
    lastupdated: '2015-09-02 00:24:27.333000000',
    year: 1915,
    imdb: { rating: 6.8, votes: 2878, id: 6206 },
    countries: [ 'France' ],
    type: 'movie',
    tomatoes: {
      viewer: { rating: 3.8, numReviews: 2118, meter: 82 },
      dvd: ISODate('2000-05-16T00:00:00.000Z'),
      critic: { rating: 8.8, numReviews: 13, meter: 100 },
      lastUpdated: ISODate('2015-09-15T17:02:33.000Z'),
      rotten: 0,
      fresh: 13
    }
  },
  {
    _id: ObjectId('573a1390f29313caabcd4135'),
    plot: 'Three men hammer on an anvil and pass a bottle of beer around.',
    genres: [ 'Short' ],
    runtime: 1,
    title: 'Blacksmith Scene',
    fullplot: 'A stationary camera looks at a large anvil with a blacksmith behind it and one on either side. The smith in the middle draws a heated metal rod from the fire, places it on the anvil, and all three begin a rhythmic hammering. After several blows, the metal goes back in the fire. One smith pulls out a bottle of beer, and they each take a swig. Then, out comes the glowing metal and the hammering resumes.',
    countries: [ 'USA' ],
    released: ISODate('1893-05-09T00:00:00.000Z'),
    directors: [ 'William K.L. Dickson' ],
    rated: 'UNRATED',
    awards: { wins: 1, nominations: 0, text: '1 win.' },
    lastupdated: '2015-08-26 00:03:50.133000000',
    year: 1893,
    imdb: { rating: 6.2, votes: 1189, id: 5 },
    type: 'movie',
    tomatoes: {
      viewer: { rating: 3, numReviews: 184, meter: 32 },
      lastUpdated: ISODate('2015-06-28T18:34:09.000Z')
    }
  }
]
Afrin_Fathima_014>
```

8.) Retrieve all movies from the 'movies' collection that have complete information and have an IMDb rating higher than 7.

QUERY:

```
db.movies.find({ 'imdb.rating': { $gt: 7 } })
```

OUTPUT:

```
Afrin_Fathima_014> db.movies.find({ 'imdb.rating': { $gt: 7 } })
[ {
  _id: ObjectId('6650f417e55a843f15cdcdf8'),
  id: ObjectId('573a1391f29313caabed7a34'),
  plot: 'A kept woman runs into her one-time fianc  and finds herself torn between love and comfort.',
  genres: [ 'Drama', 'Romance' ],
  runtime: 78,
  rated: 'TV-PG',
  cast: [
    'Edna Purviance',
    'Clarence Geldart',
    'Carl Miller',
    'Lydia Knott'
  ],
  num_mflix_comments: 3,
  poster: 'https://m.media-amazon.com/images/M/MVSBZjjiMTU2NGQtNWRKni00ZjExLwExMTUtMmNkNTU0NzRlMTA3XkEyXkFqcGdeQXVynjUwNzk3NDc@._V1_SY1000_SX677_AL.jpg',
  title: 'A Woman of Paris: A Drama of Fate',
  fullplot: 'Marie St. Clair believes she has been jilted by her artist fiance Jean when he fails to meet her at the railway station. She goes off to Paris alone. A year later, mistress of wealthy Pierre Revel, she meets Jean again. Misinterpreting events she bounces back and forth between apparent security and true love. ',
  countries: [ 'USA' ],
  released: ISODate('1923-11-04T00:00:00.000Z'),
  directors: [ 'Charles Chaplin' ],
  writers: [ 'Charles Chaplin' ],
  awards: { wins: 1, nominations: 0, text: '1 win.' },
  lastupdated: '2015-09-02 00:22:09.303000000',
  year: 1923,
  imdb: { rating: 7.1, votes: 3179, id: 14624 },
  type: 'movie',
  tomatoes: {
    viewer: { rating: 3.7, numReviews: 886, meter: 78 },
    dvd: ISODate('2004-03-02T00:00:00.000Z'),
    critic: { rating: 7.4, numReviews: 11, meter: 91 },
    lastUpdated: ISODate('2015-08-23T18:34:44.000Z'),
    rotten: 1,
    production: 'Criterion Collection',
    fresh: 10
  }
}
```

9.) Retrieve all movies from the 'movies' collection that have complete information and have a viewer rating higher than 4 on Tomatoes.

QUERY:

```
db.movies.find({ 'tomatoes.viewer.rating': { $gt: 4 } })
```

OUTPUT:

```
Afrin_Fathima_014> db.movies.find({ 'tomatoes.viewer.rating': { $gt: 4 } })
[
  {
    _id: ObjectId('6650f46ae55a843f15cdcdf9'),
    id: ObjectId('573a1391f29913caabcb8945'),
    plot: 'A married farmer falls under the spell of a slatternly woman from the city, who tries to convince him to drown his wife.',
    genres: [ 'Drama', 'Romance' ],
    runtime: 94,
    rated: 'NOT RATED',
    cast: [
      "George O'Brien",
      'Janet Gaynor',
      'Margaret Livingston',
      'Bodil Rosing'
    ],
    num_mflix_comments: 1,
    poster: 'https://m.media-amazon.com/images/M/MV5BNDVkYmYwM2ItNzRiMy00NWQ4LTlhMjMtdI1ZDYyOGVmMzJjXkEyXkFqcGdeQXVvNTgzMzU5MDI@._V1_SY1000_SX677_AL.jpg',
    title: 'Sunrise',
    fullplot: 'In this fable-morality subtitled "A Song of Two Humans", the "evil" temptress is a city woman who bewitches farmer Anses and tries to convince him to murder his neglected wife, Indre.',
    countries: [ 'USA' ],
    released: ISODate('1927-11-04T00:00:00Z'),
    directors: [ 'F.W. Murnau' ],
    writers: [
      'Carl Mayer (scenario)',
      'Hermann Sudermann (from an original theme by)',
      'Katherine Hilliker (titles)',
      'H.H. Caldwell (titles)'
    ],
    awards: {
      wins: 5,
      nominations: 1,
      text: 'Won 3 Oscars. Another 2 wins & 1 nomination.'
    },
    lastupdated: '2015-09-12 00:26:13.493000000',
    year: 1927,
    imdb: { rating: 8.4, votes: 24480, id: 18455 },
    type: 'movie',
    tomatoes: {
      viewer: { rating: 4.4, numReviews: 9134, meter: 92 },
      dvd: ISODate('2008-12-09T00:00:00Z'),
      critic: { rating: 8.9, numReviews: 48, meter: 98 },
      lastUpdated: ISODate('2015-09-10T19:15:02Z'),
      consensus: 'Boasting masterful cinematography to match its well-acted, wonderfully romantic storyline, Sunrise is perhaps the final -- and arguably definitive -- statement of the silent era.',
      rotten: 1,
      production: 'Fox Films',
      fresh: 47
    }
  }
]
Afrin_Fathima_014>
```

10.) Retrieve all movies from the 'movies' collection that have received an award.

QUERY:

```
db.movies.find({ 'awards.wins': { $gt: 0 } })
```

OUTPUT:

```
[mongosh mongod://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
]
Afrin_Fathima_014> db.movies.find({ 'awards.wins': { $gt: 0 } })
[
  {
    _id: ObjectId('573a1390f29313caabcd4135'),
    plot: 'Three men hammer on an anvil and pass a bottle of beer around.',
    genres: [ 'Short' ],
    runtime: 1,
    cast: [ 'Charles Kayser', 'John Ott' ],
    num_mflix_comments: 1,
    title: 'Blacksmith Scene',
    fullplot: 'A stationary camera looks at a large anvil with a blacksmith behind it and one on either side. The smith in the middle draws a heated metal rod from the fire, places it on the anvil, and all three begin a rhythmic hammering. After several blows, the metal goes back in the fire. One smith pulls out a bottle of beer, and they each take a swig. Then, out comes the glowing metal and the hammering resumes.',
    countries: [ 'USA' ],
    released: ISODate('1893-05-09T00:00:00.000Z'),
    directors: [ 'William K.L. Dickson' ],
    rated: 'UNRATED',
    awards: { wins: 1, nominations: 0, text: '1 win.' },
    lastupdated: '2015-08-26 00:03:50.1300000',
    year: 1893,
    imdb: { rating: 6.2, votes: 1189, id: 5 },
    type: 'movie',
    tomatoes: {
      viewer: { rating: 3, numReviews: 184, meter: 32 },
      lastUpdated: ISODate('2015-06-28T18:34:09.000Z')
    }
  },
  {
    _id: ObjectId('6650f417e55a843f15cdcdf8'),
    id: ObjectId('573a1391f29313caabcd7a34'),
    plot: 'A kept woman runs into her one-time fiancé and finds herself torn between love and comfort.',
    genres: [ 'Drama', 'Romance' ],
    runtime: 78,
    rated: 'TV-PG',
    cast: [
      'Edna Purviance',
      'Clarence Geldart',
      'Carl Miller',
      'Lydia Knott'
    ],
    num_mflix_comments: 3,
    poster: 'https://m.media-amazon.com/images/M/MV5BZjJiMTU2NGQtNWRkNi00ZjExLWExMTUtMmNkNTU0NzRlMTA3XkEyXkFqcGdeQXVyNjUwNzk3NDc@._V1_SY1000_SX677_AL.jpg',
    title: 'A Woman of Paris: A Drama of Fate',
    fullplot: 'Marie St. Clair believes she has been jilted by her artist fiance Jean when he fails to meet her at the railway station. She goes off to Paris alone. A year later, mistress of wealthy Pierre Revel, she meets Jean again. Misinterpreting events she bounces back and forth between apparent security and true love.',
    countries: [ 'USA' ],
    released: ISODate('1923-11-04T00:00:00.000Z'),
    directors: [ 'Charles Chaplin' ],
    writers: [ 'Charles Chaplin' ],
    awards: { wins: 1, nominations: 0, text: '1 win.' },
    lastupdated: '2015-09-02 00:22:09.30300000',
    year: 1923,
    imdb: { rating: 7.1, votes: 3179, id: 14624 },
    type: 'movie',
    tomatoes: {
      viewer: { rating: 3.7, numReviews: 886, meter: 78 },
      dvd: ISODate('2004-03-02T00:00:00.000Z'),
      critic: { rating: 7.4, numReviews: 11, meter: 91 },
      lastUpdated: ISODate('2015-08-23T18:34:44.000Z'),
      rotten: 1,
      production: 'Criterion Collection',
      fresh: 10
    }
  },
  {
    _id: ObjectId('6650f46ae55a843f15cdcdf9'),
    id: ObjectId('573a1391f29313caabcd8945'),
    plot: 'A married farmer falls under the spell of a slatternly woman from the city, who tries to convince him to drown his wife.',
    genres: [ 'Drama', 'Romance' ],
    runtime: 94,
    rated: 'NOT RATED',
    cast: [
      'George O'Brien',
      'Janet Gaynor',
      'Margaret Livingston',
      'Bodil Rosing'
    ],
    num_mflix_comments: 1,
    poster: 'https://m.media-amazon.com/images/M/MV5BNDVkYmYwM2ItNzRiMy00NWQ4LTlhMjMtdI1ZDYyOGVmMzJjXkEyXkFqcGdeQXVyNTgzMzU5NDI@._V1_SY1000_SX677_AL.jpg',
    title: 'Sunrise',
  }
]
```

```

mongosh mongo://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
fuliplot: In this fable-morality subtitled "A Song of Two Humans", the "evil" temptress is a city woman who bewitches farmer Anses and tries to convince him to murder his neglected wife, Indre.
countries: [ 'USA' ],
released: ISODate('1927-11-04T00:00:00.000Z'),
directors: [ 'F.W. Murnau' ],
writers: [
  'Carl Mayer (scenario)',
  'Hermann Sudermann (from an original theme by)',
  'Katherine Hilliker (titles)',
  'H.H. Caldwell (titles)'
],
awards: {
  wins: 5,
  nominations: 1,
  text: 'Won 3 Oscars. Another 2 wins & 1 nomination.'
},
lastupdated: '2015-09-12 00:26:13.493000000',
year: 1927,
imdb: { rating: 8.4, votes: 24480, id: 18455 },
type: 'movie',
tomatoes: {
  viewer: { rating: 4.4, numReviews: 9134, meter: 92 },
  dvd: ISODate('2008-12-09T00:00:00.000Z'),
  critic: { rating: 8.9, numReviews: 48, meter: 98 },
  lastUpdated: ISODate('2015-09-10T19:15:02.000Z'),
  consensus: 'Boasting masterful cinematography to match its well-acted, wonderfully romantic storyline, Sunrise is perhaps the final -- and arguably definitive -- statement of the silent era.',
  rotten: 1,
  production: 'Fox Films',
  fresh: 47
}
]
Afrin_Fathima_014>

```

11.) Find all movies with title, languages, released, directors, writers, awards, year, genres, runtime, cast, countries from the 'movies' collection in MongoDB that have at least one nomination.

QUERY:

```
db.movies.find( { 'awards.nominations': { $gt: 0 } }, { title: 1, languages: 1, released: 1, directors: 1, writers: 1, awards: 1, year: 1, genres: 1, runtime: 1, cast: 1, countries: 1 } )
```

OUTPUT:

