

Ex.No.: 2		DATA MANIPULATIONS
Date:	08/08/2024	

a) Find out the employee id, names, salaries of all the employees [select](#)

[Employee_id, First_Name, Salary from EMPLOYEES;](#)

EMPLOYEE_ID	FIRST_NAME	SALARY
1	Justin	4900
2	Emma	5500
3	Robert	9000
4	Scarlett	8000
5	Chris	7500
6	Mark	7200
7	Chris	7800
8	Jeremy	3800
9	Tom	6000

b) List out the employees who works under manager 100

[select First_Name || ' ' || Last_Name as name from EMPLOYEES where manager_id =100;](#)

NAME
Cate Austin
Justin Beiber

2 rows returned in 0.04 seconds [Download](#)

c) Find the names of the employees who have a salary greater than or equal to 4800

[select First_Name || ' ' || Last_Name as name from EMPLOYEES
Where salary >= 4800;](#)

NAME
Emma Stone
Brie Larson
Elizabeth Olsen
Cate Austin
Robert Downey
Karen Gillan
Sebastian Stan
Karl Austin
Chris Evans

d) List out the employees whose last name is AUSTIN

```
select First_Name || ' ' || Last_Name as name from EMPLOYEES  
where Last_Name = 'Austin';
```

NAME
Cate Austin
Karl Austin
Jeremy Austin
Chris Austin
Zoe Austin
Scarlett Austin

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

e) Find the names of the employees who works in departments 60,70 and 80

```
select First_Name || ' ' || Last_Name as name from EMPLOYEES  
where Department_id in (60,70,80);
```

NAME
Chadwick Boseman
Jeremy Austin
Tessa Thompson
Zoe Austin
Pom Klementieff

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

f) Display the unique Manager_Id.

```
select DISTINCT(manager_id) from EMPLOYEES;
```

MANAGER_ID
400
200
350
300
250
450
600
550
900
800

More than 10 rows available. Increase rows selector to view more rows.

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

(a) Insert Five Records and calculate GrossPay and NetPay.

```
INSERT INTO Emp (EmpNo, EmpName, Job, Basic, DA, HRA, PF, GrossPay, NetPay)  
VALUES (  
101, 'John Doe', 'Manager', 50000, 15000, 20000, 6000,0,0 ,
```

```

102, 'Jane Smith', 'Developer', 40000, 12000, 16000, 4800,0,0 ,
103, 'Alice Johnson', 'Analyst', 35000, 10500, 14000, 4200,0,0 ,
104, 'Bob Brown', 'Designer', 30000, 9000, 12000, 3600,0,0 ,
105, 'Charlie Davis', 'Tester', 25000, 7500, 10000, 3000,0,0
)

```

```

update emp
set GrossPay = Basic+DA+HRA
where Grosspay = 0;

```

```

update emp
set NetPay = Grosspay - PF
where Netpay = 0;

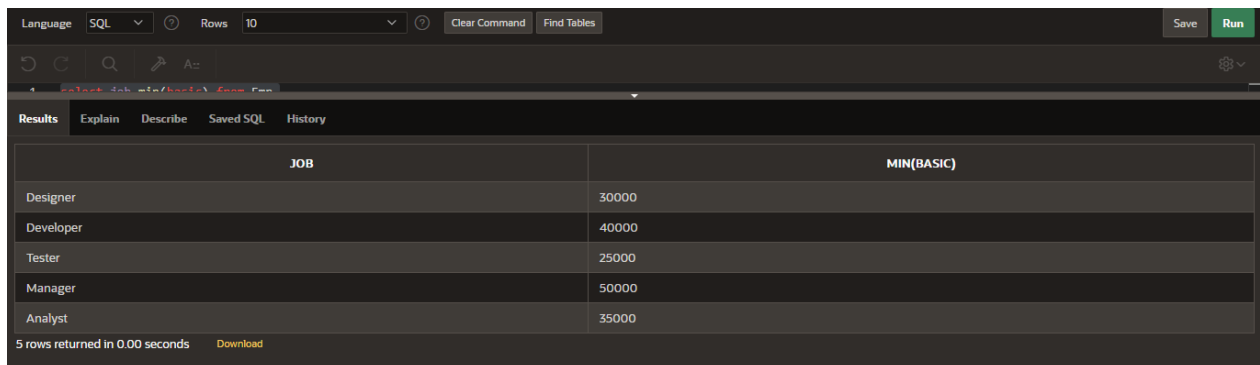
```

(b) Display the employees whose Basic is lowest in each department.

```

select job,min(basic) from Emp
group by Job;

```



The screenshot shows a SQL query execution interface. The query entered is `select job,min(basic) from Emp group by Job;`. The results are displayed in a table with two columns: **JOB** and **MIN(BASIC)**. The results are as follows:

JOB	MIN(BASIC)
Designer	30000
Developer	40000
Tester	25000
Manager	50000
Analyst	35000

At the bottom of the interface, it states "5 rows returned in 0.00 seconds" and provides a "Download" link.

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

Create table DEPT(

```

ID Number(7),
Name varchar(25)
);

```

```
Desc DEPT;
```

Results

Explain

Describe

Saved SQL

History

Object Type

TABLE

Object

DEPT

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
DEPT	ID	NUMBER	-	7	0	-		-	-
	NAME	VARCHAR2	25	-	-	-		-	-

2) Create the EMP1 table based on the following instance chart. Confirm that the table is created.

```

create table EMP1(
  ID Number(7),
  First_name varchar(25),
  Last_name varchar(25),
  Dept_id Number(7)
);

```

```
Desc EMP1;
```

Results	Explain	Describe	Saved SQL	History					
Object Type TABLE ⌵ Object EMP1 ⌵									
Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
EMP1	ID	NUMBER	-	7	0	-		-	-
	FIRST_NAME	VARCHAR2	25	-	-	-		-	-
	LAST_NAME	VARCHAR2	25	-	-	-		-	-
	DEPT_ID	NUMBER	-	7	0	-		-	-

3) Modify the EMP1 table to allow for longer employee last names. Confirm the modification.(Hint: Increase the size to 50)

```

ALTER TABLE EMP1
modify Last_name varchar(50);

```

ResultsExplainDescribeSaved SQLHistory

Object TypeTABLEObjectEMP1

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
EMP1	ID	NUMBER	-	7	0	-	✓	-	-
	FIRST_NAME	VARCHAR2	25	-	-	-	✓	-	-
	LAST_NAME	VARCHAR2	50	-	-	-	✓	-	-
	DEPT_ID	NUMBER	-	7	0	-	✓	-	-

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 columns. Name the columns Id, First_name, Last_name, salary and Dept_id respectively.

```
create table EMPLOYEES2(
  ID Number(10),
  First_name varchar(50),
  Last_name varchar(50),
  Salary Number(10),
  Dept_id Number(10)
);
```

5) Drop the EMP1 table.

```
drop table EMP1;
```

6) Rename the EMPLOYEES2 table as EMP1.

```
ALTER TABLE EMPLOYEES2 RENAME TO EMP1;
```

7) Add a comment on DEPT and EMP1 tables. Confirm the modification by describing the table.

```
comment on TABLE DEPT IS 'this table contains the fields ID and NAME..';
```

```
SELECT TABLE_NAME, COMMENTS
FROM USER_TAB_COMMENTS
WHERE TABLE_NAME = 'DEPT';
```

Results

Explain

Describe

Saved SQL

History

TABLE_NAME	COMMENTS
DEPT	this table contains the fields ID and NAME..

1 rows returned in 0.06 seconds [Download](#)

comment on TABLE EMP1 IS 'this table contains the fields ID,first name,last name,salary,DEPT_id..';

```
SELECT TABLE_NAME, COMMENTS  
FROM USER_TAB_COMMENTS  
WHERE TABLE_NAME = 'EMP1';
```

Results

Explain

Describe

Saved SQL

History

TABLE_NAME	COMMENTS
EMP1	this table contains the fields ID,first name,last name,salary,DEPT_id..

1 rows returned in 0.04 seconds

Download

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

```
ALTER TABLE EMP1  
drop column First_name;
```

Results

Explain

Describe

Saved SQL

History

Object Type

TABLE

Object

EMP1

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
EMP1	ID	NUMBER	-	10	0	-		-	-
	LAST_NAME	VARCHAR2	50	-	-	-		-	-
	SALARY	NUMBER	-	10	0	-		-	-
	DEPT_ID	NUMBER	-	10	0	-		-	-