```

mongosh mongo://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
]
Afrin_Fathima_014> db.movies.find( { 'awards.nominations': { $gt: 0 } }, { title: 1, languages: 1, released: 1, directors: 1, writers: 1, awards: 1, year: 1, genres: 1, runtime: 1, cast: 1, countries: 1 } )
[
{
  _id: ObjectId('6650ef78e55a843f15cdcdf7'),
  genres: [ 'Action', 'Adventure', 'Crime' ],
  runtime: 399,
  cast: [ 'Musidora', 'éduardMathé', 'Marcel Lèvesque', 'Jean Aymè' ],
  title: 'Les vampires',
  languages: [ 'French' ],
  released: ISODate('1916-11-23T00:00:00.000Z'),
  directors: [ 'Louis Feuillade' ],
  writers: [ 'Louis Feuillade' ],
  awards: { wins: 0, nominations: 1, text: '1 nomination.' },
  year: 1915,
  countries: [ 'France' ]
},
{
  _id: ObjectId('6650f46ae55a843f15cdcdf9'),
  genres: [ 'Drama', 'Romance' ],
  runtime: 94,
  cast: [
    "George O'Brien",
    'Janet Gaynor',
    'Margaret Livingston',
    'BodilRosing'
  ],
  title: 'Sunrise',
  countries: [ 'USA' ],
  released: ISODate('1927-11-04T00:00:00.000Z'),
  directors: [ 'F.W. Murnau' ],
  writers: [
    'Carl Mayer (scenario)',
    'Hermann Sudermann (from an original theme by)',
    'Katherine Hilliker (titles)',
    'H.H. Caldwell (titles)'
  ],
  awards: {
    wins: 5,
    nominations: 1,
    text: 'Won 3 Oscars. Another 2 wins & 1 nomination.'
  }
}

```

12.) Find all movies with title, languages, released, directors, writers, awards, year, genres, runtime, cast, countries from the 'movies' collection in MongoDB with cast including "Charles Kayser".

QUERY:

```
db.movies.find( { cast: 'Charles Kayser' }, { title: 1, languages: 1, released: 1, directors: 1, writers: 1, awards: 1, year: 1, genres: 1, runtime: 1, cast: 1, countries: 1 } )
```

OUTPUT:

```
Afrin_Fathima_014> db.movies.find( { cast: 'Charles Kayser' }, { title: 1, languages: 1, released: 1, directors: 1, writers: 1, awards: 1, year: 1, genres: 1, runtime: 1, cast: 1, countries: 1 } )
[
  {
    _id: ObjectId('573a1390f29313caabcd4135'),
    genres: [ 'Short' ],
    runtime: 1,
    cast: [ 'Charles Kayser', 'John Ott' ],
    title: 'Blacksmith Scene',
    countries: [ 'USA' ],
    released: ISODate('1893-05-09T00:00:00.000Z'),
    directors: [ 'William K.L. Dickson' ],
    awards: { wins: 1, nominations: 0, text: '1 win.' },
    year: 1893
  }
]
Afrin_Fathima_014>
```

13.) Retrieve all movies with title, languages, released, directors, writers, countries from the 'movies' collection in MongoDB that released on May 9, 1893.

QUERY:

```
db.movies.find( { released: ISODate("1893-05-09T00:00:00.000Z") }, { title: 1, languages: 1, released: 1, directors: 1, writers: 1, countries: 1 } )
```

OUTPUT:

```
Afrin_Fathima_014> db.movies.find( { released: ISODate("1893-05-09T00:00:00.000Z") }, { title: 1, languages: 1, released: 1, directors: 1, writers: 1, countries: 1 } )
[
  {
    _id: ObjectId('573a1390f29313caabcd4135'),
    title: 'Blacksmith Scene',
    countries: [ 'USA' ],
    released: ISODate('1893-05-09T00:00:00.000Z'),
    directors: [ 'William K.L. Dickson' ]
  }
]
Afrin_Fathima_014>
```

14.) Retrieve all movies with title, languages, released, directors, writers, countries from the 'movies' collection in MongoDB that have a word "scene" in the title.

QUERY:

```
db.movies.find( { title: /scene/i }, { title: 1, languages: 1, released: 1, directors: 1, writers: 1, countries: 1 })
```

OUTPUT:

```
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
Afrin_Fathima_014> db.movies.find( { title: /scene/i }, { title: 1, languages: 1, released: 1, directors: 1, writers: 1, countries: 1 })
[
  {
    _id: ObjectId('573a1390f29313caabcd4135'),
    title: 'Blacksmith Scene',
    countries: [ 'USA' ],
    released: ISODate('1893-05-09T00:00:00.000Z'),
    directors: [ 'William K.L. Dickson' ]
  }
]
Afrin_Fathima_014>
```

Evaluation Procedure	Marks awarded
Query(5)	
Execution (5)	
Viva(5)	
Total (15)	
Faculty Signature	

RESULT: