1.Group-6

Members:-
1.Hamza Ahmad (Leader)(FA18-BCS-084)
2.Hanzala Shahid (FA18-BCS-014)
3.Usama Fareed (FA18-BCS-026)
4.Sharjeel Khan Niazi (FA18-BCS-019)
5.Hamza Aslam (SP17-BCS-050)
6.Abdullah Noor Niazi (FA18-BCS-004)
7.Usman Jadoon (FA18-BCS-100)

Common Solution:-

Lab 6

Qno1:-

Answer:-

```
create database DreamHome;
USE DreamHome;
create table Branch
(
branchNo varchar(20) NOT NULL Primary Key,
street varchar(100) NOT NULL,
city varchar(50) NOT NULL,
postcode varchar(20) NOT NULL
);
create table Staff
(
staffNo varchar(20) NOT NULL PRIMARY KEY,
fName varchar(50) NOT NULL,
lName varchar(50) NOT NULL,
position varchar(50) NOT NULL,
sex varchar(1) NOT NULL, DOB DateTime NOT NULL,
salary DECIMAL NOT NULL,
branchNo varchar(20) NOT NULL References Branch(branchNo)
);
create table Client
(
clientNo varchar(20) NOT NULL PRIMARY KEY,
fName varchar(50) NOT NULL,
lName varchar(50) NOT NULL,
telNo varchar(20) NOT NULL,
prefType varchar(50) NOT NULL,
maxRent DECIMAL NOT NULL
);
create table PrivateOwner
(
ownerNo varchar(20) NOT NULL PRIMARY KEY,
fName varchar(50) NOT NULL,
lName varchar(50) NOT NULL,
address varchar(50) NOT NULL,
telNo varchar(20) NOT NULL
);
create table PropertyForRent
(
propertyNo varchar(20) NOT NULL PRIMARY KEY,
street varchar(100) NOT NULL,
city varchar(50) NOT NULL,
postcode varchar(20) NOT NULL,
type varchar(10) NOT NULL,
rooms int NOT NULL,
rent DECIMAL NOT NULL,
ownerNo varchar(20) References PrivateOwner(ownerNo),
staffNo varchar(20) NOT NULL References Staff(staffNo),
branchNo varchar(20) NOT NULL References Branch(branchNo)
);
create table Viewing
(
clientNo varchar(20) NOT NULL References Client(clientNo),
propertyNo varchar(20) NOT NULL References PropertyForRent(propertyNo),
viewDate DateTime NOT NULL,
comment varchar(200) NOT NULL
);
create table Registration
(
clientNo varchar(20) NOT NULL References Client(clientNo),
branchNo varchar(20) NOT NULL References Branch(branchNo),
staffNo varchar(20) NOT NULL References Staff(staffNo),
dateJoined DateTime NOT NULL
);


INSERT into Branch
(
branchNo,
street,
city,
postcode
)
VALUES
(N'B001',N'H#7 I-10/2', N'ISB', N'52000'),
(N'B002',N'H#78 Supply', N'ABT', N'53000'),
(N'B003',N'H#79 I-10/2', N'ISB', N'52000'),
(N'B004',N'H#78 Mandian', N'ABT', N'53000');

insert into Staff
(
staffNo,
fName,
lName,
position,
sex,
DOB,
salary,
branchNo
)
VALUES
(
N'SA9', N'Mary', N'Howe', N'Assistant', N'F',CAST(0x0000641000000000 AS DateTime),
CAST(9000 AS Decimal(18, 0)),N'B002'
),
(N'SG14', N'David', N'Ford', N'Supervisor', N'M',
CAST(0x0000531200000000 AS DateTime), CAST(18000 AS Decimal(18,0)),
N'B003'),
(N'SG37', N'Ann', N'Beech', N'Assistant', N'F',
CAST(0x000056D400000000 AS DateTime), CAST(12000 AS Decimal(18,0)),
N'B003'),
(N'SG5',N'Susan', N'Brand', N'Manager', N'F',
CAST(0x0000C85800000000 AS DateTime), CAST(24000 AS Decimal(18,0)),
N'B003'),
(N'SL21', N'John', N'White', N'Manager', N'M',
CAST(0x0000CFF200000000 AS DateTime), CAST(30000 AS Decimal(18,0)),
N'B004'),
(N'SL41', N'Julie', N'Lee', N'Assistant', N'F',
CAST(0x00005D6000000000 AS DateTime), CAST(9000 AS Decimal(18, 0)),
N'B002');


insert into Client
(
clientNo,
fName,
lName,
telNo,
prefType,
maxRent
)
values
(
'B1001','Mahad','Ali','030078601','yes',1000.0
),
(
'B1002','Sharjeel','Khan','030054621','yes',2000.0
),
(
'B1003','Hanzala','Shahid','030456601','no',1500.0
),
(
'B1004','Hamza','Aslam','0306446641','yes',8800.0
),
(
'B1005','Hamza','Ahmad','0354654401','noo',800.0
),
(
'B1006','Usama','Fareed','030074541','yes',4000.0
```

```
);

insert into PrivateOwner
(
ownerNo,
fName,
lName,
[address],
telNo
)
values
(
'B1','Azid','Ali','F18-4A','0354654264'
),
(
'B2','Mahad','Ali','F17-4A','0345154264'
),
(
'B3','Sharjeel','Khan','F14-7A','0352354264'
),
(
'B4','Hanzala','Shahid','F88-4A','0359354264'
),
(
'B5','Hamza','Aslam','F11-3A','0351694264'
),
(
'B6','Hamza','Ahmad','F19-5A','0354654264'
);


insert into PropertyForRent
(
propertyNo,street,city,postcode,[type],rooms,rent,ownerNo,staffNo,branchNo
)
values
(
'BF2','H2-h2','ABT','22010','large',8,'30000','B2','SG14','B002'
),
(
'BF3','H3-h3','ISB','62010','medium',6,'20000','B3','SG37','B003'
),
(
'BF4','H4-h4','ISB','62010','small',4,'10000','B4','SG5','B004'
);


insert into Viewing
(
clientNo,
propertyNo,
viewDate,
comment
)
values
(
'B1002','BF2','2020-2-10','No, i dont wanna say anything'
```

```
),
(
'B1003','BF3','2020-3-10','No, i dont wanna say anything'
),
(
'B1004','BF4','2020-4-10','No, i dont wanna say anything'
);


insert into Viewing
(
clientNo,
propertyNo,
viewDate,
comment
)
values
(
'B1002','BF2','2020-2-10','No, i dont wanna say anything'
),
(
'B1003','BF3','2020-3-10','No, i dont wanna say anything'
),
(
'B1004','BF4','2020-4-10','No, i dont wanna say anything'
);
```

End of qno1

Qno:-2

Answer:-

```
select * from Branch update Branch set city='ABT' where city='ISB';
```

Lab 7

Qno1:-

Answer

```
select distinct(postcode) from Branch
```

Qno2:-

Answer

```
select distinct(fName) from Staff
```

Qno3:-

Answer

```
select staffNo as [Cadre No], fName as [Baptism Name], lName as [Sur name],
position as [Locale], sex as [Gender],DOB as [Birtday] ,salary as Income,
branchNo as [Section No] from Staff
```

Qno4:-

Answer

```
select clientNo as [Buyer No], fName as [Baptism Name], lName as [Sur name],
telNo as [Fax Number],prefType as [Proclivity Type],maxRent as [Supreme Cost] from
Client;
```

Qno5:-

Answer

```
select * from Staff where salary>10000
```

Qno6:-

Answer

```
select * from Staff where position='Manager' or position='Supervisor'
```

Lab 8

Qno1:-

Answer

```
select staffNo,fName,lName,salary from staff order by salary desc
```

Qno2:-

Answer

```
select propertyNo,type,rooms,rent from PropertyForRent
order by type
select propertyNo,type,rooms,rent
from PropertyForRent
order by type,rent desc
```

Qno3:-

Answer

```
select count(*) as myCount
from PropertyForRent
where rent<=500
```

Qno4:-

Answer

```
select count(Distinct propertyNo) As myCount from Viewing
WHERE viewDate BETWEEN '1-May-04' AND '31-May-04';
```

Qno5:-

Answer

```
select count(staffNo) as myCount,sum(salary) as mySalary from staff
where
position='Manager'
```

Qno6:-

Answer

```
select MIN(salary) as myMin,
MAX(salary) as myMax,
AVG(salary) as myAVG from Staff
```

Qno7:-

Answer

```
SELECT staffNo, fName, lName, position, salary
FROM Staff
WHERE (SELECT AVG(salary) FROM Staff) < salary;
```

Qno8:-

Answer

```
select *from Staff where salary> any(select salary from Staff where branchNo='B003')
```

Qno9:-

Answer

```
select *from Staff where salary> all(select salary from Staff where branchNo='B003')
```

LAB 9

Qno1:

list all tables in the employees database

Answer:

USE EMPLOYEE;

show TABLES;

Qno2:

Write a query to find the names (first_name, last_name) and the salaries of the employees who have a higher salary than the employee whose last_name='Bull'.

Answer:-

Select FIRST_NAME, LAST_NAME, SALARY

FROM employees

WHERE SALARY>(SELECT salary FROM employees WHERE last_name='Bull');

Qno3:

Answer:-

```
Select first_name , last_name

FROM employees

WHERE department_id

IN(SELECT department_id FROM departments WHERE department_name=’IT’);
```

LAB 10

Qno1:-

Write a query to find the names (first_name, last_name) of the employees who have a manager and work for a department based in the United States.

Answer:-

```
SELECT first_name, last_name FROM employees

WHERE manager_id in(select employee_id FROM employees WHERE department_id

IN(SELECT department_id FROM departments WHERE location_id IN(select location_id from locations

Where country_id=’US’)));
```

Qno2:-

Write a query to find the names (first_name, last_name) of the employees who are managers.

Answer:-

```
SELECT first_name, last_name

FROM employees

WHERE (employee_id IN(SELECT manager_id FROM employees));
```

Qno3:-

Write a query to find the names (first_name, last_name), the salary of the employees whose salary is greater than the average salary

Answer:-

```
SELECT first_name, last_name, salary FROM employees

WHERE salary > (SELECT AVG(salary) FROM employees);
```

Qno4:-

Write a query to find the names (first_name, last_name), the salary of the employees whose salary is equal to the minimum salary for their job grade.

Answer:-

```
SELECT first_name,last_name,salary FROM employees WHERE employees.salary=(SELECT min_salary
FROM jobs WHERE employees.job_id=jobs.job_id);
```

Qno5:-

Write a query to find the names (first_name, last_name), the salary of the employees who earn more than the average salary and who works in any of the IT departments.

Answer:-

```
SELECT first_name,last_name,salary

FROM employees WHERE department_id IN (SELECT department_id FROM departments WHERE
department_name LIKE ‘IT%’)AND salary>(SELECT avg(salary) From employees);
```

Qno6:-

Write a query to find the names (first_name, last_name), the salary of the employees who earn more than Mr. Bell

Answer:-

```
SELECT * FROM employees WHERE salary > ALL(SELECT AVG(salary) FROM employees GROUP BY

Department_id);
```

Qno7:-

Write a query to find the names (first_name, last_name), the salary of the employees who earn the same salary as the minimum salary for all departments

Answer:-

```
SELECT * FROM employees

WHERE salary=(SELECT MIN(salary) FROM employees);
```

Qno8:-

Write a query to find the names (first_name, last_name), the salary of the employees whose salary greater than the average salary of all departments

Answer:-

```
SELECT first_name,last_name from employees whose(SELECT AVG(salary) from departments)
```

Qno9:-

Write a query to find the names (first_name, last_name) and salary of the employees who earn a salary that is higher than the salary of all the Shipping Clerk (JOB_ID = 'SH_CLERK'). Sort the results of the salary of the lowest to highest

Answer:-

```
SELECT first_name,last_name, job_id, salary

FROM employees

WHERE salary >

ALL (SELECT salary FROM employees WHERE job_id = 'SH_CLERK') ORDER BY salary;
```

Qno10:-

.Write a query to find the names (first_name, last_name) of the employees who are not supervisors.

Answer:-

```
SELECT b.first_name,b.last_name

FROM employees b

WHERE NOT EXISTS (SELECT 'X' FROM employees a WHERE a.manager_id = b.employee_id);
```

Qno11:-

Write a query to display the employee ID, first name, last names, and department names of all employees.

Answer:-

```
SELECT employee_id, first_name, last_name,
(SELECT department_name FROM departments d
 WHERE e.department_id = d.department_id) department
 FROM employees e ORDER BY department;
```

Qno12:-

Write a query to display the employee ID, first name, last names, salary of all employees whose salary is above average for their departments

Answer:-

```
SELECT employee_id, first_name
FROM employees AS A
WHERE salary >
(SELECT AVG(salary) FROM employees WHERE department_id = A.department_id);
```

Qno13:-

Write a query to fetch even numbered records from employees table

Answer:-

```
SET @i = 0;
SELECT i, employee_id
FROM (SELECT @i := @i + 1 AS i, employee_id FROM employees)
a WHERE MOD(a.i, 2) = 0;
```

Qno14:-

Write a query to find the 5th maximum salary in the employees table.

Answer:-

```
SELECT DISTINCT salary
```

```
FROM employees e1
WHERE 5 = (SELECT COUNT(DISTINCT salary)
FROM employees  e2
WHERE e2.salary >= e1.salary);
```

Qno15:-

Write a query to find the 4th minimum salary in the employees table

Answer:-

```
SELECT DISTINCT salary
FROM employees e1
WHERE 4 = (SELECT COUNT(DISTINCT salary)
FROM employees  e2
WHERE e2.salary <= e1.salary);
```

Qno16:-

Write a query to select last 10 records from a table.

Answer:-

```
SELECT * FROM (
SELECT * FROM employees ORDER BY employee_id DESC LIMIT 10) sub
ORDER BY employee_id ASC;
```

Qno17:-

Write a query to list department number, name for all the departments in which there are no employees in the department

Answer:-

```
SELECT * FROM departments
WHERE department_id
NOT IN (select department_id FROM employees);
```

Qno18:-

Write a query to get 3 maximum salaries.

Answer:-

```
SELECT DISTINCT salary
FROM employees a
WHERE 3 >= (SELECT COUNT(DISTINCT salary)
FROM employees b
WHERE b.salary >= a.salary)
ORDER BY a.salary DESC;
```

Qno19:-

Write a query to get 3 minimum salaries

Answer:-

```
SELECT DISTINCT salary
FROM employees a
WHERE  3 >= (SELECT COUNT(DISTINCT salary)
FROM employees b
WHERE b.salary <= a.salary)
ORDER BY a.salary DESC;
```

Qno20:-

Write a query to get nth max salaries of employees.Further practice with nested queries

Answer:-

```
SELECT *
FROM employees emp1
WHERE (1) = (
SELECT COUNT(DISTINCT(emp2.salary))
FROM employees emp2
WHERE emp2.salary > emp1.salary);
```

LAB11

Qno1:-

Create a table tow columns for name and family_name respectively. Insert the names your three friends in lower case case caracters. Write a query to create columns aliased fullname by using the INITCAT() and CONCAT() functions.

Answer:-

```
SELECT customer_id, CONCAT(first_name,second_name,last_name)AS All_names from customer
```

LAB12

Qno1:-

Print countrycode and sum of percentage from countrylangauge, apply groupby on countrycode.

Answer:-

```
SELECT countrycode ,SUM(PER(countrylanguage) from country language Group By(countrycode)
```

Qno2:-

Find sum of any integer column from country table.

Answer:-

```
Select SUM(population) from country table;
```

Qno3:-

Count number of records in country table

Answer:-

SELECT COUNT(records) from countrytable;

Qno4:-

Count Distinct (languages) from countrylanguage

Answer:-

SELECT DISTINCT(languages) from countrylanguage;

LAB13

Qno1:-

Select customers name, number , phone from customers table, select checknumber from payments table. Display it for all customers. [either they have made payment or they haven't include all customers].

Answer:-

SELECT customer,phone from customertable;

SELECT checknumber from payment;

LAB14

Qno1:-

Update customer with any a particular order number (you can select any order number).

```
UPDATE customer
SET order_no = 5;
```

Qno2:-

Applying union print data of orders and order details table.

Answer:-

```
SELECT data.customerdata from customers
UNION ALL
SELECT order.orderdata FROM orders;
```

LAB15

Qno1:-

Apply update on any country name.

Answer:-

UPDATE country SET column1=”England” WHERE column=”Islamabad”;

Qno2:-

Delete Islamabad city by applying delete query on city table with it’s ID

Answer:-

```
DELETE FROM city
WHERE country.id = ANY (SELECT id FROM city WHERE id = 2);
```

Qno3:-

Try to update values for null column COMM column

Answer:-

UPDATE[city]

SET [COMM]=0

WHERE [COMM] is null;

Qno4:-

Try to update it for a specific employee whose salary is less than 1000

Answer:-

```
UPDATE employee   WHERE employee<1000;
```

Qno5:-

Answer:-

```
INSERT INTO city
VALUES (“Abbottabad”, “Haripur”, “Mansehra”);
```

LAB16

Qno1:-

The first column is called supplier_id which is created as a number data type (maximum 10 digits in length) and cannot contain null values.

Answer:-

```
CREATE TABLE suppliers(supplier_id int(10) NOT NULL);
```

Qno2:-

The second column is called supplier_name which is a varchar2 datatype (50 maximum characters in length) and also can not contain null values

Answer:-

```
CREATE TABLE suppliers(supplier_id int(10) NOT NULL,supplier_name varchar2(50) NOT NULL);
```

Qno3:-

The third column is called address which is a varchar2 data type but can contain null values.

Answer:-

```
CREATE TABLE suppliers(supplier_id int(10) NOT NULL,supplier_name varchar(50) NOT
NULL,address varchar2 NOT NULL);
```

Qno4:-

Define the supplier_id as the primary key

Answer:-

CREATE TABLE suppliers(supplier_id NOT NULL AUTO_INCREMENT,supplier_id int(10) NOT NULL,supplier_name varchar(50) NOT NULL,address varchar2 NOT NULL,PRIMARY KEY(supplier_id);

Qno5:-

Create a second table named as Item with columns:

Answer:-

CREATE TABLE Item();

Qno6:-

The first column itemname any length you want

Answer:-

CREATE TABLE Item(itemname varchar(255);

Qno7:-

The second column supplierId as foreignkey in item table

Answer:-

```
CREATE TABLE item (
    itemname varchar(255),
    FOREIGN KEY (supplier_id) REFERENCES Persons(supplier_id)
);
```

Qno8:-

The third column should be itemprice In INT

Answer:-

```
CREATE TABLE item (
    itemname varchar(255),

    Itemprice int NOT NULL,
    FOREIGN KEY (supplier_id) REFERENCES Persons(supplier_id)
);
```

LAB20

Qno1:-

Write a SQL function to convert temperature from Fahrenheit to Celsius scale

Answer:-

```
DECLARE
  temp1    NUMBER := &input_a_temp;
  t_scale  CHAR := '&input_temp_scale';
  new_temp  NUMBER;
  new_scale CHAR;
 BEGIN
  IF t_scale != 'C'
   AND
   t_scale != 'F' THEN
   dbms_output.Put_line ('The scale you input is not a valid scale');
   new_temp := 0;
   new_scale := 'C';
  ELSE
   IF t_scale = 'C' THEN
    new_temp := ( ( 9 * temp1 ) / 5 ) + 32;
    new_scale := 'F';
   ELSE
    new_temp := ( ( temp1 - 32 ) * 5 ) / 9;
    new_scale := 'C';
   END IF;
  END IF;
```

```
   dbms_output.Put_line ('The new temperature in scale '

   ||new_scale

   ||' is: '

   ||new_temp);

  END;

/
```

Individual Solutions:-

1.Hanzala Shahid:-

RegNo:Fa18-BCS-014

Lab 6

Qno1:-

Answer:-

```
create database DreamHome;
USE DreamHome;
create table Branch
(
branchNo varchar(20) NOT NULL Primary Key,
street varchar(100) NOT NULL,
city varchar(50) NOT NULL,
postcode varchar(20) NOT NULL
);
create table Staff
(
staffNo varchar(20) NOT NULL PRIMARY KEY,
fName varchar(50) NOT NULL,
lName varchar(50) NOT NULL,
position varchar(50) NOT NULL,
sex varchar(1) NOT NULL, DOB DateTime NOT NULL,
salary DECIMAL NOT NULL,
branchNo varchar(20) NOT NULL References Branch(branchNo)
);
create table Client
(
clientNo varchar(20) NOT NULL PRIMARY KEY,
fName varchar(50) NOT NULL,
lName varchar(50) NOT NULL,
telNo varchar(20) NOT NULL,
prefType varchar(50) NOT NULL,
maxRent DECIMAL NOT NULL
);
create table PrivateOwner
(
ownerNo varchar(20) NOT NULL PRIMARY KEY,
fName varchar(50) NOT NULL,
lName varchar(50) NOT NULL,
address varchar(50) NOT NULL,
telNo varchar(20) NOT NULL
);
create table PropertyForRent
(
propertyNo varchar(20) NOT NULL PRIMARY KEY,
street varchar(100) NOT NULL,
city varchar(50) NOT NULL,
postcode varchar(20) NOT NULL,
type varchar(10) NOT NULL,
rooms int NOT NULL,
rent DECIMAL NOT NULL,
ownerNo varchar(20) References PrivateOwner(ownerNo),
staffNo varchar(20) NOT NULL References Staff(staffNo),
branchNo varchar(20) NOT NULL References Branch(branchNo)
);
create table Viewing
(
clientNo varchar(20) NOT NULL References Client(clientNo),
propertyNo varchar(20) NOT NULL References PropertyForRent(propertyNo),
viewDate DateTime NOT NULL,
comment varchar(200) NOT NULL
);
create table Registration
(
clientNo varchar(20) NOT NULL References Client(clientNo),
branchNo varchar(20) NOT NULL References Branch(branchNo),
staffNo varchar(20) NOT NULL References Staff(staffNo),
dateJoined DateTime NOT NULL
);


INSERT into Branch
(
branchNo,
street,
city,
postcode
)
VALUES
(N'B001',N'H#7 I-10/2', N'ISB', N'52000'),
(N'B002',N'H#78 Supply', N'ABT', N'53000'),
(N'B003',N'H#79 I-10/2', N'ISB', N'52000'),
(N'B004',N'H#78 Mandian', N'ABT', N'53000');

insert into Staff
(
staffNo,
fName,
lName,
position,
sex,
DOB,
salary,
branchNo
)
VALUES
(
N'SA9', N'Mary', N'Howe', N'Assistant', N'F',CAST(0x0000641000000000 AS DateTime),
CAST(9000 AS Decimal(18, 0)),N'B002'
),
(N'SG14', N'David', N'Ford', N'Supervisor', N'M',
CAST(0x0000531200000000 AS DateTime), CAST(18000 AS Decimal(18,0)),
N'B003'),
(N'SG37', N'Ann', N'Beech', N'Assistant', N'F',
CAST(0x000056D400000000 AS DateTime), CAST(12000 AS Decimal(18,0)),
N'B003'),
(N'SG5',N'Susan', N'Brand', N'Manager', N'F',
CAST(0x0000C85800000000 AS DateTime), CAST(24000 AS Decimal(18,0)),
N'B003'),
(N'SL21', N'John', N'White', N'Manager', N'M',
CAST(0x0000CFF200000000 AS DateTime), CAST(30000 AS Decimal(18,0)),
N'B004'),
(N'SL41', N'Julie', N'Lee', N'Assistant', N'F',
CAST(0x00005D6000000000 AS DateTime), CAST(9000 AS Decimal(18, 0)),
N'B002');


insert into Client
(
clientNo,
fName,
lName,
telNo,
prefType,
maxRent
)
values
(
'B1001','Mahad','Ali','030078601','yes',1000.0
),
(
'B1002','Sharjeel','Khan','030054621','yes',2000.0
),
(
'B1003','Hanzala','Shahid','030456601','no',1500.0
),
(
'B1004','Hamza','Aslam','0306446641','yes',8800.0
),
(
'B1005','Hamza','Ahmad','0354654401','noo',800.0
),
(
'B1006','Usama','Fareed','030074541','yes',4000.0
);

insert into PrivateOwner
(
ownerNo,
fName,
lName,
[address],
telNo
)
values
(
'B1','Azid','Ali','F18-4A','0354654264'
),
(
'B2','Mahad','Ali','F17-4A','0345154264'
),
(
'B3','Sharjeel','Khan','F14-7A','0352354264'
),
(
'B4','Hanzala','Shahid','F88-4A','0359354264'
),
(
'B5','Hamza','Aslam','F11-3A','0351694264'
),
(
'B6','Hamza','Ahmad','F19-5A','0354654264'
);


insert into PropertyForRent
(
propertyNo,street,city,postcode,[type],rooms,rent,ownerNo,staffNo,branchNo
)
values
(
'BF2','H2-h2','ABT','22010','large',8,'30000','B2','SG14','B002'
),
(
'BF3','H3-h3','ISB','62010','medium',6,'20000','B3','SG37','B003'
),
(
'BF4','H4-h4','ISB','62010','small',4,'10000','B4','SG5','B004'
);


insert into Viewing
(
clientNo,
propertyNo,
viewDate,
comment
)
values
(
'B1002','BF2','2020-2-10','No, i dont wanna say anything'
),
(
'B1003','BF3','2020-3-10','No, i dont wanna say anything'
),
(
'B1004','BF4','2020-4-10','No, i dont wanna say anything'
);


insert into Viewing
(
clientNo,
propertyNo,
viewDate,
comment
)
values
(
'B1002','BF2','2020-2-10','No, i dont wanna say anything'
),
(
'B1003','BF3','2020-3-10','No, i dont wanna say anything'
),
(
'B1004','BF4','2020-4-10','No, i dont wanna say anything'
);
```

End of qno1

Qno:-2

Answer:-

```
select * from Branch update Branch set city='ABT' where city='ISB';
```

Lab 7

Qno1:-

Answer

```
select distinct(postcode) from Branch
```

Qno2:-

Answer

```
select distinct(fName) from Staff
```

Qno3:-

Answer

```
select staffNo as [Cadre No], fName as [Baptism Name], lName as [Sur name],
position as [Locale], sex as [Gender],DOB as [Birtday] ,salary as Income,
branchNo as [Section No] from Staff
```

Qno4:-

Answer

```
select clientNo as [Buyer No], fName as [Baptism Name], lName as [Sur name],
telNo as [Fax Number],prefType as [Proclivity Type],maxRent as [Supreme Cost] from
Client;
```

Qno5:-

Answer

```
select * from Staff where salary>10000
```

Qno6:-

Answer

```
select * from Staff where position='Manager' or position='Supervisor'
```

Lab 8

Qno1:-

Answer

```
select staffNo,fName,lName,salary from staff order by salary desc
```

Qno2:-

Answer

```
select propertyNo,type,rooms,rent from PropertyForRent
order by type
select propertyNo,type,rooms,rent
from PropertyForRent
order by type,rent desc
```

Qno3:-

Answer

```
select count(*) as myCount
from PropertyForRent
where rent<=500
```

Qno4:-

Answer

```
select count(Distinct propertyNo) As myCount from Viewing
WHERE viewDate BETWEEN '1-May-04' AND '31-May-04';
```

Qno5:-

Answer

```
select count(staffNo) as myCount,sum(salary) as mySalary from staff
where
position='Manager'
```

Qno6:-

Answer

```
select MIN(salary) as myMin,
MAX(salary) as myMax,
AVG(salary) as myAVG from Staff
```

Qno7:-

Answer

```
SELECT staffNo, fName, lName, position, salary
FROM Staff
WHERE (SELECT AVG(salary) FROM Staff) < salary;
```

Qno8:-

Answer

```
select *from Staff where salary> any(select salary from Staff where branchNo='B003')
```

Qno9:-

Answer

```
select *from Staff where salary> all(select salary from Staff where branchNo='B003')
```

Lab 9

2. Write the following queries.

1.Write a query to find the names (first_name, last_name) and the salaries of the employees who have a higher salary than the employee who’s last_name=’Bull’.

Answer#

```
SELECT FIRST_NAME, LAST_NAME, SALARY
FROM employees
WHERE SALARY >
(SELECT salary FROM employees WHERE last_name = 'Bull');
```

2• Write a query to find the names (first_name, last_name) of all employeeswho works in the IT department.

Answer#

```
SELECT first_name, last_name
FROM employees
WHERE department_id
IN (SELECT department_id FROM departments WHERE department_name='IT');
```

Lab 10

Qno1:-

Write a query to find the names (first_name, last_name) of the employees who have a manager and work for a department based in the United States.

Answer:-

```
SELECT first_name, last_name FROM employees

WHERE manager_id in(select employee_id FROM employees WHERE department_id

IN(SELECT department_id FROM departments WHERE location_id IN(select location_id from locations

Where country_id=’US’)));
```

Qno2:-

Write a query to find the names (first_name, last_name) of the employees who are managers.

Answer:-

```
SELECT first_name, last_name

FROM employees

WHERE (employee_id IN(SELECT manager_id FROM employees));
```

Qno3:-

Write a query to find the names (first_name, last_name), the salary of the employees whose salary is greater than the average salary

Answer:-

```
SELECT first_name, last_name, salary FROM employees

WHERE salary > (SELECT AVG(salary) FROM employees);
```

Qno4:-

Write a query to find the names (first_name, last_name), the salary of the employees whose salary is equal to the minimum salary for their job grade.

Answer:-

```
SELECT first_name,last_name,salary FROM employees WHERE employees.salary=(SELECT min_salary
FROM jobs WHERE employees.job_id=jobs.job_id);
```

Qno5:-

Write a query to find the names (first_name, last_name), the salary of the employees who earn more than the average salary and who works in any of the IT departments.

Answer:-

```
SELECT first_name,last_name,salary

FROM employees WHERE department_id IN (SELECT department_id FROM departments WHERE
department_name LIKE ‘IT%’)AND salary>(SELECT avg(salary) From employees);
```

Qno6:-

Write a query to find the names (first_name, last_name), the salary of the employees who earn more than Mr. Bell

Answer:-

```
SELECT * FROM employees WHERE salary > ALL(SELECT AVG(salary) FROM employees GROUP BY

Department_id);
```

Qno7:-

Write a query to find the names (first_name, last_name), the salary of the employees who earn the same salary as the minimum salary for all departments

Answer:-

```
SELECT * FROM employees

WHERE salary=(SELECT MIN(salary) FROM employees);
```

Qno8:-

Write a query to find the names (first_name, last_name), the salary of the employees whose salary greater than the average salary of all departments

Answer:-

```
SELECT first_name,last_name from employees whose(SELECT AVG(salary) from departments)
```

Qno9:-

Write a query to find the names (first_name, last_name) and salary of the employees who earn a salary that is higher than the salary of all the Shipping Clerk (JOB_ID = ’SH_CLERK’). Sort the results of the salary of the lowest to highest

Answer:-

```
SELECT first_name,last_name, job_id, salary

FROM employees

WHERE salary >

ALL (SELECT salary FROM employees WHERE job_id = 'SH_CLERK') ORDER BY salary;
```

Qno10:-

.Write a query to find the names (first_name, last_name) of the employees who are not supervisors.

Answer:-

```
SELECT b.first_name,b.last_name

FROM employees b

WHERE NOT EXISTS (SELECT 'X' FROM employees a WHERE a.manager_id = b.employee_id);
```

Qno11:-

Write a query to display the employee ID, first name, last names, and department names of all employees.

Answer:-

```
SELECT employee_id, first_name, last_name,

(SELECT department_name FROM departments d

 WHERE e.department_id = d.department_id) department
 FROM employees e ORDER BY department;
```

Qno12:-

Write a query to display the employee ID, first name, last names, salary of all employees whose salary is above average for their departments

Answer:-

```
SELECT employee_id, first_name

FROM employees AS A

WHERE salary >

(SELECT AVG(salary) FROM employees WHERE department_id = A.department_id);
```

Qno13:-

Write a query to fetch even numbered records from employees table

Answer:-

```
SET @i = 0;

SELECT i, employee_id

FROM (SELECT @i := @i + 1 AS i, employee_id FROM employees)

a WHERE MOD(a.i, 2) = 0;
```

Qno14:-

Write a query to find the 5th maximum salary in the employees table.

Answer:-

```
SELECT DISTINCT salary

FROM employees e1

WHERE 5 = (SELECT COUNT(DISTINCT salary)

FROM employees  e2

WHERE e2.salary >= e1.salary);
```

Qno15:-

Write a query to find the 4th minimum salary in the employees table

Answer:-

```
SELECT DISTINCT salary
FROM employees e1
WHERE 4 = (SELECT COUNT(DISTINCT salary)
FROM employees  e2
WHERE e2.salary <= e1.salary);
```

Qno16:-

Write a query to select last 10 records from a table.

Answer:-

```
SELECT * FROM (
SELECT * FROM employees ORDER BY employee_id DESC LIMIT 10) sub
ORDER BY employee_id ASC;
```

Qno17:-

Write a query to list department number, name for all the departments in which there are no employees in the department

Answer:-

```
SELECT * FROM departments
WHERE department_id
NOT IN (select department_id FROM employees);
```

Qno18:-

Write a query to get 3 maximum salaries.

Answer:-

```
SELECT DISTINCT salary
```

```
FROM employees a
WHERE 3 >= (SELECT COUNT(DISTINCT salary)
FROM employees b
WHERE b.salary >= a.salary)
ORDER BY a.salary DESC;
```

Qno19:-

Write a query to get 3 minimum salaries

Answer:-

```
SELECT DISTINCT salary
FROM employees a
WHERE  3 >= (SELECT COUNT(DISTINCT salary)
FROM employees b
WHERE b.salary <= a.salary)
ORDER BY a.salary DESC;
```

Qno20:-

Write a query to get nth max salaries of employees.Further practice with nested queries

Answer:-

```
SELECT *
FROM employees emp1
WHERE (1) = (
SELECT COUNT(DISTINCT(emp2.salary))
FROM employees emp2
WHERE emp2.salary > emp1.salary);
```

LAB11

Qno1:-

Create a table tow columns for name and family_name respectively. Insert the names your three friends in lower case case caracters. Write a query to create columns aliased fullname by using the INITCAT() and CONCAT() functions.

Answer:-

SELECT customer_id, CONCAT(first_name,second_name,last_name)AS All_names from customer

LAB12

Qno1:-

Print countrycode and sum of percentage from countrylangauge, apply groupby on countrycode.

Answer:-

SELECT countrycode ,SUM(PER(countrylanguage) from country language Group By(countrycode)

Qno2:-

Find sum of any integer column from country table.

Answer:-

Select SUM(population) from country table;

Qno3:-

Count number of records in country table

Answer:-

SELECT COUNT(records) from countrytable;

Qno4:-

Count Distinct (languages) from countrylanguage

Answer:-

SELECT DISTINCT(languages) from countrylanguage;

LAB13

Qno1:-

Select customers name, number , phone from customers table, select checknumber from payments table. Display it for all customers. [either they have made payment or they haven’t include all customers].

Answer:-

SELECT customer,phone from customertable;

SELECT checknumber from payment;

LAB14

Qno1:-

Update customer with any a particular order number (you can select any order number).

```
UPDATE customer
SET order_no = 5;
```

Qno2:-

Applying union print data of orders and order details table.

Answer:-

```
SELECT data.customerdata from customers
UNION ALL
SELECT order.orderdata FROM orders;
```

LAB15

Qno1:-

Apply update on any country name.

Answer:-

UPDATE country SET column1=”England” WHERE column=”Islamabad”;

Qno2:-

Delete Islamabad city by applying delete query on city table with it’s ID

Answer:-

```
DELETE FROM city
WHERE country.id = ANY (SELECT id FROM city WHERE id = 2);
```

Qno3:-

Try to update values for null column COMM column

Answer:-

UPDATE[city]

SET [COMM]=0

WHERE [COMM] is null;

Qno4:-

Try to update it for a specific employee whose salary is less than 1000

Answer:-

UPDATE employee   WHERE employee<1000;

Qno5:-

Answer:-

```
INSERT INTO city
VALUES (“Rawalpindi”, “Sialkot”, “Mirpur”);
```

LAB16

Qno1:-

The first column is called supplier_id which is created as a number data type (maximum 10 digits in length) and cannot contain null values.

Answer:-

```
CREATE TABLE suppliers(supplier_id int(10) NOT NULL);
```

Qno2:-

The second column is called supplier_name which is a varchar2 datatype (50 maximum characters in length) and also can not contain null values

Answer:-

```
CREATE TABLE suppliers(supplier_id int(10) NOT NULL,supplier_name varchar2(50) NOT NULL);
```

Qno3:-

The third column is called address which is a varchar2 data type but can contain null values.

Answer:-

```
CREATE TABLE suppliers(supplier_id int(10) NOT NULL,supplier_name varchar(50) NOT NULL,address varchar2 NOT NULL);
```

Qno4:-

Define the supplier_id as the primary key

Answer:-

```
CREATE TABLE suppliers(supplier_id NOT NULL AUTO_INCREMENT,supplier_id int(10) NOT NULL,supplier_name varchar(50) NOT NULL,address varchar2 NOT NULL,PRIMARY KEY(supplier_id);
```

Qno5:-

Create a second table named as Item with columns:

Answer:-

```
CREATE TABLE Item();
```

Qno6:-

The first column itemname any length you want

Answer:-

```
CREATE TABLE Item(itemname varchar(255);
```

Qno7:-

The second column supplierId as foreignkey in item table

Answer:-

```
CREATE TABLE item (
    itemname varchar(255),
    FOREIGN KEY (supplier_id) REFERENCES Persons(supplier_id)
);
```

Qno8:-

The third column should be itemprice In INT

Answer:-

```
CREATE TABLE item (
    itemname varchar(255),

    Itemprice int NOT NULL,
    FOREIGN KEY (supplier_id) REFERENCES Persons(supplier_id)
);
```

LAB20

Qno1:-

Write a SQL function to convert temperature from Fahrenheit to Celsius scale

Answer:-

```
DECLARE

  temp1    NUMBER := &input_a_temp;

  t_scale   CHAR := '&input_temp_scale';

  new_temp  NUMBER;
  new_scale CHAR;
 BEGIN
  IF t_scale != 'C'
   AND
   t_scale != 'F' THEN
   dbms_output.Put_line ('The scale you input is not a valid scale');
   new_temp := 0;
   new_scale := 'C';
  ELSE
   IF t_scale = 'C' THEN
    new_temp := ( ( 9 * temp1 ) / 5 ) + 32;
    new_scale := 'F';
   ELSE
    new_temp := ( ( temp1 - 32 ) * 5 ) / 9;
    new_scale := 'C';
   END IF;
  END IF;
  dbms_output.Put_line ('The new temperature in scale '
  ||new_scale
  ||' is: '
  ||new_temp);
 END;
/
```

2-) Sharjeel Khan Niazi:

Reg No:FA18-BCS-019

Lab 6

Qno1: -

Answer: -

```
create database DreamHome;
USE DreamHome;
create table Branch
(
Branch_No varchar(18) NOT NULL Primary Key,
Street_No varchar(90) NOT NULL,
City_Name varchar(50) NOT NULL,
postcode int(20) NOT NULL
);
create table Staff
(
Staff_No varchar(20) NOT NULL PRIMARY KEY,
First_N varchar(50) NOT NULL,
Last_N varchar(50) NOT NULL,
position varchar(50) NOT NULL,
sex varchar(1) NOT NULL, DOB Date-Time NOT NULL,
salary DECIMAL NOT NULL,
branch_No varchar(20) NOT NULL References Branch(branch_No)
);
create table Client
(
Client_No varchar(20) NOT NULL PRIMARY KEY,
First_N varchar(50) NOT NULL,
Last_N varchar(50) NOT NULL,
Tel_No int(20) NOT NULL,
prefType varchar(50) NOT NULL,
maxRent DECIMAL NOT NULL
);
create table PrivateOwner
(
Owner_No varchar(20) NOT NULL PRIMARY KEY,
First_N varchar(50) NOT NULL,
Last_N varchar(50) NOT NULL,
address varchar(50) NOT NULL,
tel_No int(20) NOT NULL
);
create table PropertyForRent
(
Property_No varchar(20) NOT NULL PRIMARY KEY,
Street_NO varchar(100) NOT NULL,
City_Name varchar(50) NOT NULL,
postcode int(20) NOT NULL,
type varchar(10) NOT NULL,
rooms int NOT NULL,
rent DECIMAL NOT NULL,
owner_No varchar(20) References PrivateOwner(owner_No),
staff_No varchar(20) NOT NULL References Staff(staff_No),
branch_No varchar(20) NOT NULL References Branch(branch_No)
);
create table Viewing
(
Client_No varchar(20) NOT NULL References Client(client_No),
Property_No varchar(20) NOT NULL References PropertyForRent(property_No),
View-Date DateTime NOT NULL,
comment varchar(200) NOT NULL
```

```
);
create table Registration
(
Client_No varchar(20) NOT NULL References Client(client_No),
Branch_No varchar(20) NOT NULL References Branch(branch_No),
Staff_No varchar(20) NOT NULL References Staff(staff_No),
Date-Joined Date-Time NOT NULL
);


INSERT into Branch
(
Branch_No,
Street_no,
City_Name,
postcode
)
VALUES
(N'B001',N'H#7 I-10/2', N'ISB', N'52000'),
(N'B002',N'H#78 Supply', N'ABT', N'53000'),
(N'B003',N'H#79 I-10/2', N'ISB', N'52000'),
(N'B004',N'H#78 Mandian', N'ABT', N'53000');

insert into Staff
(
Staff_No,
First_N,
Last_N,
position,
sex,
DOB,
salary,
branch_No
)
VALUES
(
N'SA9', N'Mary', N'Howe', N'Assistant', N'F',CAST(0x0000641000000000 AS Date-Time),
CAST(9000 AS Decimal(18, 0)),N'B002'
),
(N'SG14', N'David', N'Ford', N'Supervisor', N'M',
CAST(0x0000531200000000 AS Date-Time), CAST(18000 AS Decimal(18,0)),
N'B003'),
(N'SG37', N'Ann', N'Beech', N'Assistant', N'F',
CAST(0x000056D400000000 AS Date-Time), CAST(12000 AS Decimal(18,0)),
N'B003'),
(N'SG5',N'Susan', N'Brand', N'Manager', N'F',
CAST(0x0000C85800000000 AS Date-Time), CAST(24000 AS Decimal(18,0)),
N'B003'),
(N'SL21', N'John', N'White', N'Manager', N'M',
CAST(0x0000CFF200000000 AS Date-Time), CAST(30000 AS Decimal(18,0)),
N'B004'),
(N'SL41', N'Julie', N'Lee', N'Assistant', N'F',
CAST(0x00005D6000000000 AS Date-Time), CAST(9000 AS Decimal(18, 0)),
N'B002');

insert into Client
(
Client_No,
First_N,
Last_N,
Tel_No,
prefType,
maxRent
)
```

```
values
(
'B1001','Mahad','Ali','030078601','yes',1000.0
),
(
'B1002','Sharjeel','Khan','030054621','yes',2000.0
),
(
'B1003','Hanzala','Shahid','030456601','no',1500.0
),
(
'B1004','Hamza','Aslam','0306446641','yes',8800.0
),
(
'B1005','Hamza','Ahmad','0354654401','noo',800.0
),
(
'B1006','Usama','Fareed','030074541','yes',4000.0
),
(
'B1007','Abdullah','Niazi','03359613366','yes',2500.0
);

insert into PrivateOwner
(
Owner_No,
First_n,
Last_n,
[address],
Tel_No
)
values
```

```
(
'B1','Azid','Ali','F18-4A','0354654264'
),
(
'B2','Mahad','Ali','F17-4A','0345154264'
),
(
'B3','Sharjeel','Khan','F14-7A','0352354264'
),
(
'B4','Hanzala','Shahid','F88-4A','0359354264'
),
(
'B5','Hamza','Aslam','F11-3A','0351694264'
),
(
'B6','Hamza','Ahmad','F19-5A','0354654264'
)’
(
'B7','Abdullah','Niazi','F18-5A','03359613366'
);

insert into PropertyForRent
(
Property_No,street_NO,city_Name,postcode,[type],rooms,rent,owner_No,staff_No,branch_No
)
values
(
'BF2','H2-h2','ABT','22010','large',8,'30000','B2','SG14','B002'
),
(
'BF3','H3-h3','ISB','62010','medium',6,'20000','B3','SG37','B003'
```

```sql
),
(
'BF4','H4-h4','ISB','62010','small',4,'10000','B4','SG5','B004'
);

insert into Viewing
(
Client_No,
Property_No,
View-Date,
comment
)
values
(
'B1002','BF2','2020-2-10','No, i dont wanna say anything'
),
(
'B1003','BF3','2020-3-10','No, i dont wanna say anything'
),
(
'B1004','BF4','2020-4-10','No, i dont wanna say anything'
);

insert into Viewing
(
Client_No,
Property_No,
View-Date,
comment
)
values
(
'B1002','BF2','2020-2-10','No, i dont wanna say anything'

),

(

'B1003','BF3','2020-3-10','No, i dont wanna say anything'

),

(

'B1004','BF4','2020-4-10','No, i dont wanna say anything'

);
```

End of qno1

Qno:-2

Answer:-

```
select * from Branch update Branch set city='ABT' where city='ISB';
```

Lab 7:-

```
select distinct(postcode) from Branch

select distinct(fName) from Staff

select staffNo as [Cadre No], fName as [Baptism Name], lName as [Sur name],
position as [Locale], sex as [Gender],DOB as [Birtday] ,salary as Income,
branchNo as [Section No] from Staff

select clientNo as [Buyer No], fName as [Baptism Name], lName as [Sur name],
telNo as [Fax Number],prefType as [Proclivity Type],maxRent as [Supreme Cost] from
Client;

select * from Staff where salary>10000

select * from Staff where position='Manager' or position='Supervisor'
```

Lab 8

```sql
select staffNo,fName,lName,salary from staff order by salary desc

select propertyNo,type,rooms,rent from PropertyForRent
order by type
select propertyNo,type,rooms,rent
from PropertyForRent
order by type,rent desc

select count(*) as myCount
from PropertyForRent
where rent<=500

select count(Distinct propertyNo) As myCount from Viewing
WHERE viewDate BETWEEN '1-May-04' AND '31-May-04';

select count(staffNo) as myCount,sum(salary) as mySalary from staff
where
position='Manager'

select MIN(salary) as myMin,
MAX(salary) as myMax,
AVG(salary) as myAVG from Staff

SELECT staffNo, fName, lName, position, salary
FROM Staff
WHERE (SELECT AVG(salary) FROM Staff) < salary;

select *from Staff where salary> any(select salary from Staff where branchNo='B003')

select *from Staff where salary> all(select salary from Staff where branchNo='B003')
```

LAB 9

Qno1:

list all tables in the employees database

Answer:

USE EMPLOYEE;

show TABLES;

Qno2:

Write a query to find the names (first_name, last_name) and the salaries of the employees who have a higher salary than the employee whose last_name='Bull'.

Answer:-

Select FIRST_NAME, LAST_NAME, SALARY

```
FROM employees

WHERE SALARY>(SELECT salary FROM employees WHERE  last_name=’Bull’);
```

Qno3:

Answer:-

```
Select first_name , last_name

FROM employees

WHERE department_id

IN(SELECT department_id FROM departments WHERE department_name=’IT’);
```

LAB 10

Qno1:-

Write a query to find the names (first_name, last_name) of the employees who have a manager and work for a department based in the United States.

Answer:-

```
SELECT first_name, last_name FROM employees

WHERE manager_id in(select employee_id FROM employees WHERE department_id

IN(SELECT department_id FROM departments WHERE location_id IN(select location_id from locations

Where country_id=’US’)));
```

Qno2:-

Write a query to find the names (first_name, last_name) of the employees who are managers.

Answer:-

```
SELECT first_name, last_name

FROM employees

WHERE (employee_id IN(SELECT manager_id FROM employees));
```

Qno3:-

Write a query to find the names (first_name, last_name), the salary of the employees whose salary is greater than the average salary

Answer:-

```
SELECT first_name, last_name, salary FROM employees

WHERE salary > (SELECT AVG(salary) FROM employees);
```

Qno4:-

Write a query to find the names (first_name, last_name), the salary of the employees whose salary is equal to the minimum salary for their job grade.

Answer:-

```
SELECT first_name,last_name,salary FROM employees WHERE employees.salary=(SELECT min_salary
FROM jobs WHERE employees.job_id=jobs.job_id);
```

Qno5:-

Write a query to find the names (first_name, last_name), the salary of the employees who earn more than the average salary and who works in any of the IT departments.

Answer:-

```
SELECT first_name,last_name,salary

FROM employees WHERE department_id IN (SELECT department_id FROM departments WHERE
department_name LIKE ‘IT%’)AND salary>(SELECT avg(salary) From employees);
```

Qno6:-

Write a query to find the names (first_name, last_name), the salary of the employees who earn more than Mr. Bell

Answer:-

```
SELECT * FROM employees WHERE salary > ALL(SELECT AVG(salary) FROM employees GROUP BY
Department_id);
```

Qno7:-

Write a query to find the names (first_name, last_name), the salary of the employees who earn the same salary as the minimum salary for all departments

Answer:-

```
SELECT * FROM employees

WHERE salary=(SELECT MIN(salary) FROM employees);
```

Qno8:-

Write a query to find the names (first_name, last_name), the salary of the employees whose salary greater than the average salary of all departments

Answer:-

```
SELECT first_name,last_name from employees whose(SELECT AVG(salary) from departments)
```

Qno9:-

Write a query to find the names (first_name, last_name) and salary of the employees who earn a salary that is higher than the salary of all the Shipping Clerk (JOB_ID = ’SH_CLERK’). Sort the results of the salary of the lowest to highest

Answer:-

```
SELECT first_name,last_name, job_id, salary

FROM employees

WHERE salary >

ALL (SELECT salary FROM employees WHERE job_id = 'SH_CLERK') ORDER BY salary;
```

Qno10:-

.Write a query to find the names (first_name, last_name) of the employees who are not supervisors.

Answer:-

```
SELECT b.first_name,b.last_name
```

```
FROM employees b

WHERE NOT EXISTS (SELECT 'X' FROM employees a WHERE a.manager_id = b.employee_id);
```

Qno11:-

Write a query to display the employee ID, first name, last names, and department names of all employees.

Answer:-

```
SELECT employee_id, first_name, last_name,

(SELECT department_name FROM departments d

 WHERE e.department_id = d.department_id) department

 FROM employees e ORDER BY department;
```

Qno12:-

Write a query to display the employee ID, first name, last names, salary of all employees whose salary is above average for their departments

Answer:-

```
SELECT employee_id, first_name

FROM employees AS A

WHERE salary >

(SELECT AVG(salary) FROM employees WHERE department_id = A.department_id);
```

Qno13:-

Write a query to fetch even numbered records from employees table

Answer:-

```
SET @i = 0;

SELECT i, employee_id

FROM (SELECT @i := @i + 1 AS i, employee_id FROM employees)

a WHERE MOD(a.i, 2) = 0;
```

Qno14:-

Write a query to find the 5th maximum salary in the employees table.

Answer:-

```
SELECT DISTINCT salary
FROM employees e1
WHERE 5 = (SELECT COUNT(DISTINCT salary)
FROM employees  e2
WHERE e2.salary >= e1.salary);
```

Qno15:-

Write a query to find the 4th minimum salary in the employees table

Answer:-

```
SELECT DISTINCT salary
FROM employees e1
WHERE 4 = (SELECT COUNT(DISTINCT salary)
FROM employees  e2
WHERE e2.salary <= e1.salary);
```

Qno16:-

Write a query to select last 10 records from a table.

Answer:-

```
SELECT * FROM (
SELECT * FROM employees ORDER BY employee_id DESC LIMIT 10) sub
ORDER BY employee_id ASC;
```

Qno17:-

Write a query to list department number, name for all the departments in which there are no employees in the department

Answer:-

```
SELECT * FROM departments
WHERE department_id
NOT IN (select department_id FROM employees);
```

Qno18:-

Write a query to get 3 maximum salaries.

Answer:-

```
SELECT DISTINCT salary
FROM employees a
WHERE 3 >= (SELECT COUNT(DISTINCT salary)
FROM employees b
WHERE b.salary >= a.salary)
ORDER BY a.salary DESC;
```

Qno19:-

Write a query to get 3 minimum salaries

Answer:-

```
SELECT DISTINCT salary
FROM employees a
WHERE  3 >= (SELECT COUNT(DISTINCT salary)
FROM employees b
WHERE b.salary <= a.salary)
ORDER BY a.salary DESC;
```

Qno20:-

Write a query to get nth max salaries of employees.Further practice with nested queries

Answer:-

```
SELECT *

FROM employees emp1

WHERE (1) = (

SELECT COUNT(DISTINCT(emp2.salary))

FROM employees emp2

WHERE emp2.salary > emp1.salary);
```

LAB11

Qno1:-

Create a table tow columns for name and family_name respectively. Insert the names your three friends in lower case case caracters. Write a query to create columns aliased fullname by using the INITCAT() and CONCAT() functions.

Answer:-

```
SELECT customer_id, CONCAT(first_name,second_name,last_name)AS All_names from customer
```

LAB12

Qno1:-

Print countrycode and sum of percentage from countrylangauge, apply groupby on countrycode.

Answer:-

SELECT countrycode ,SUM(PER(countrylanguage) from country language Group By(countrycode)

Qno2:-

Find sum of any integer column from country table.

Answer:-

Select SUM(population) from country table;

Qno3:-

Count number of records in country table

Answer:-

SELECT COUNT(records) from countrytable;

Qno4:-

Count Distinct (languages) from countrylanguage

Answer:-

SELECT DISTINCT(languages) from countrylanguage;

LAB13

Qno1:-

Select customers name, number , phone from customers table, select checknumber from payments table. Display it for all customers. [either they have made payment or they haven't include all customers].

Answer:-

SELECT customer,phone from customertable;

SELECT checknumber from payment;

Qno2:-

Display orderdetails for products. Use inner join.

Answer:-

LAB14

Qno1:-

Update customer with any a particular order number (you can select any order number).

```
UPDATE customer
SET order_no = 5;
```

Qno2:-

Applying union print data of orders and order details table.

Answer:-

```
SELECT data.customerdata from customers
UNION ALL
SELECT order.orderdata FROM orders;
```

LAB15

Qno1:-

Apply update on any country name.

Answer:-

```
UPDATE country SET column1=”England”  WHERE column=”Islamabad”;
```

Qno2:-

Delete Islamabad city by applying delete query on city table with it’s ID

Answer:-

```
DELETE FROM city
WHERE country.id = ANY (SELECT id FROM city WHERE id = 2);
```

Qno3:-

Try to update values for null column COMM column

Answer:-

```
UPDATE[city]

SET [COMM]=0

WHERE [COMM] is null;
```

Qno4:-

Try to update it for a specific employee whose salary is less than 1000

Answer:-

```
UPDATE employee   WHERE employee<1000;
```

Qno5:-

Answer:-

```
INSERT INTO city
VALUES (“Rawalpindi”, “Sialkot”, “Mirpur”);
```

LAB16

Qno1:-

The first column is called supplier_id which is created as a number data type (maximum 10 digits in length) and cannot contain null values.

Answer:-

CREATE TABLE suppliers(supplier_id int(10) NOT NULL);

Qno2:-

The second column is called supplier_name which is a varchar2 datatype (50 maximum characters in length) and also can not contain null values

Answer:-

CREATE TABLE suppliers(supplier_id int(10) NOT NULL,supplier_name varchar2(50) NOT NULL);

Qno3:-

The third column is called address which is a varchar2 data type but can contain null values.

Answer:-

CREATE TABLE suppliers(supplier_id int(10) NOT NULL,supplier_name varchar(50) NOT NULL,address varchar2 NOT NULL);

Qno4:-

Define the supplier_id as the primary key

Answer:-

CREATE TABLE suppliers(supplier_id NOT NULL AUTO_INCREMENT,supplier_id int(10) NOT NULL,supplier_name varchar(50) NOT NULL,address varchar2 NOT NULL,PRIMARY KEY(supplier_id);

Qno5:-

Create a second table named as Item with columns:

Answer:-

CREATE TABLE Item();

Qno6:-

The first column itemname any length you want

Answer:-

CREATE TABLE Item(itemname varchar(255);

Qno7:-

The second column supplierId as foreignkey in item table

Answer:-

```
CREATE TABLE item (
    itemname varchar(255),
    FOREIGN KEY (supplier_id) REFERENCES Persons(supplier_id)
);
```

Qno8:-

The third column should be itemprice In INT

Answer:-

```
CREATE TABLE item (
    itemname varchar(255),

    Itemprice int NOT NULL,
    FOREIGN KEY (supplier_id) REFERENCES Persons(supplier_id)
);
```

LAB20

Qno1:-

Write a SQL function to convert temperature from Fahrenheit to Celsius scale

Answer:-

```
DECLARE
  temp1     NUMBER := &input_a_temp;
  t_scale   CHAR := '&input_temp_scale';
  new_temp  NUMBER;
  new_scale CHAR;
 BEGIN
  IF t_scale != 'C'
   AND
   t_scale != 'F' THEN
   dbms_output.Put_line ('The scale you input is not a valid scale');
   new_temp := 0;
   new_scale := 'C';
  ELSE
   IF t_scale = 'C' THEN
    new_temp := ( ( 9 * temp1 ) / 5 ) + 32;
    new_scale := 'F';
   ELSE
    new_temp := ( ( temp1 - 32 ) * 5 ) / 9;
    new_scale := 'C';
   END IF;
  END IF;
  dbms_output.Put_line ('The new temperature in scale '
  ||new_scale
  ||' is: '
  ||new_temp);
 END;
/
```

Abdullah Niazi

Fa18-bcs-004

Lab 6

Qno1:-

Answer:-

```
create database HomeSweetHome;

USE HomeSweetHome;

create table Branch

(

branchNumber varchar(25) NOT NULL Primary Key,

streetNumber varchar(110) NOT NULL,

cityName varchar(60) NOT NULL,

postcodeNumber int NOT NULL

);

create table Staff

(

staffNumber int NOT NULL PRIMARY KEY,

firstName varchar(50) NOT NULL,

lastName varchar(50) NOT NULL,

position varchar(50) NOT NULL,

sex char NOT NULL, DOB DateTime NOT NULL,

salary DECIMAL NOT NULL,

branchNumber varchar(25) NOT NULL References Branch(branchNumber)
```

```
);
create table Client
(
clientNumber varchar(20) NOT NULL PRIMARY KEY,
firstName varchar(50) NOT NULL,
lastName varchar(50) NOT NULL,
phoneNo int NOT NULL,
prefType varchar(50) NOT NULL,
maxRent DECIMAL NOT NULL
);
create table PrivateOwner
(
ownerNumber varchar(20) NOT NULL PRIMARY KEY,
firstName varchar(50) NOT NULL,
lastName varchar(50) NOT NULL,
address varchar(50) NOT NULL,
phoneNo int NOT NULL
);
create table PropertyForRent
(
propertyNumber varchar(20) NOT NULL PRIMARY KEY,
streetNumber varchar(100) NOT NULL,
cityName varchar(50) NOT NULL,
postcodeNumber int NOT NULL,
type varchar(10) NOT NULL,
rooms int NOT NULL,
rent DECIMAL NOT NULL,
ownerNumber varchar(20) References PrivateOwner(ownerNumber),
staffNumber varchar(20) NOT NULL References Staff(staffNumber),
branchNumber varchar(20) NOT NULL References Branch(branchNumber)
);
create table Viewing
(
clientNumber varchar(20) NOT NULL References Client(clientNumber),
propertyNumber varchar(20) NOT NULL References
PropertyForRent(propertyNumber),
viewDate DateTime NOT NULL,
comment varchar(200) NOT NULL
);
create table Registration
(
clientNumber varchar(20) NOT NULL References Client(clientNumber),
branchNumber varchar(20) NOT NULL References Branch(branchNumber),
staffNumber varchar(20) NOT NULL References Staff(staffNumber),
dateJoined DateTime NOT NULL
);


INSERT into Branch
(
branchNumber,
streetNumber,
cityNumber,
postcodeNumber
)
VALUES
(N'B001',N'H#7 I-10/2', N'ISB', N'52000'),
(N'B002',N'H#78 Supply', N'ABT', N'53000'),
(N'B003',N'H#79 I-10/2', N'ISB', N'52000'),
(N'B004',N'H#78 Mandian', N'ABT', N'53000');

insert into Staff
(
staffNumber,
firstName,
lastName,
position,
sex,
DOB,
salary,
branchNumber
)
VALUES
(
N'SA9', N'Mary', N'Howe', N'Assistant', N'F',CAST(0x0000641000000000 AS DateTime),
CAST(9000 AS Decimal(18, 0)),N'B002'
```

```
),
(N'SG14', N'David', N'Ford', N'Supervisor', N'M',
CAST(0x0000531200000000 AS DateTime), CAST(18000 AS Decimal(18,0)),
N'B003'),
(N'SG37', N'Ann', N'Beech', N'Assistant', N'F',
CAST(0x000056D400000000 AS DateTime), CAST(12000 AS Decimal(18,0)),
N'B003'),
(N'SG5',N'Susan', N'Brand', N'Manager', N'F',
CAST(0x0000C85800000000 AS DateTime), CAST(24000 AS Decimal(18,0)),
N'B003'),
(N'SL21', N'John', N'White', N'Manager', N'M',
CAST(0x0000CFF200000000 AS DateTime), CAST(30000 AS Decimal(18,0)),
N'B004'),
(N'SL41', N'Julie', N'Lee', N'Assistant', N'F',
CAST(0x00005D6000000000 AS DateTime), CAST(9000 AS Decimal(18, 0)),
N'B002');

insert into Client
(
clientNumber,
firstName,
lastName,
PhoneNumber,
prefType,
maxRent
)
```

```
values
(
'B1001','Mahad','Ali','030078601','yes',1000.0
),
(
'B1002','Sharjeel','Khan','030054621','yes',2000.0
),
(
'B1003','Hanzala','Shahid','030456601','no',1500.0
),
(
'B1004','Hamza','Aslam','0306446641','yes',8800.0
),
(
'B1005','Hamza','Ahmad','0354654401','noo',800.0
),
(
'B1006','Usama','Fareed','030074541','yes',4000.0
),
(
'B1007','Abdullah','Niazi','03359613366','yes',2500.0
);
insert into PrivateOwner
(
ownerNumber,
firstName,
lastName,
[address],
PhoneNumber
)
values
(
'B1','Azid','Ali','F18-4A','0354654264'
),
(
'B2','Mahad','Ali','F17-4A','0345154264'
),
(
'B3','Sharjeel','Khan','F14-7A','0352354264'
),
(
'B4','Hanzala','Shahid','F88-4A','0359354264'
),
(
'B5','Hamza','Aslam','F11-3A','0351694264'
),
(
'B6','Hamza','Ahmad','F19-5A','0354654264'
)
(
'B7','Abdullah','Niazi','F18-5A','03359613366'
);

insert into PropertyForRent
(
propertyNumber,streetNumber,cityName,postcodeNumber,[type],rooms,rent,ownerNumnber,staffNumber,branchNumber
)
values
(
'BF2','H2-h2','ABT','22010','large',8,'30000','B2','SG14','B002'
),
(
'BF3','H3-h3','ISB','62010','medium',6,'20000','B3','SG37','B003'
),
(
'BF4','H4-h4','ISB','62010','small',4,'10000','B4','SG5','B004'
);

insert into Viewing
(
clientNumber,
propertyNumber,
viewDate,
comment
)
```

```
values
(
'B1002','BF2','2020-2-10','No, i dont wanna say anything'
),
(
'B1003','BF3','2020-3-10','No, i dont wanna say anything'
),
(
'B1004','BF4','2020-4-10','No, i dont wanna say anything'
);

insert into Viewing
(
clientNumber,
propertyNumber,
viewDate,
comment
)
values
(
'B1002','BF2','2020-2-10','No, i dont wanna say anything'
),
(
'B1003','BF3','2020-3-10','No, i dont wanna say anything'
),
(
'B1004','BF4','2020-4-10','No, i dont wanna say anything'
);
```

Qno:-2

Answer:-

```
select * from Branch update Branch set city='ABT' where city='ISB';
```

Lab 7

Qno1:-

Answer

```
select distinct(postcodeNumber) from Branch
```

Qno2:-

Answer

```
select distinct(firstName) from Staff
```

Qno3:-

Answer

```
select staffNumber as [Cadre No], firstName as [Baptism Name], lastName as [Sur name],
position as [Locale], sex as [Gender],DOB as [Birtday] ,salary as Income,
branchNumber as [Section No] from Staff
```

Qno4:-

Answer

```sql
select clientNumber as [Buyer No], firstName as [Baptism Name], lastName as [Sur name],

PhoneNumber as [Fax Number],prefType as [Proclivity Type],maxRent as [Supreme Cost] from Client;
```

Qno5:-

Answer

```sql
select * from Staff where salary>10000
```

Qno6:-

Answer

```sql
select * from Staff where position='Manager' or position='Supervisor'
```

Lab 8

Qno1:-

Answer

```sql
select staffNumber,firstName,lastName,salary from staff order by salary desc
```

Qno2:-

Answer

```sql
select propertyNumber,type,rooms,rent from PropertyForRent

order by type

select propertyNumber,type,rooms,rent

from PropertyForRent

order by type,rent desc
```

Qno3:-

Answer

```
select count(*) as myCount
from PropertyForRent
where rent<=500
```

Qno4:-

Answer

```
select count(Distinct propertyNumber) As myCount from Viewing
WHERE viewDate BETWEEN '1-May-04' AND '31-May-04';
```

Qno5:-

Answer

```
select count(staffNumber) as myCount,sum(salary) as mySalary from staff
where
position='Manager'
```

Qno6:-

Answer

```
select MIN(salary) as myMin,
MAX(salary) as myMax,
AVG(salary) as myAVG from Staff
```

Qno7:-

Answer

```
SELECT staffNumber, firstName, lastName, position, salary
FROM Staff
```

```
WHERE (SELECT AVG(salary) FROM Staff) < salary;
```

Qno8:-

Answer

```
select *from Staff where salary> any(select salary from Staff where
branchNumber='B003')
```

Qno9:-

Answer

```
select *from Staff where salary> all(select salary from Staff where
branchNumber='B003')
```

LAB 9

Qno1:

list all tables in the employees database

Answer:

```
 USE EMPLOYEE;
show TABLES;
```

Qno2:

Write a query to find the names (first_name, last_name) and the salaries of the employees who have a higher salary than the employee whose last_name='Bull'.

Answer:-

```
Select FIRST_NAME, LAST_NAME, SALARY
```

```
FROM employees
WHERE SALARY>(SELECT salary FROM employees WHERE last_name=’Bull’);
```

Qno3:

Answer:-

```
Select first_name , last_name
FROM employees
WHERE department_id
IN(SELECT department_id FROM departments WHERE department_name=’IT’);
```

LAB 10

Qno1:-

Write a query to find the names (first_name, last_name) of the employees who have a manager and work for a department based in the United States.

Answer:-

```
SELECT first_name, last_name FROM employees
WHERE manager_id in(select employee_id FROM employees WHERE department_id
IN(SELECT department_id FROM departments WHERE location_id IN(select location_id from locations
Where country_id=’US’)));
```

Qno2:-

Write a query to find the names (first_name, last_name) of the employees who are managers.

Answer:-

```
SELECT first_name, last_name

FROM employees

WHERE (employee_id IN(SELECT manager_id FROM employees));
```

Qno3:-

Write a query to find the names (first_name, last_name), the salary of the employees whose salary is greater than the average salary

Answer:-

```
SELECT first_name, last_name, salary FROM employees

WHERE salary > (SELECT AVG(salary) FROM employees);
```

Qno4:-

Write a query to find the names (first_name, last_name), the salary of the employees whose salary is equal to the minimum salary for their job grade.

Answer:-

```
SELECT first_name,last_name,salary FROM employees WHERE
employees.salary=(SELECT min_salary FROM jobs WHERE
employees.job_id=jobs.job_id);
```

Qno5:-

Write a query to find the names (first_name, last_name), the salary of the employees who earn more than the average salary and who works in any of the IT departments.

Answer:-

```
SELECT first_name,last_name,salary

FROM employees WHERE department_id IN (SELECT department_id FROM
departments WHERE department_name LIKE ‘IT%’)AND salary>(SELECT
avg(salary) From employees);
```

Qno6:-

Write a query to find the names (first_name, last_name), the salary of the employees who earn more than Mr. Bell

Answer:-

```
SELECT * FROM employees WHERE salary > ALL(SELECT AVG(salary)
FROM employees GROUP BY

Department_id);
```

Qno7:-

Write a query to find the names (first_name, last_name), the salary of the employees who earn the same salary as the minimum salary for all departments

Answer:-

```
SELECT * FROM employees

WHERE salary=(SELECT MIN(salary) FROM employees);
```

Qno8:-

Write a query to find the names (first_name, last_name), the salary of the employees whose salary greater than the average salary of all departments

Answer:-

```
SELECT first_name,last_name from employees whose(SELECT AVG(salary)
from departments)
```

Qno9:-

Write a query to find the names (first_name, last_name) and salary of the employees who earn a salary that is higher than the salary of all the Shipping Clerk (JOB_ID = ’SH_CLERK’). Sort the results of the salary of the lowest to highest

Answer:-

```
SELECT first_name,last_name, job_id, salary

FROM employees

WHERE salary >

ALL (SELECT salary FROM employees WHERE job_id = 'SH_CLERK')
ORDER BY salary;
```

Qno10:-

.Write a query to find the names (first_name, last_name) of the employees who are not supervisors.

Answer:-

```
SELECT b.first_name,b.last_name

FROM employees b

WHERE NOT EXISTS (SELECT 'X' FROM employees a WHERE a.manager_id
= b.employee_id);
```

Qno11:-

Write a query to display the employee ID, first name, last names, and department names of all employees.

Answer:-

```
SELECT employee_id, first_name, last_name,
(SELECT department_name FROM departments d
 WHERE e.department_id = d.department_id) department
 FROM employees e ORDER BY department;
```

Qno12:-

Write a query to display the employee ID, first name, last names, salary of all employees whose salary is above average for their departments

Answer:-

```
SELECT employee_id, first_name
FROM employees AS A
WHERE salary >
(SELECT AVG(salary) FROM employees WHERE department_id =
A.department_id);
```

Qno13:-

Write a query to fetch even numbered records from employees table

Answer:-

```
SET @i = 0;
SELECT i, employee_id
FROM (SELECT @i := @i + 1 AS i, employee_id FROM employees)
a WHERE MOD(a.i, 2) = 0;
```

Qno14:-

Write a query to find the 5th maximum salary in the employees table.

Answer:-

```
SELECT DISTINCT salary
```

```
FROM employees e1
WHERE 5 = (SELECT COUNT(DISTINCT salary)
FROM employees  e2
WHERE e2.salary >= e1.salary);
```

Qno15:-

Write a query to find the 4th minimum salary in the employees table

Answer:-

```
SELECT DISTINCT salary
FROM employees e1
WHERE 4 = (SELECT COUNT(DISTINCT salary)
FROM employees  e2
WHERE e2.salary <= e1.salary);
```

Qno16:-

Write a query to select last 10 records from a table.

Answer:-

```
SELECT * FROM (
SELECT * FROM employees ORDER BY employee_id DESC LIMIT 10) sub
ORDER BY employee_id ASC;
```

Qno17:-

Write a query to list department number, name for all the departments in which there are no employees in the department

Answer:-

```
SELECT * FROM departments
```

```
WHERE department_id
NOT IN (select department_id FROM employees);
```

Qno18:-

Write a query to get 3 maximum salaries.

Answer:-

```
SELECT DISTINCT salary
FROM employees a
WHERE 3 >= (SELECT COUNT(DISTINCT salary)
FROM employees b
WHERE b.salary >= a.salary)
ORDER BY a.salary DESC;
```

Qno19:-

Write a query to get 3 minimum salaries

Answer:-

```
SELECT DISTINCT salary
FROM employees a
WHERE  3 >= (SELECT COUNT(DISTINCT salary)
FROM employees b
WHERE b.salary <= a.salary)
ORDER BY a.salary DESC;
```

Qno20:-

Write a query to get nth max salaries of employees.Further practice with nested queries

Answer:-

```
SELECT *
FROM employees emp1
WHERE (1) = (
SELECT COUNT(DISTINCT(emp2.salary))
FROM employees emp2
WHERE emp2.salary > emp1.salary);
```

LAB11

Qno1:-

Create a table tow columns for name and family_name respectively. Insert the names your three friends in lower case case caracters. Write a query to create columns aliased fullname by using the INITCAT() and CONCAT() functions.

Answer:-

```
SELECT customer_id, CONCAT(first_name,second_name,last_name)AS
All_names from customer
```

LAB12

Qno1:-

Print countrycode and sum of percentage from countrylangauge, apply groupby on countrycode.

Answer:-

```
SELECT countrycode ,SUM(PER(countrylanguage) from country language Group By(countrycode)
```

Qno2:-

Find sum of any integer column from country table.

Answer:-

```
Select SUM(population) from country table;
```

Qno3:-

Count number of records in country table

Answer:-

```
SELECT COUNT(records) from countrytable;
```

Qno4:-

Count Distinct (languages) from countrylanguage

Answer:-

```
SELECT DISTINCT(languages) from countrylanguage;
```

LAB13

Qno1:-

Select customers name, number , phone from customers table, select checknumber from payments table. Display it for all customers. [either they have made payment or they haven’t include all customers].

Answer:-

SELECT customer,phone from customertable;

SELECT checknumber from payment;

Qno2:-

Display orderdetails for products. Use inner join.

Answer:-

LAB14

Qno1:-

Update customer with any a particular order number (you can select any order number).

```
UPDATE customer
SET order_no = 5;
```

Qno2:-

Applying union print data of orders and order details table.

Answer:-

```
SELECT data.customerdata from customers
UNION ALL
SELECT order.orderdata FROM orders;
```

LAB15

Qno1:-

Apply update on any country name.

Answer:-

```
UPDATE country SET column1=”England”  WHERE column=”Islamabad”;
```

Qno2:-

Delete Islamabad city by applying delete query on city table with it’s ID

Answer:-

```
DELETE FROM city
WHERE country.id = ANY (SELECT id FROM city WHERE id = 2);
```

Qno3:-

Try to update values for null column COMM column

Answer:-

```
UPDATE[city]
```

```
SET [COMM]=0
WHERE [COMM] is null;
```

Qno4:-

Try to update it for a specific employee whose salary is less than 1000

Answer:-

```
UPDATE employee   WHERE employee<1000;
```

Qno5:-

Try to insert values in customers table

Answer:-

```
INSERT INTO city
VALUES (“Rawalpindi”, “Sialkot”, “Mirpur”);
```

LAB16

Qno1:-

The first column is called supplier_id which is created as a number data type (maximum 10 digits in length) and cannot contain null values.

Answer:-

```
CREATE TABLE suppliers(supplier_id int(10) NOT NULL);
```

Qno2:-

The second column is called supplier_name which is a varchar2 datatype (50 maximum characters in length) and also can not contain null values

Answer:-

```
CREATE TABLE suppliers(supplier_id int(10) NOT NULL,supplier_name
varchar2(50) NOT NULL);
```

Qno3:-

The third column is called address which is a varchar2 data type but can contain null values.

Answer:-

```
CREATE TABLE suppliers(supplier_id int(10) NOT NULL,supplier_name
varchar(50) NOT NULL,address varchar2 NOT NULL);
```

Qno4:-

Define the supplier_id as the primary key

Answer:-

```
CREATE TABLE suppliers(supplier_id NOT NULL
AUTO_INCREMENT,supplier_id int(10) NOT NULL,supplier_name varchar(50)
NOT NULL,address varchar2 NOT NULL,PRIMARY KEY(supplier_id);
```

Qno5:-

Create a second table named as Item with columns:

Answer:-

```
CREATE TABLE Item();
```

Qno6:-

The first column itemname any length you want

Answer:-

```
CREATE TABLE Item(itemname varchar(255);
```

Qno7:-

The second column supplierId as foreignkey in item table

Answer:-

```
CREATE TABLE item (
   itemname varchar(255),
   FOREIGN KEY (supplier_id) REFERENCES Persons(supplier_id)
);
```

Qno8:-

The third column should be itemprice In INT

Answer:-

```
CREATE TABLE item (
   itemname varchar(255),

   Itemprice int NOT NULL,
   FOREIGN KEY (supplier_id) REFERENCES Persons(supplier_id)
);
```

LAB20

Qno1:-

Write a SQL function to convert temperature from Fahrenheit to Celsius scale

Answer:-

```
DECLARE

   temp1     NUMBER := &input_a_temp;

   t_scale   CHAR := '&input_temp_scale';

   new_temp  NUMBER;
```

```
    new_scale CHAR;
  BEGIN
    IF t_scale != 'C'
      AND
      t_scale != 'F' THEN
      dbms_output.Put_line ('The scale you input is not a valid scale');
      new_temp := 0;
      new_scale := 'C';
    ELSE
      IF t_scale = 'C' THEN
        new_temp := ( ( 9 * temp1 ) / 5 ) + 32;
        new_scale := 'F';
      ELSE
        new_temp := ( ( temp1 - 32 ) * 5 ) / 9;
        new_scale := 'C';
      END IF;
    END IF;
    dbms_output.Put_line ('The new temperature in scale '
    ||new_scale
    ||' is: '
    ||new_temp);
  END;
/
```

Qno2:-

Write a SQL function to find GP and letter grade from percentage marks as per CIIT grading system.

Answer:-

Hamza Aslam

SP17-bcs-050

Lab 6

Qno1:-

Answer:-

```
create database HomeSweetHome;

USE HomeSweetHome;

create table Branch

(

branchNumber varchar(25) NOT NULL Primary Key,

streetNumber varchar(110) NOT NULL,

cityName varchar(60) NOT NULL,

postcodeNumber int NOT NULL

);

create table Staff

(

staffNumber int NOT NULL PRIMARY KEY,

firstName varchar(50) NOT NULL,

lastName varchar(50) NOT NULL,
position varchar(50) NOT NULL,
sex char NOT NULL, DOB DateTime NOT NULL,
salary DECIMAL NOT NULL,
branchNumber varchar(25) NOT NULL References Branch(branchNumber)
);
create table Client
(
clientNumber varchar(20) NOT NULL PRIMARY KEY,
firstName varchar(50) NOT NULL,
lastName varchar(50) NOT NULL,
phoneNo int NOT NULL,
prefType varchar(50) NOT NULL,
maxRent DECIMAL NOT NULL
);
create table PrivateOwner
(
ownerNumber varchar(20) NOT NULL PRIMARY KEY,
firstName varchar(50) NOT NULL,
lastName varchar(50) NOT NULL,
address varchar(50) NOT NULL,
phoneNo int NOT NULL
);
create table PropertyForRent
(
propertyNumber varchar(20) NOT NULL PRIMARY KEY,
streetNumber varchar(100) NOT NULL,
cityName varchar(50) NOT NULL,
postcodeNumber int NOT NULL,
type varchar(10) NOT NULL,
rooms int NOT NULL,
rent DECIMAL NOT NULL,
ownerNumber varchar(20) References PrivateOwner(ownerNumber),
staffNumber varchar(20) NOT NULL References Staff(staffNumber),
branchNumber varchar(20) NOT NULL References Branch(branchNumber)
);
create table Viewing
(
clientNumber varchar(20) NOT NULL References Client(clientNumber),
propertyNumber varchar(20) NOT NULL References
PropertyForRent(propertyNumber),
viewDate DateTime NOT NULL,
comment varchar(200) NOT NULL
);
create table Registration
(
clientNumber varchar(20) NOT NULL References Client(clientNumber),
branchNumber varchar(20) NOT NULL References Branch(branchNumber),
staffNumber varchar(20) NOT NULL References Staff(staffNumber),
dateJoined DateTime NOT NULL
);
```

```
INSERT into Branch
(
branchNumber,
streetNumber,
cityNumber,
postcodeNumber
)
VALUES
(N'B001',N'H#7 I-10/2', N'ISB', N'52000'),
(N'B002',N'H#78 Supply', N'ABT', N'53000'),
(N'B003',N'H#79 I-10/2', N'ISB', N'52000'),
(N'B004',N'H#78 Mandian', N'ABT', N'53000');

insert into Staff
(
staffNumber,
firstName,
lastName,
position,
sex,
DOB,
salary,
branchNumber
)
VALUES
```

```
(
N'SA9', N'Mary', N'Howe', N'Assistant', N'F',CAST(0x0000641000000000 AS
DateTime),
CAST(9000 AS Decimal(18, 0)),N'B002'
),
(N'SG14', N'David', N'Ford', N'Supervisor', N'M',
CAST(0x0000531200000000 AS DateTime), CAST(18000 AS Decimal(18,0)),
N'B003'),
(N'SG37', N'Ann', N'Beech', N'Assistant', N'F',
CAST(0x000056D400000000 AS DateTime), CAST(12000 AS Decimal(18,0)),
N'B003'),
(N'SG5',N'Susan', N'Brand', N'Manager', N'F',
CAST(0x0000C85800000000 AS DateTime), CAST(24000 AS Decimal(18,0)),
N'B003'),
(N'SL21', N'John', N'White', N'Manager', N'M',
CAST(0x0000CFF200000000 AS DateTime), CAST(30000 AS Decimal(18,0)),
N'B004'),
(N'SL41', N'Julie', N'Lee', N'Assistant', N'F',
CAST(0x00005D6000000000 AS DateTime), CAST(9000 AS Decimal(18, 0)),
N'B002');

insert into Client
(
clientNumber,
firstName,
lastName,
PhoneNumber,
prefType,
maxRent
)
values
(
'B1001','Mahad','Ali','030078601','yes',1000.0
),
(
'B1002','Sharjeel','Khan','030054621','yes',2000.0
),
(
'B1003','Hanzala','Shahid','030456601','no',1500.0
),
(
'B1004','Hamza','Aslam','0306446641','yes',8800.0
),
(
'B1005','Hamza','Ahmad','0354654401','noo',800.0
),
(
'B1006','Usama','Fareed','030074541','yes',4000.0
),
(
'B1007','Abdullah','Niazi','03359613366','yes',2500.0
```

```
);
insert into PrivateOwner
(
ownerNumber,
firstName,
lastName,
[address],
PhoneNumber
)
values
(
'B1','Azid','Ali','F18-4A','0354654264'
),
(
'B2','Mahad','Ali','F17-4A','0345154264'
),
(
'B3','Sharjeel','Khan','F14-7A','0352354264'
),
(
'B4','Hanzala','Shahid','F88-4A','0359354264'
),
(
'B5','Hamza','Aslam','F11-3A','0351694264'
),
(
'B6','Hamza','Ahmad','F19-5A','0354654264'
)
(
'B7','Abdullah','Niazi','F18-5A','03359613366'
);

insert into PropertyForRent
(
propertyNumber,streetNumber,cityName,postcodeNumber,[type],rooms,rent,ownerNumnber,staffNumber,branchNumber
)
values
(
'BF2','H2-h2','ABT','22010','large',8,'30000','B2','SG14','B002'
),
(
'BF3','H3-h3','ISB','62010','medium',6,'20000','B3','SG37','B003'
),
(
'BF4','H4-h4','ISB','62010','small',4,'10000','B4','SG5','B004'
);

insert into Viewing
(
clientNumber,
propertyNumber,
viewDate,
comment
)
values
(
'B1002','BF2','2020-2-10','No, i dont wanna say anything'
),
(
'B1003','BF3','2020-3-10','No, i dont wanna say anything'
),
(
'B1004','BF4','2020-4-10','No, i dont wanna say anything'
);

insert into Viewing
(
clientNumber,
propertyNumber,
viewDate,
comment
)
values
(
'B1002','BF2','2020-2-10','No, i dont wanna say anything'
),
(
'B1003','BF3','2020-3-10','No, i dont wanna say anything'
),
(
'B1004','BF4','2020-4-10','No, i dont wanna say anything'
);
```

Qno:-2

Answer:-

```
select * from Branch update Branch set city='ABT' where city='ISB';
```

Lab 7

Qno1:-

Answer

```
select distinct(postcodeNumber) from Branch
```

Qno2:-

Answer

```
select distinct(firstName) from Staff
```

Qno3:-

Answer

```
select staffNumber as [Cadre No], firstName as [Baptism Name], lastName as [Sur name],

position as [Locale], sex as [Gender],DOB as [Birtday] ,salary as Income,
branchNumber as [Section No] from Staff
```

Qno4:-

Answer

```
select clientNumber as [Buyer No], firstName as [Baptism Name], lastName as
[Sur name],

PhoneNumber as [Fax Number],prefType as [Proclivity Type],maxRent as
[Supreme Cost] from Client;
```

Qno5:-

Answer

```
select * from Staff where salary>10000
```

Qno6:-

Answer

```
select * from Staff where position='Manager' or position='Supervisor'
```

Lab 8

Qno1:-

Answer

```
select staffNumber,firstName,lastName,salary from staff order by salary desc
```

Qno2:-

Answer

```
select propertyNumber,type,rooms,rent from PropertyForRent

order by type

select propertyNumber,type,rooms,rent
```

```
from PropertyForRent
order by type,rent desc
```

Qno3:-

Answer

```
select count(*) as myCount
from PropertyForRent
where rent<=500
```

Qno4:-

Answer

```
select count(Distinct propertyNumber) As myCount from Viewing
WHERE viewDate BETWEEN '1-May-04' AND '31-May-04';
```

Qno5:-

Answer

```
select count(staffNumber) as myCount,sum(salary) as mySalary from staff
where
position='Manager'
```

Qno6:-

Answer

```
select MIN(salary) as myMin,
MAX(salary) as myMax,
AVG(salary) as myAVG from Staff
```

Qno7:-

Answer

```
SELECT staffNumber, firstName, lastName, position, salary
FROM Staff
WHERE (SELECT AVG(salary) FROM Staff) < salary;
```

Qno8:-

Answer

```
select *from Staff where salary> any(select salary from Staff where branchNumber='B003')
```

Qno9:-

Answer

```
select *from Staff where salary> all(select salary from Staff where branchNumber='B003')
```

LAB 9

Qno1:

list all tables in the employees database

Answer:

```
 USE EMPLOYEE;
show TABLES;
```

Qno2:

Write a query to find the names (first_name, last_name) and the salaries of the employees who have a higher salary than the employee whose last_name=’Bull’.

Answer:-

```
Select FIRST_NAME, LAST_NAME, SALARY

FROM employees

WHERE SALARY>(SELECT salary FROM employees WHERE  last_name='Bull');
```

Qno3:

Answer:-

```
Select first_name , last_name

FROM employees

WHERE department_id

IN(SELECT department_id FROM departments WHERE department_name='IT');
```

LAB 10

Qno1:-

Write a query to find the names (first_name, last_name) of the employees who have a manager and work for a department based in the United States.

Answer:-

```
SELECT first_name, last_name FROM employees

WHERE manager_id in(select employee_id FROM employees WHERE department_id

IN(SELECT department_id FROM departments WHERE location_id IN(select location_id from locations

Where country_id='US')));
```

Qno2:-

Write a query to find the names (first_name, last_name) of the employees who are managers.

Answer:-

```
SELECT first_name, last_name

FROM employees

WHERE (employee_id IN(SELECT manager_id FROM employees));
```

Qno3:-

Write a query to find the names (first_name, last_name), the salary of the employees whose salary is greater than the average salary

Answer:-

```
SELECT first_name, last_name, salary FROM employees

WHERE salary > (SELECT AVG(salary) FROM employees);
```

Qno4:-

Write a query to find the names (first_name, last_name), the salary of the employees whose salary is equal to the minimum salary for their job grade.

Answer:-

```
SELECT first_name,last_name,salary FROM employees WHERE employees.salary=(SELECT min_salary
FROM jobs WHERE employees.job_id=jobs.job_id);
```

Qno5:-

Write a query to find the names (first_name, last_name), the salary of the employees who earn more than the average salary and who works in any of the IT departments.

Answer:-

```
SELECT first_name,last_name,salary

FROM employees WHERE department_id IN (SELECT department_id FROM departments WHERE
department_name LIKE ‘IT%’)AND salary>(SELECT avg(salary) From employees);
```

Qno6:-

Write a query to find the names (first_name, last_name), the salary of the employees who earn more than Mr. Bell

Answer:-

```
SELECT * FROM employees WHERE salary > ALL(SELECT AVG(salary) FROM employees GROUP BY

Department_id);
```

Qno7:-

Write a query to find the names (first_name, last_name), the salary of the employees who earn the same salary as the minimum salary for all departments

Answer:-

```
SELECT * FROM employees

WHERE salary=(SELECT MIN(salary) FROM employees);
```

Qno8:-

Write a query to find the names (first_name, last_name), the salary of the employees whose salary greater than the average salary of all departments

Answer:-

```
SELECT first_name,last_name from employees whose(SELECT AVG(salary) from departments)
```

Qno9:-

Write a query to find the names (first_name, last_name) and salary of the employees who earn a salary that is higher than the salary of all the Shipping Clerk (JOB_ID = ’SH_CLERK’). Sort the results of the salary of the lowest to highest

Answer:-

```
SELECT first_name,last_name, job_id, salary

FROM employees

WHERE salary >

ALL (SELECT salary FROM employees WHERE job_id = 'SH_CLERK') ORDER BY salary;
```

Qno10:-

.Write a query to find the names (first_name, last_name) of the employees who are not supervisors.

Answer:-

```
SELECT b.first_name,b.last_name
```

```
FROM employees b
WHERE NOT EXISTS (SELECT 'X' FROM employees a WHERE a.manager_id = b.employee_id);
```

Qno11:-

Write a query to display the employee ID, first name, last names, and department names of all employees.

Answer:-

```
SELECT employee_id, first_name, last_name,
(SELECT department_name FROM departments d
 WHERE e.department_id = d.department_id) department
 FROM employees e ORDER BY department;
```

Qno12:-

Write a query to display the employee ID, first name, last names, salary of all employees whose salary is above average for their departments

Answer:-

```
SELECT employee_id, first_name
FROM employees AS A
WHERE salary >
(SELECT AVG(salary) FROM employees WHERE department_id = A.department_id);
```

Qno13:-

Write a query to fetch even numbered records from employees table

Answer:-

```
SET @i = 0;
SELECT i, employee_id
FROM (SELECT @i := @i + 1 AS i, employee_id FROM employees)
a WHERE MOD(a.i, 2) = 0;
```

Qno14:-

Write a query to find the 5th maximum salary in the employees table.

Answer:-

```
SELECT DISTINCT salary

FROM employees e1

WHERE 5 = (SELECT COUNT(DISTINCT salary)

FROM employees  e2

WHERE e2.salary >= e1.salary);
```

Qno15:-

Write a query to find the 4th minimum salary in the employees table

Answer:-

```
SELECT DISTINCT salary

FROM employees e1

WHERE 4 = (SELECT COUNT(DISTINCT salary)

FROM employees  e2

WHERE e2.salary <= e1.salary);
```

Qno16:-

Write a query to select last 10 records from a table.

Answer:-

```
SELECT * FROM (

SELECT * FROM employees ORDER BY employee_id DESC LIMIT 10) sub

ORDER BY employee_id ASC;
```

Qno17:-

Write a query to list department number, name for all the departments in which there are no employees in the department

Answer:-

```
SELECT * FROM departments
WHERE department_id
NOT IN (select department_id FROM employees);
```

Qno18:-

Write a query to get 3 maximum salaries.

Answer:-

```
SELECT DISTINCT salary
FROM employees a
WHERE 3 >= (SELECT COUNT(DISTINCT salary)
FROM employees b
WHERE b.salary >= a.salary)
ORDER BY a.salary DESC;
```

Qno19:-

Write a query to get 3 minimum salaries

Answer:-

```
SELECT DISTINCT salary
FROM employees a
WHERE  3 >= (SELECT COUNT(DISTINCT salary)
FROM employees b
WHERE b.salary <= a.salary)
ORDER BY a.salary DESC;
```

Qno20:-

Write a query to get nth max salaries of employees.Further practice with nested queries

Answer:-

```
SELECT *
FROM employees emp1
WHERE (1) = (
SELECT COUNT(DISTINCT(emp2.salary))
FROM employees emp2
WHERE emp2.salary > emp1.salary);
```

LAB11

Qno1:-

Create a table tow columns for name and family_name respectively. Insert the names your three friends in lower case case caracters. Write a query to create columns aliased fullname by using the INITCAT() and CONCAT() functions.

Answer:-

```
SELECT customer_id, CONCAT(first_name,second_name,last_name)AS All_names from customer
```

LAB12

Qno1:-

Print countrycode and sum of percentage from countrylangauge, apply groupby on countrycode.

Answer:-

SELECT countrycode ,SUM(PER(countrylanguage) from country language Group By(countrycode)

Qno2:-

Find sum of any integer column from country table.

Answer:-

Select SUM(population) from country table;

Qno3:-

Count number of records in country table

Answer:-

SELECT COUNT(records) from countrytable;

Qno4:-

Count Distinct (languages) from countrylanguage

Answer:-

SELECT DISTINCT(languages) from countrylanguage;

LAB13

Qno1:-

Select customers name, number , phone from customers table, select checknumber from payments table. Display it for all customers. [either they have made payment or they haven't include all customers].

Answer:-

SELECT customer,phone from customertable;

SELECT checknumber from payment;

LAB14

Qno1:-

Update customer with any a particular order number (you can select any order number).

```
UPDATE customer
SET order_no = 5;
```

Qno2:-

Applying union print data of orders and order details table.

Answer:-

```
SELECT data.customerdata from customers
UNION ALL
SELECT order.orderdata FROM orders;
```

LAB15

Qno1:-

Apply update on any country name.

Answer:-

UPDATE country SET column1=”England” WHERE column=”Islamabad”;

Qno2:-

Delete Islamabad city by applying delete query on city table with it’s ID

Answer:-

```
DELETE FROM city
WHERE country.id = ANY (SELECT id FROM city WHERE id = 2);
```

Qno3:-

Try to update values for null column COMM column

Answer:-

```
UPDATE[city]

SET [COMM]=0

WHERE [COMM] is null;
```

Qno4:-

Try to update it for a specific employee whose salary is less than 1000

Answer:-

```
UPDATE employee   WHERE employee<1000;
```

Qno5:-

Answer:-

```
INSERT INTO city
VALUES (“Mian Channu”, “Chakwal”, “Abbottabad”);
```

LAB16

Qno1:-

The first column is called supplier_id which is created as a number data type (maximum 10 digits in length) and cannot contain null values.

Answer:-

```
CREATE TABLE suppliers(supplier_id int(10) NOT NULL);
```

Qno2:-

The second column is called supplier_name which is a varchar2 datatype (50 maximum characters in length) and also can not contain null values

Answer:-

```
CREATE TABLE suppliers(supplier_id int(10) NOT NULL,supplier_name varchar2(50) NOT NULL);
```

Qno3:-

The third column is called address which is a varchar2 data type but can contain null values.

Answer:-

CREATE TABLE suppliers(supplier_id int(10) NOT NULL,supplier_name varchar(50) NOT NULL,address varchar2 NOT NULL);

Qno4:-

Define the supplier_id as the primary key

Answer:-

CREATE TABLE suppliers(supplier_id NOT NULL AUTO_INCREMENT,supplier_id int(10) NOT NULL,supplier_name varchar(50) NOT NULL,address varchar2 NOT NULL,PRIMARY KEY(supplier_id);

Qno5:-

Create a second table named as Item with columns:

Answer:-

CREATE TABLE Item();

Qno6:-

The first column itemname any length you want

Answer:-

CREATE TABLE Item(itemname varchar(255);

Qno7:-

The second column supplierId as foreignkey in item table

Answer:-

```
CREATE TABLE item (
    itemname varchar(255),
    FOREIGN KEY (supplier_id) REFERENCES Persons(supplier_id)
);
```

Qno8:-

The third column should be itemprice In INT

Answer:-

```
CREATE TABLE item (
    itemname varchar(255),

    Itemprice int NOT NULL,
    FOREIGN KEY (supplier_id) REFERENCES Persons(supplier_id)
);
```

LAB20

Qno1:-

Write a SQL function to convert temperature from Fahrenheit to Celsius scale

Answer:-

```
DECLARE
  temp1    NUMBER := &input_a_temp;
  t_scale   CHAR := '&input_temp_scale';
  new_temp  NUMBER;
  new_scale CHAR;
 BEGIN
  IF t_scale != 'C'
   AND
   t_scale != 'F' THEN
   dbms_output.Put_line ('The scale you input is not a valid scale');
   new_temp := 0;
   new_scale := 'C';
  ELSE
   IF t_scale = 'C' THEN
    new_temp := ( ( 9 * temp1 ) / 5 ) + 32;
    new_scale := 'F';

   ELSE

    new_temp := ( ( temp1 - 32 ) * 5 ) / 9;

    new_scale := 'C';

   END IF;

  END IF;

  dbms_output.Put_line ('The new temperature in scale '

  ||new_scale

  ||' is: '

  ||new_temp);

 END;

/
```

Hamza Ahmad

FA18-BCS-084

Qno1:-

Answer:-

```
create database DreamHome;
USE DreamHome;
create table Branch
(
branchNo varchar(20) NOT NULL Primary Key,
street varchar(100) NOT NULL,
city varchar(50) NOT NULL,
postcode varchar(20) NOT NULL
);
create table Staff
(
staffNo varchar(20) NOT NULL PRIMARY KEY,
fName varchar(50) NOT NULL,
lName varchar(50) NOT NULL,
position varchar(50) NOT NULL,
sex varchar(1) NOT NULL, DOB DateTime NOT NULL,
salary DECIMAL NOT NULL,
branchNo varchar(20) NOT NULL References Branch(branchNo)
);
create table Client
(
clientNo varchar(20) NOT NULL PRIMARY KEY,
fName varchar(50) NOT NULL,
lName varchar(50) NOT NULL,
telNo varchar(20) NOT NULL,
prefType varchar(50) NOT NULL,
maxRent DECIMAL NOT NULL
);
create table PrivateOwner
(
ownerNo varchar(20) NOT NULL PRIMARY KEY,
fName varchar(50) NOT NULL,
lName varchar(50) NOT NULL,
address varchar(50) NOT NULL,
telNo varchar(20) NOT NULL
);
create table PropertyForRent
(
propertyNo varchar(20) NOT NULL PRIMARY KEY,
street varchar(100) NOT NULL,
city varchar(50) NOT NULL,
postcode varchar(20) NOT NULL,
type varchar(10) NOT NULL,
rooms int NOT NULL,
rent DECIMAL NOT NULL,
ownerNo varchar(20) References PrivateOwner(ownerNo),
staffNo varchar(20) NOT NULL References Staff(staffNo),
branchNo varchar(20) NOT NULL References Branch(branchNo)
);
create table Viewing
(
clientNo varchar(20) NOT NULL References Client(clientNo),
propertyNo varchar(20) NOT NULL References PropertyForRent(propertyNo),
viewDate DateTime NOT NULL,
comment varchar(200) NOT NULL
);
create table Registration
(
clientNo varchar(20) NOT NULL References Client(clientNo),
branchNo varchar(20) NOT NULL References Branch(branchNo),
staffNo varchar(20) NOT NULL References Staff(staffNo),
dateJoined DateTime NOT NULL
);


INSERT into Branch
(
branchNo,
street,
city,
postcode
)
VALUES
(N'B001',N'H#7 I-10/2', N'ISB', N'52000'),
(N'B002',N'H#78 Supply', N'ABT', N'53000'),
(N'B003',N'H#79 I-10/2', N'ISB', N'52000'),
(N'B004',N'H#78 Mandian', N'ABT', N'53000');

insert into Staff
(
staffNo,
fName,
lName,
position,
sex,
DOB,
salary,
branchNo
)
VALUES
(
N'SA9', N'Mary', N'Howe', N'Assistant', N'F',CAST(0x0000641000000000 AS DateTime),
CAST(9000 AS Decimal(18, 0)),N'B002'
),
(N'SG14', N'David', N'Ford', N'Supervisor', N'M',
CAST(0x0000531200000000 AS DateTime), CAST(18000 AS Decimal(18,0)),
N'B003'),
(N'SG37', N'Ann', N'Beech', N'Assistant', N'F',
CAST(0x000056D400000000 AS DateTime), CAST(12000 AS Decimal(18,0)),
N'B003'),
(N'SG5',N'Susan', N'Brand', N'Manager', N'F',
CAST(0x0000C85800000000 AS DateTime), CAST(24000 AS Decimal(18,0)),
N'B003'),
(N'SL21', N'John', N'White', N'Manager', N'M',
CAST(0x0000CFF200000000 AS DateTime), CAST(30000 AS Decimal(18,0)),
N'B004'),
(N'SL41', N'Julie', N'Lee', N'Assistant', N'F',
CAST(0x00005D6000000000 AS DateTime), CAST(9000 AS Decimal(18, 0)),
N'B002');


insert into Client
(
clientNo,
fName,
lName,
telNo,
prefType,
maxRent
)
values
(
'B1001','Mahad','Ali','030078601','yes',1000.0
),
(
'B1002','Sharjeel','Khan','030054621','yes',2000.0
),
(
'B1003','Hanzala','Shahid','030456601','no',1500.0
),
(
'B1004','Hamza','Aslam','0306446641','yes',8800.0
),
(
'B1005','Hamza','Ahmad','0354654401','noo',800.0
),
(
'B1006','Usama','Fareed','030074541','yes',4000.0
);

insert into PrivateOwner
(
ownerNo,
fName,
lName,
[address],
telNo
)
values
(
'B1','Azid','Ali','F18-4A','0354654264'
),
(
'B2','Mahad','Ali','F17-4A','0345154264'
),
(
'B3','Sharjeel','Khan','F14-7A','0352354264'
),
(
'B4','Hanzala','Shahid','F88-4A','0359354264'
),
(
'B5','Hamza','Aslam','F11-3A','0351694264'
),
(
'B6','Hamza','Ahmad','F19-5A','0354654264'
);


insert into PropertyForRent
(
propertyNo,street,city,postcode,[type],rooms,rent,ownerNo,staffNo,branchNo
)
values
(
'BF2','H2-h2','ABT','22010','large',8,'30000','B2','SG14','B002'
),
(
'BF3','H3-h3','ISB','62010','medium',6,'20000','B3','SG37','B003'
),
(
'BF4','H4-h4','ISB','62010','small',4,'10000','B4','SG5','B004'
);


insert into Viewing
(
clientNo,
propertyNo,
viewDate,
comment
)
values
(
'B1002','BF2','2020-2-10','No, i dont wanna say anything'
),
(
'B1003','BF3','2020-3-10','No, i dont wanna say anything'
),
(
'B1004','BF4','2020-4-10','No, i dont wanna say anything'
);


insert into Viewing
(
clientNo,
propertyNo,
viewDate,
comment
)
values
(
'B1002','BF2','2020-2-10','No, i dont wanna say anything'
),
(
'B1003','BF3','2020-3-10','No, i dont wanna say anything'
),
(
'B1004','BF4','2020-4-10','No, i dont wanna say anything'
);
```

End of qno1

Qno:-2

Answer:-

```
select * from Branch update Branch set city='ABT' where city='ISB';
```

Lab 7

Qno1:-

Answer

```
select distinct(postcode) from Branch
```

Qno2:-

Answer

```
select distinct(fName) from Staff
```

Qno3:-

Answer

```
select staffNo as [Cadre No], fName as [Baptism Name], lName as [Sur name],
position as [Locale], sex as [Gender],DOB as [Birtday] ,salary as Income,
branchNo as [Section No] from Staff
```

Qno4:-

Answer

```
select clientNo as [Buyer No], fName as [Baptism Name], lName as [Sur name],
telNo as [Fax Number],prefType as [Proclivity Type],maxRent as [Supreme Cost] from
Client;
```

Qno5:-

Answer

```
select * from Staff where salary>10000
```

Qno6:-

Answer

```
select * from Staff where position='Manager' or position='Supervisor'
```

Lab 8

Qno1:-

Answer

```
select staffNo,fName,lName,salary from staff order by salary desc
```

Qno2:-

Answer

```
select propertyNo,type,rooms,rent from PropertyForRent
order by type
select propertyNo,type,rooms,rent
from PropertyForRent
order by type,rent desc
```

Qno3:-

Answer

```
select count(*) as myCount
from PropertyForRent
where rent<=500
```

Qno4:-

Answer

```
select count(Distinct propertyNo) As myCount from Viewing
WHERE viewDate BETWEEN '1-May-04' AND '31-May-04';
```

Qno5:-

Answer

```
select count(staffNo) as myCount,sum(salary) as mySalary from staff
where
position='Manager'
```

Qno6:-

Answer

```
select MIN(salary) as myMin,
MAX(salary) as myMax,
AVG(salary) as myAVG from Staff
```

Qno7:-

Answer

```
SELECT staffNo, fName, lName, position, salary
FROM Staff
WHERE (SELECT AVG(salary) FROM Staff) < salary;
```

Qno8:-

Answer

```
select *from Staff where salary> any(select salary from Staff where branchNo='B003')
```

Qno9:-

Answer

```
select *from Staff where salary> all(select salary from Staff where branchNo='B003')
```

LAB 9

Qno1:

list all tables in the employees database

Answer:

USE EMPLOYEE;

show TABLES;

Qno2:

Write a query to find the names (first_name, last_name) and the salaries of the employees who have a higher salary than the employee whose last_name='Bull'.

Answer:-

```
Select FIRST_NAME, LAST_NAME, SALARY

FROM employees

WHERE SALARY>(SELECT salary FROM employees WHERE  last_name=’Bull’);
```

Qno3:

Answer:-

```
Select first_name , last_name

FROM employees

WHERE department_id

IN(SELECT department_id FROM departments WHERE department_name=’IT’);
```

LAB 10

Qno1:-

Write a query to find the names (first_name, last_name) of the employees who have a manager and work for a department based in the United States.

Answer:-

```
SELECT first_name, last_name FROM employees

WHERE manager_id in(select employee_id FROM employees WHERE department_id

IN(SELECT department_id FROM departments WHERE location_id IN(select location_id from locations

Where country_id=’US’)));
```

Qno2:-

Write a query to find the names (first_name, last_name) of the employees who are managers.

Answer:-

```
SELECT first_name, last_name

FROM employees

WHERE (employee_id IN(SELECT manager_id FROM employees));
```

Qno3:-

Write a query to find the names (first_name, last_name), the salary of the employees whose salary is greater than the average salary

Answer:-

```
SELECT first_name, last_name, salary FROM employees

WHERE salary > (SELECT AVG(salary) FROM employees);
```

Qno4:-

Write a query to find the names (first_name, last_name), the salary of the employees whose salary is equal to the minimum salary for their job grade.

Answer:-

```
SELECT first_name,last_name,salary FROM employees WHERE employees.salary=(SELECT min_salary
FROM jobs WHERE employees.job_id=jobs.job_id);
```

Qno5:-

Write a query to find the names (first_name, last_name), the salary of the employees who earn more than the average salary and who works in any of the IT departments.

Answer:-

```
SELECT first_name,last_name,salary

FROM employees WHERE department_id IN (SELECT department_id FROM departments WHERE
department_name LIKE ‘IT%’)AND salary>(SELECT avg(salary) From employees);
```

Qno6:-

Write a query to find the names (first_name, last_name), the salary of the employees who earn more than Mr. Bell

Answer:-

```
SELECT * FROM employees WHERE salary > ALL(SELECT AVG(salary) FROM employees GROUP BY

Department_id);
```

Qno7:-

Write a query to find the names (first_name, last_name), the salary of the employees who earn the same salary as the minimum salary for all departments

Answer:-

```
SELECT * FROM employees

WHERE salary=(SELECT MIN(salary) FROM employees);
```

Qno8:-

Write a query to find the names (first_name, last_name), the salary of the employees whose salary greater than the average salary of all departments

Answer:-

```
SELECT first_name,last_name from employees whose(SELECT AVG(salary) from departments)
```

Qno9:-

Write a query to find the names (first_name, last_name) and salary of the employees who earn a salary that is higher than the salary of all the Shipping Clerk (JOB_ID = ’SH_CLERK’). Sort the results of the salary of the lowest to highest

Answer:-

```
SELECT first_name,last_name, job_id, salary

FROM employees

WHERE salary >

ALL (SELECT salary FROM employees WHERE job_id = 'SH_CLERK') ORDER BY salary;
```

Qno10:-

.Write a query to find the names (first_name, last_name) of the employees who are not supervisors.

Answer:-

```
SELECT b.first_name,b.last_name
```

```
FROM employees b
WHERE NOT EXISTS (SELECT 'X' FROM employees a WHERE a.manager_id = b.employee_id);
```

Qno11:-

Write a query to display the employee ID, first name, last names, and department names of all employees.

Answer:-

```
SELECT employee_id, first_name, last_name,
(SELECT department_name FROM departments d
 WHERE e.department_id = d.department_id) department
 FROM employees e ORDER BY department;
```

Qno12:-

Write a query to display the employee ID, first name, last names, salary of all employees whose salary is above average for their departments

Answer:-

```
SELECT employee_id, first_name
FROM employees AS A
WHERE salary >
(SELECT AVG(salary) FROM employees WHERE department_id = A.department_id);
```

Qno13:-

Write a query to fetch even numbered records from employees table

Answer:-

```
SET @i = 0;
SELECT i, employee_id
FROM (SELECT @i := @i + 1 AS i, employee_id FROM employees)
a WHERE MOD(a.i, 2) = 0;
```

Qno14:-

Write a query to find the 5th maximum salary in the employees table.

Answer:-

```
SELECT DISTINCT salary
FROM employees e1
WHERE 5 = (SELECT COUNT(DISTINCT salary)
FROM employees  e2
WHERE e2.salary >= e1.salary);
```

Qno15:-

Write a query to find the 4th minimum salary in the employees table

Answer:-

```
SELECT DISTINCT salary
FROM employees e1
WHERE 4 = (SELECT COUNT(DISTINCT salary)
FROM employees  e2
WHERE e2.salary <= e1.salary);
```

Qno16:-

Write a query to select last 10 records from a table.

Answer:-

```
SELECT * FROM (
SELECT * FROM employees ORDER BY employee_id DESC LIMIT 10) sub
ORDER BY employee_id ASC;
```

Qno17:-

Write a query to list department number, name for all the departments in which there are no employees in the department

Answer:-

```
SELECT * FROM departments

WHERE department_id

NOT IN (select department_id FROM employees);
```

Qno18:-

Write a query to get 3 maximum salaries.

Answer:-

```
SELECT DISTINCT salary

FROM employees a

WHERE 3 >= (SELECT COUNT(DISTINCT salary)

FROM employees b

WHERE b.salary >= a.salary)

ORDER BY a.salary DESC;
```

Qno19:-

Write a query to get 3 minimum salaries

Answer:-

```
SELECT DISTINCT salary

FROM employees a

WHERE  3 >= (SELECT COUNT(DISTINCT salary)

FROM employees b

WHERE b.salary <= a.salary)

ORDER BY a.salary DESC;
```

Qno20:-

Write a query to get nth max salaries of employees.Further practice with nested queries

Answer:-

```
SELECT *
FROM employees emp1
WHERE (1) = (
SELECT COUNT(DISTINCT(emp2.salary))
FROM employees emp2
WHERE emp2.salary > emp1.salary);
```

LAB11

Qno1:-

Create a table tow columns for name and family_name respectively. Insert the names your three friends in lower case case caracters. Write a query to create columns aliased fullname by using the INITCAT() and CONCAT() functions.

Answer:-

```
SELECT customer_id, CONCAT(first_name,second_name,last_name)AS All_names from customer
```

LAB12

Qno1:-

Print countrycode and sum of percentage from countrylangauge, apply groupby on countrycode.

Answer:-

SELECT countrycode ,SUM(PER(countrylanguage) from country language Group By(countrycode)

Qno2:-

Find sum of any integer column from country table.

Answer:-

Select SUM(population) from country table;

Qno3:-

Count number of records in country table

Answer:-

SELECT COUNT(records) from countrytable;

Qno4:-

Count Distinct (languages) from countrylanguage

Answer:-

SELECT DISTINCT(languages) from countrylanguage;

LAB13

Qno1:-

Select customers name, number , phone from customers table, select checknumber from payments table. Display it for all customers. [either they have made payment or they haven't include all customers].

Answer:-

SELECT customer,phone from customertable;

SELECT checknumber from payment;

LAB14

Qno1:-

Update customer with any a particular order number (you can select any order number).

```
UPDATE customer
SET order_no = 5;
```

Qno2:-

Applying union print data of orders and order details table.

Answer:-

```
SELECT data.customerdata from customers
UNION ALL
SELECT order.orderdata FROM orders;
```

LAB15

Qno1:-

Apply update on any country name.

Answer:-

```
UPDATE country SET column1=”England”  WHERE column=”Islamabad”;
```

Qno2:-

Delete Islamabad city by applying delete query on city table with it’s ID

Answer:-

```
DELETE FROM city
WHERE country.id = ANY (SELECT id FROM city WHERE id = 2);
```

Qno3:-

Try to update values for null column COMM column

Answer:-

```
UPDATE[city]
SET [COMM]=0
WHERE [COMM] is null;
```

Qno4:-

Try to update it for a specific employee whose salary is less than 1000

Answer:-

```
UPDATE employee   WHERE employee<1000;
```

Qno5:-

Answer:-

```
INSERT INTO city
VALUES (“Lahore”, “Bahria”, “Karachi”);
```

LAB16

Qno1:-

The first column is called supplier_id which is created as a number data type (maximum 10 digits in length) and cannot contain null values.

Answer:-

```
CREATE TABLE suppliers(supplier_id int(10) NOT NULL);
```

Qno2:-

The second column is called supplier_name which is a varchar2 datatype (50 maximum characters in length) and also can not contain null values

Answer:-

```
CREATE TABLE suppliers(supplier_id int(10) NOT NULL,supplier_name varchar2(50) NOT NULL);
```

Qno3:-

The third column is called address which is a varchar2 data type but can contain null values.

Answer:-

```
CREATE TABLE suppliers(supplier_id int(10) NOT NULL,supplier_name varchar(50) NOT
NULL,address varchar2 NOT NULL);
```

Qno4:-

Define the supplier_id as the primary key

Answer:-

```
CREATE TABLE suppliers(supplier_id NOT NULL AUTO_INCREMENT,supplier_id int(10) NOT
NULL,supplier_name varchar(50) NOT NULL,address varchar2 NOT NULL,PRIMARY
KEY(supplier_id);
```

Qno5:-

Create a second table named as Item with columns:

Answer:-

```
CREATE TABLE Item();
```

Qno6:-

The first column itemname any length you want

Answer:-

```
CREATE TABLE Item(itemname varchar(255);
```

Qno7:-

The second column supplierId as foreignkey in item table

Answer:-

```
CREATE TABLE item (
    itemname varchar(255),
    FOREIGN KEY (supplier_id) REFERENCES Persons(supplier_id)
);
```

Qno8:-

The third column should be itemprice In INT

Answer:-

```
CREATE TABLE item (
    itemname varchar(255),

    Itemprice int NOT NULL,
    FOREIGN KEY (supplier_id) REFERENCES Persons(supplier_id)
);
```

LAB20

Qno1:-

Write a SQL function to convert temperature from Fahrenheit to Celsius scale

Answer:-

```
DECLARE
  temp1    NUMBER := &input_a_temp;
  t_scale   CHAR := '&input_temp_scale';
  new_temp  NUMBER;
  new_scale CHAR;
 BEGIN
  IF t_scale != 'C'
   AND
   t_scale != 'F' THEN
   dbms_output.Put_line ('The scale you input is not a valid scale');
   new_temp := 0;
   new_scale := 'C';
  ELSE
   IF t_scale = 'C' THEN
```

```
        new_temp := ( ( 9 * temp1 ) / 5 ) + 32;

        new_scale := 'F';

      ELSE

        new_temp := ( ( temp1 - 32 ) * 5 ) / 9;

        new_scale := 'C';

      END IF;

    END IF;

    dbms_output.Put_line ('The new temperature in scale '

    ||new_scale

    ||' is: '

    ||new_temp);

  END;

/
```

Usama Farid

FA18-BCS-026

```
Create database DreamHome;

USE DreamHome;

Create table Branch

(

branch No varchar(20)NOTNULL PrimaryKey,

street varchar(100)NOTNULL,

city varchar(20)NOTNULL,

postcode varchar(20)NOTNULL

);

CreatetABLE
```

```
(
stafF No varchar(20)NOTNULL PRIMARYKEY,
fName varchar(15)NOTNULL,
lName varchar(15)NOTNULL,
position varchar(15)NOTNULL,
sex varchar(15)NOTNULL,DOB DateTimeNOTNULL,
salary DECIMALNOTNULL,
branchNovarchar(15)NOTNULLReferences Branch(branchNo)
);
createtable Client
(
client No varchar(20)NOtnullPRIMARYKY,
fName varchar(50)NOTNULL,
lName varchar(50)NOTNULL,
telNo varchar(20)NOTNULL,
PREFType varchar(50)NOTNULL,
maxRent DECMALNOTNULL,
);
CreatetablePrivateOwner
(
ownerNo varchar(20)NOtnullPRIMARYKY,
fName varchar(15)NOTNULL,
lName varchar(15)NOTNULL,
telNo varchar(15)NOTNULL,
);
CreatetablePropertyForRent
(
property No varchar(20)NOtnullPRIMARYKY,
street varchar(50)NOTNULL,
city varchar(15)NOTNULL,
postcode (15)NOTNULL,
rooms intNOTNULL,
rent DECIMALNOTNULL,
ownerNo varchar(20) References PrivateOwner(ownerNo);
staffNo varchar(20) References PrivateOwner(staffNo);
branchNo varchar(20) References PrivateOwner(branchNo);
);
createtable Viewing
(
clientNo varchar(20)NOTNULL References client(clientNo),
propertyNo varchar(15)NOTNULL References client(clientNo),
viewDateTimeNOTNULL,
COMMENT varchar(150)NOTNULL
);
createtable Registration
(
clientNovarchar(50)NOTNULL References client(clientNo),
branchNovarchar(50)NOTNULL References Branch(branchNo),
staffNovarchar(50)NOTNULL   References Staff(staffNo),
dateJoinedDateTimeNOTNULL
);
INSERTinto Branch
(
branchNo,
street,
city,
postcode
)
```

```
VALUES
(N 'B005'N'H#7 I-12/2',N'KAR' '50000'),

(N 'B006'N'H#75 supply',N'ABT' '54000'),
(N 'B007'N'H#79 I-14/2',N'ISB' '55000'),
(N 'B008'N'H#78 Mandian',N'LHR' '55000'),



InserT into Client
(
ClientNo,
fName,
lName,
telNo,
prefType,
maxRent
)
values
(
'81001','Ali','0312789653','yes',1000.0
),
(
'81001','Nabeel','0312789753','yes',1500.0
),
(
'81001','usman','0312789153','no',1800.0
),
(
'81001','Ahmed','0312789253','yes',1300.0
),
(
'81001','Akhtar','0312789553','yes',14400.0
),
(
'81001','Sohail','0312789553','no',1300.0
),
insert into private owner
{
ownnerNo,
fNAME,
lName,
[adress],
telno,
values
(
'B1','Ali','Jamal',F18-4a',0354654264
),
(
'B2','Ajmal','Akmal',F17-4a','0354654264,
),
(
'B3','Khatak','Kamran',F16-4a',03541654264
),
(
'B4','Adnan','Amir',F19-4a',03554654264
),
(
'B5','Inam','Akhtar',F14-4a',03554654264
),
(
'b6','adnan','Umair',F13-4a',03584654264
),

insertinto  viewing
(
Clientno,
PropertyNo,
viewData
comment
)
values
(
'B1001',BF2','2020-3-10','HELLO HOW ARE YOU'
),
(
'B1002',BF3','2020-3-10','HELLO HOW ARE YOU'
),
(
'B1003',BF4','2020-5-10','HELLO HOW ARE YOU'
),
```

Question 2

```
select*from Branch update Branch set city='ABT'where city ='ISB';
```

Lab 7

Qno1:-

Answer

```
select distinct(postcode) from Branch
```

Qno2:-

Answer

```
select distinct(fName) from Staff
```

Qno3:-

Answer

```
select staffNo as [Cadre No], fName as [Baptism Name], lName as [Sur name],
position as [Locale], sex as [Gender],DOB as [Birtday] ,salary as Income,
branchNo as [Section No] from Staff
```

Qno4:-

Answer

```
select clientNo as [Buyer No], fName as [Baptism Name], lName as [Sur name],
telNo as [Fax Number],prefType as [Proclivity Type],maxRent as [Supreme Cost] from
Client;
```

Qno5:-

Answer

```
select * from Staff where salary>10000
```

Qno6:-

Answer

```
select * from Staff where position='Manager' or position='Supervisor'
```

Lab 8

Qno1:-

Answer

```
select staffNo,fName,lName,salary from staff order by salary desc
```

Qno2:-

Answer

```
select propertyNo,type,rooms,rent from PropertyForRent
order by type
select propertyNo,type,rooms,rent
from PropertyForRent
order by type,rent desc
```

Qno3:-

Answer

```
select count(*) as myCount
from PropertyForRent
where rent<=500
```

Qno4:-

Answer

```
select count(Distinct propertyNo) As myCount from Viewing
WHERE viewDate BETWEEN '1-May-04' AND '31-May-04';
```

Qno5:-

Answer

```
select count(staffNo) as myCount,sum(salary) as mySalary from staff
where
position='Manager'
```

Qno6:-

Answer

```
select MIN(salary) as myMin,
MAX(salary) as myMax,
AVG(salary) as myAVG from Staff
```

Qno7:-

Answer

```
SELECT staffNo, fName, lName, position, salary
FROM Staff
WHERE (SELECT AVG(salary) FROM Staff) < salary;
```

Qno8:-

Answer

```
select *from Staff where salary> any(select salary from Staff where branchNo='B003')
```

Qno9:-

Answer

```
select *from Staff where salary> all(select salary from Staff where branchNo='B003')
```

LAB 9

Qno1:

list all tables in the employees database

Answer:

```
 USE EMPLOYEE;

show TABLES;
```

Qno2:

Write a query to find the names (first_name, last_name) and the salaries of the employees who have a higher salary than the employee whose last_name='Bull'.

Answer:-

```
Select FIRST_NAME, LAST_NAME, SALARY

FROM employees

WHERE SALARY>(SELECT salary FROM employees WHERE  last_name='Bull');
```

Qno3:

Answer:-

```
Select first_name , last_name
FROM employees
WHERE department_id
IN(SELECT department_id FROM departments WHERE department_name=’IT’);
```

LAB 10

Qno1:-

Write a query to find the names (first_name, last_name) of the employees who have a manager and work for a department based in the United States.

Answer:-

```
SELECT first_name, last_name FROM employees
WHERE manager_id in(select employee_id FROM employees WHERE department_id
IN(SELECT department_id FROM departments WHERE location_id IN(select location_id from locations
Where country_id=’US’)));
```

Qno2:-

Write a query to find the names (first_name, last_name) of the employees who are managers.

Answer:-

```
SELECT first_name, last_name
FROM employees
WHERE (employee_id IN(SELECT manager_id FROM employees));
```

Qno3:-

Write a query to find the names (first_name, last_name), the salary of the employees whose salary is greater than the average salary

Answer:-

```
SELECT first_name, last_name, salary FROM employees

WHERE salary > (SELECT AVG(salary) FROM employees);
```

Qno4:-

Write a query to find the names (first_name, last_name), the salary of the employees whose salary is equal to the minimum salary for their job grade.

Answer:-

```
SELECT first_name,last_name,salary FROM employees WHERE employees.salary=(SELECT min_salary
FROM jobs WHERE employees.job_id=jobs.job_id);
```

Qno5:-

Write a query to find the names (first_name, last_name), the salary of the employees who earn more than the average salary and who works in any of the IT departments.

Answer:-

```
SELECT first_name,last_name,salary

FROM employees WHERE department_id IN (SELECT department_id FROM departments WHERE
department_name LIKE ‘IT%’)AND salary>(SELECT avg(salary) From employees);
```

Qno6:-

Write a query to find the names (first_name, last_name), the salary of the employees who earn more than Mr. Bell

Answer:-

```
SELECT * FROM employees WHERE salary > ALL(SELECT AVG(salary) FROM employees GROUP BY

Department_id);
```

Qno7:-

Write a query to find the names (first_name, last_name), the salary of the employees who earn the same salary as the minimum salary for all departments

Answer:-

```
SELECT * FROM employees

WHERE salary=(SELECT MIN(salary) FROM employees);
```

Qno8:-

Write a query to find the names (first_name, last_name), the salary of the employees whose salary greater than the average salary of all departments

Answer:-

```
SELECT first_name,last_name from employees whose(SELECT AVG(salary) from departments)
```

Qno9:-

Write a query to find the names (first_name, last_name) and salary of the employees who earn a salary that is higher than the salary of all the Shipping Clerk (JOB_ID = ’SH_CLERK’). Sort the results of the salary of the lowest to highest

Answer:-

```
SELECT first_name,last_name, job_id, salary

FROM employees

WHERE salary >

ALL (SELECT salary FROM employees WHERE job_id = 'SH_CLERK') ORDER BY salary;
```

Qno10:-

.Write a query to find the names (first_name, last_name) of the employees who are not supervisors.

Answer:-

```
SELECT b.first_name,b.last_name

FROM employees b

WHERE NOT EXISTS (SELECT 'X' FROM employees a WHERE a.manager_id = b.employee_id);
```

Qno11:-

Write a query to display the employee ID, first name, last names, and department names of all employees.

Answer:-

```
SELECT employee_id, first_name, last_name,

(SELECT department_name FROM departments d

 WHERE e.department_id = d.department_id) department

 FROM employees e ORDER BY department;
```

Qno12:-

Write a query to display the employee ID, first name, last names, salary of all employees whose salary is above average for their departments

Answer:-

```
SELECT employee_id, first_name

FROM employees AS A

WHERE salary >

(SELECT AVG(salary) FROM employees WHERE department_id = A.department_id);
```

Qno13:-

Write a query to fetch even numbered records from employees table

Answer:-

```
SET @i = 0;

SELECT i, employee_id

FROM (SELECT @i := @i + 1 AS i, employee_id FROM employees)

a WHERE MOD(a.i, 2) = 0;
```

Qno14:-

Write a query to find the 5th maximum salary in the employees table.

Answer:-

```
SELECT DISTINCT salary
FROM employees e1
WHERE 5 = (SELECT COUNT(DISTINCT salary)
FROM employees  e2
WHERE e2.salary >= e1.salary);
```

Qno15:-

Write a query to find the 4th minimum salary in the employees table

Answer:-

```
SELECT DISTINCT salary
FROM employees e1
WHERE 4 = (SELECT COUNT(DISTINCT salary)
FROM employees  e2
WHERE e2.salary <= e1.salary);
```

Qno16:-

Write a query to select last 10 records from a table.

Answer:-

```
SELECT * FROM (
SELECT * FROM employees ORDER BY employee_id DESC LIMIT 10) sub
ORDER BY employee_id ASC;
```

Qno17:-

Write a query to list department number, name for all the departments in which there are no employees in the department

Answer:-

```
SELECT * FROM departments
```

```
WHERE department_id
NOT IN (select department_id FROM employees);
```

Qno18:-

Write a query to get 3 maximum salaries.

Answer:-

```
SELECT DISTINCT salary
FROM employees a
WHERE 3 >= (SELECT COUNT(DISTINCT salary)
FROM employees b
WHERE b.salary >= a.salary)
ORDER BY a.salary DESC;
```

Qno19:-

Write a query to get 3 minimum salaries

Answer:-

```
SELECT DISTINCT salary
FROM employees a
WHERE  3 >= (SELECT COUNT(DISTINCT salary)
FROM employees b
WHERE b.salary <= a.salary)
ORDER BY a.salary DESC;
```

Qno20:-

Write a query to get nth max salaries of employees.Further practice with nested queries

Answer:-

```
SELECT *
```

```
FROM employees emp1

WHERE (1) = (

SELECT COUNT(DISTINCT(emp2.salary))

FROM employees emp2

WHERE emp2.salary > emp1.salary);
```

LAB11

Qno1:-

Create a table tow columns for name and family_name respectively. Insert the names your three friends in lower case case caracters. Write a query to create columns aliased fullname by using the INITCAT() and CONCAT() functions.

Answer:-

```
SELECT customer_id, CONCAT(first_name,second_name,last_name)AS All_names from customer
```

LAB12

Qno1:-

Print countrycode and sum of percentage from countrylangauge, apply groupby on countrycode.

Answer:-

```
SELECT countrycode ,SUM(PER(countrylanguage) from country language Group By(countrycode)
```

Qno2:-

Find sum of any integer column from country table.

Answer:-

Select SUM(population) from country table;

Qno3:-

Count number of records in country table

Answer:-

SELECT COUNT(records) from countrytable;

Qno4:-

Count Distinct (languages) from countrylanguage

Answer:-

SELECT DISTINCT(languages) from countrylanguage;

LAB13

Qno1:-

Select customers name, number , phone from customers table, select checknumber from payments table. Display it for all customers. [either they have made payment or they haven't include all customers].

Answer:-

SELECT customer,phone from customertable;

SELECT checknumber from payment;

LAB14

Qno1:-

Update customer with any a particular order number (you can select any order number).

```
UPDATE customer
SET order_no = 5;
```

Qno2:-

Applying union print data of orders and order details table.

Answer:-

```
SELECT data.customerdata from customers
UNION ALL
SELECT order.orderdata FROM orders;
```

LAB15

Qno1:-

Apply update on any country name.

Answer:-

```
UPDATE country SET column1=”England”  WHERE column=”Islamabad”;
```

Qno2:-

Delete Islamabad city by applying delete query on city table with it’s ID

Answer:-

```
DELETE FROM city
WHERE country.id = ANY (SELECT id FROM city WHERE id = 2);
```

Qno3:-

Try to update values for null column COMM column

Answer:-

```
UPDATE[city]

SET [COMM]=0

WHERE [COMM] is null;
```

Qno4:-

Try to update it for a specific employee whose salary is less than 1000

Answer:-

```
UPDATE employee   WHERE employee<1000;
```

Qno5:-

Answer:-

```
INSERT INTO city
VALUES (“Karachi”, “Lahore”, “pindi”);
```

LAB16

Qno1:-

The first column is called supplier_id which is created as a number data type (maximum 10 digits in length) and cannot contain null values.

Answer:-

```
CREATE TABLE suppliers(supplier_id int(10) NOT NULL);
```

Qno2:-

The second column is called supplier_name which is a varchar2 datatype (50 maximum characters in length) and also can not contain null values

Answer:-

```
CREATE TABLE suppliers(supplier_id int(10) NOT NULL,supplier_name varchar2(50) NOT NULL);
```

Qno3:-

The third column is called address which is a varchar2 data type but can contain null values.

Answer:-

```
CREATE TABLE suppliers(supplier_id int(10) NOT NULL,supplier_name varchar(50) NOT
NULL,address varchar2 NOT NULL);
```

Qno4:-

Define the supplier_id as the primary key

Answer:-

```
CREATE TABLE suppliers(supplier_id NOT NULL AUTO_INCREMENT,supplier_id int(10) NOT
NULL,supplier_name varchar(50) NOT NULL,address varchar2 NOT NULL,PRIMARY
KEY(supplier_id);
```

Qno5:-

Create a second table named as Item with columns:

Answer:-

```
CREATE TABLE Item();
```

Qno6:-

The first column itemname any length you want

Answer:-

```
CREATE TABLE Item(itemname varchar(255);
```

Qno7:-

The second column supplierId as foreignkey in item table

Answer:-

```
CREATE TABLE item (
    itemname varchar(255),
    FOREIGN KEY (supplier_id) REFERENCES Persons(supplier_id)
);
```

Qno8:-

The third column should be itemprice In INT

Answer:-

```
CREATE TABLE item (
    itemname varchar(255),

    Itemprice int NOT NULL,
    FOREIGN KEY (supplier_id) REFERENCES Persons(supplier_id)
);
```

LAB20

Qno1:-

Write a SQL function to convert temperature from Fahrenheit to Celsius scale

Answer:-

```
DECLARE

  temp1    NUMBER := &input_a_temp;

  t_scale   CHAR := '&input_temp_scale';

  new_temp  NUMBER;

  new_scale CHAR;

 BEGIN

  IF t_scale != 'C'

   AND

   t_scale != 'F' THEN

   dbms_output.Put_line ('The scale you input is not a valid scale');

   new_temp := 0;

   new_scale := 'C';

  ELSE

   IF t_scale = 'C' THEN

    new_temp := ( ( 9 * temp1 ) / 5 ) + 32;

    new_scale := 'F';

   ELSE
```

```
    new_temp := ( ( temp1 - 32 ) * 5 ) / 9;

    new_scale := 'C';

   END IF;

  END IF;

  dbms_output.Put_line ('The new temperature in scale '

  ||new_scale

  ||' is: '

  ||new_temp);

 END;

/
```

Usman jadoon

REG No FA18-BCS-100

```
Create database DreamHome;

USE DreamHome;

Create table Branch

(

branch No varchar(20)NOTNULL PrimaryKey,

street varchar(100)NOTNULL,

city varchar(20)NOTNULL,

postcode varchar(20)NOTNULL

);

Create table

(

stafF No varchar(20)NOTNULL PRIMARYKEY,

fName varchar(15)NOTNULL,

lName varchar(15)NOTNULL,
position varchar(15)NOTNULL,
sex varchar(15)NOTNULL,DOB DateTimeNOTNULL,
salary DECIMALNOTNULL,
branchNovarchar(15)NOTNULLReferences Branch(branchNo)
);
createtable Client
(
client No varchar(20)NOtnull PRIMARYKEY,
fName varchar(50)NOTNULL,
lName varchar(50)NOTNULL,
telNo varchar(20)NOTNULL,
PREFType varchar(50)NOTNULL,
maxRent DECMALNOTNULL,
);
CreatetablePrivateOwner
(
ownerNo varchar(20)NOtnullPRIMARYKY,
fName varchar(15)NOTNULL,
lName varchar(15)NOTNULL,
telNo varchar(15)NOTNULL,
);
CreatetablePropertyForRent
(
property No varchar(20)NOtnullPRIMARYKY,
street varchar(50)NOTNULL,
city varchar(15)NOTNULL,
postcode (15)NOTNULL,
rooms intNOTNULL,
rent DECIMALNOTNULL,
ownerNo varchar(20) References PrivateOwner(ownerNo);
staffNo varchar(20) References PrivateOwner(staffNo);
branchNo varchar(20) References PrivateOwner(branchNo);
);
createtable Viewing
(
clientNo varchar(20)NOTNULL References client(clientNo),
propertyNo varchar(15)NOTNULL References client(clientNo),
viewDateTimeNOTNULL,
COMMENT varchar(150)NOTNULL
);
createtable Registration
(
clientNovarchar(50)NOTNULL References client(clientNo),
branchNovarchar(50)NOTNULL References Branch(branchNo),
staffNovarchar(50)NOTNULL   References Staff(staffNo),
dateJoinedDateTimeNOTNULL
);
INSERTinto Branch
(
branchNo,
street,
city,
postcode
)
VALUES
(N 'B005'N'H#7 I-12/2',N'KAR' '50000'),

(N 'B006'N'H#75 supply',N'ABT' '54000'),
(N 'B007'N'H#79 I-14/2',N'ISB' '55000'),
(N 'B008'N'H#78 Mandian',N'LHR' '55000'),


InserT into Client
(
ClientNo,
fName,
lName,
telNo,
prefType,
maxRent
)
values
(
'81001','Ali','0312789653','yes',1000.0
),
(
'81001','Nabeel','0312789753','yes',1500.0
),
(
'81001','usman','0312789153','no',1800.0
),
(
'81001','Ahmed','0312789253','yes',1300.0
),
(
'81001','Akhtar','0312789553','yes',14400.0
```

```
),
(
'81001','Sohail','0312789553','no',1300.0
),
insert into private owner
{
ownnerNo,
fNAME,
lName,
[adress],
telno,
values
(
'B1','Ali','Jamal',F18-4a',0354654264
),
(
'B2','Ajmal','Akmal',F17-4a','0354654264,
),
(
'B3','Khatak','Kamran',F16-4a',03541654264
),
(
'B4','Adnan','Amir',F19-4a',03554654264
),
(
'B5','Inam','Akhtar',F14-4a',03554654264
),
(
'b6','adnan','Umair',F13-4a',03584654264
),

insertinto  viewing
(
Clientno,
PropertyNo,
viewData
comment
)
values
(
'B1001',BF2','2020-3-10','HELLO HOW ARE YOU'
),
(
'B1002',BF3','2020-3-10','HELLO HOW ARE YOU'
),
(
'B1003',BF4','2020-5-10','HELLO HOW ARE YOU'
),
```

Question 2

```
select*from Branch update Branch set city='ABT'where city ='ISB';
```

Lab 7

Qno1:-

Answer

```
select distinct(postcode) from Branch
```

Qno2:-

Answer

```
select distinct(fName) from Staff
```

Qno3:-

Answer

```
select staffNo as [Cadre No], fName as [Baptism Name], lName as [Sur name],
position as [Locale], sex as [Gender],DOB as [Birtday] ,salary as Income,
branchNo as [Section No] from Staff
```

Qno4:-

Answer

```
select clientNo as [Buyer No], fName as [Baptism Name], lName as [Sur name],
telNo as [Fax Number],prefType as [Proclivity Type],maxRent as [Supreme Cost] from
Client;
```

Qno5:-

Answer

```
select * from Staff where salary>10000
```

Qno6:-

Answer

```
select * from Staff where position='Manager' or position='Supervisor'
```

Lab 8

Qno1:-

Answer

```
select staffNo,fName,lName,salary from staff order by salary desc
```

Qno2:-

Answer

```
select propertyNo,type,rooms,rent from PropertyForRent
order by type
select propertyNo,type,rooms,rent
from PropertyForRent
order by type,rent desc
```

Qno3:-

Answer

```
select count(*) as myCount
from PropertyForRent
where rent<=500
```

Qno4:-

Answer

```
select count(Distinct propertyNo) As myCount from Viewing
WHERE viewDate BETWEEN '1-May-04' AND '31-May-04';
```

Qno5:-

Answer

```
select count(staffNo) as myCount,sum(salary) as mySalary from staff
where
position='Manager'
```

Qno6:-

Answer

```
select MIN(salary) as myMin,
MAX(salary) as myMax,
AVG(salary) as myAVG from Staff
```

Qno7:-

Answer

```
SELECT staffNo, fName, lName, position, salary
FROM Staff
WHERE (SELECT AVG(salary) FROM Staff) < salary;
```

Qno8:-

Answer

```
select *from Staff where salary> any(select salary from Staff where branchNo='B003')
```

Qno9:-

Answer

```
select *from Staff where salary> all(select salary from Staff where branchNo='B003')
```

LAB 9

Qno1:

list all tables in the employees database

Answer:

USE EMPLOYEE;

show TABLES;

Qno2:

Write a query to find the names (first_name, last_name) and the salaries of the employees who have a higher salary than the employee whose last_name='Bull'.

Answer:-

Select FIRST_NAME, LAST_NAME, SALARY

FROM employees

WHERE SALARY>(SELECT salary FROM employees WHERE last_name='Bull');

Qno3:

Answer:-

```
Select first_name , last_name

FROM employees

WHERE department_id

IN(SELECT department_id FROM departments WHERE department_name=’IT’);
```

LAB 10

Qno1:-

Write a query to find the names (first_name, last_name) of the employees who have a manager and work for a department based in the United States.

Answer:-

```
SELECT first_name, last_name FROM employees

WHERE manager_id in(select employee_id FROM employees WHERE department_id

IN(SELECT department_id FROM departments WHERE location_id IN(select location_id from locations

Where country_id=’US’)));
```

Qno2:-

Write a query to find the names (first_name, last_name) of the employees who are managers.

Answer:-

```
SELECT first_name, last_name

FROM employees

WHERE (employee_id IN(SELECT manager_id FROM employees));
```

Qno3:-

Write a query to find the names (first_name, last_name), the salary of the employees whose salary is greater than the average salary

Answer:-

```
SELECT first_name, last_name, salary FROM employees

WHERE salary > (SELECT AVG(salary) FROM employees);
```

Qno4:-

Write a query to find the names (first_name, last_name), the salary of the employees whose salary is equal to the minimum salary for their job grade.

Answer:-

```
SELECT first_name,last_name,salary FROM employees WHERE employees.salary=(SELECT min_salary
FROM jobs WHERE employees.job_id=jobs.job_id);
```

Qno5:-

Write a query to find the names (first_name, last_name), the salary of the employees who earn more than the average salary and who works in any of the IT departments.

Answer:-

```
SELECT first_name,last_name,salary

FROM employees WHERE department_id IN (SELECT department_id FROM departments WHERE
department_name LIKE ‘IT%’)AND salary>(SELECT avg(salary) From employees);
```

Qno6:-

Write a query to find the names (first_name, last_name), the salary of the employees who earn more than Mr. Bell

Answer:-

```
SELECT * FROM employees WHERE salary > ALL(SELECT AVG(salary) FROM employees GROUP BY

Department_id);
```

Qno7:-

Write a query to find the names (first_name, last_name), the salary of the employees who earn the same salary as the minimum salary for all departments

Answer:-

```
SELECT * FROM employees

WHERE salary=(SELECT MIN(salary) FROM employees);
```

Qno8:-

Write a query to find the names (first_name, last_name), the salary of the employees whose salary greater than the average salary of all departments

Answer:-

```
SELECT first_name,last_name from employees whose(SELECT AVG(salary) from departments)
```

Qno9:-

Write a query to find the names (first_name, last_name) and salary of the employees who earn a salary that is higher than the salary of all the Shipping Clerk (JOB_ID = ’SH_CLERK’). Sort the results of the salary of the lowest to highest

Answer:-

```
SELECT first_name,last_name, job_id, salary

FROM employees

WHERE salary >

ALL (SELECT salary FROM employees WHERE job_id = 'SH_CLERK') ORDER BY salary;
```

Qno10:-

.Write a query to find the names (first_name, last_name) of the employees who are not supervisors.

Answer:-

```
SELECT b.first_name,b.last_name

FROM employees b

WHERE NOT EXISTS (SELECT 'X' FROM employees a WHERE a.manager_id = b.employee_id);
```

Qno11:-

Write a query to display the employee ID, first name, last names, and department names of all employees.

Answer:-

```
SELECT employee_id, first_name, last_name,
(SELECT department_name FROM departments d
 WHERE e.department_id = d.department_id) department
 FROM employees e ORDER BY department;
```

Qno12:-

Write a query to display the employee ID, first name, last names, salary of all employees whose salary is above average for their departments

Answer:-

```
SELECT employee_id, first_name
FROM employees AS A
WHERE salary >
(SELECT AVG(salary) FROM employees WHERE department_id = A.department_id);
```

Qno13:-

Write a query to fetch even numbered records from employees table

Answer:-

```
SET @i = 0;
SELECT i, employee_id
FROM (SELECT @i := @i + 1 AS i, employee_id FROM employees)
a WHERE MOD(a.i, 2) = 0;
```

Qno14:-

Write a query to find the 5th maximum salary in the employees table.

Answer:-

```
SELECT DISTINCT salary
FROM employees e1
WHERE 5 = (SELECT COUNT(DISTINCT salary)
FROM employees  e2
WHERE e2.salary >= e1.salary);
```

Qno15:-

Write a query to find the 4th minimum salary in the employees table

Answer:-

```
SELECT DISTINCT salary
FROM employees e1
WHERE 4 = (SELECT COUNT(DISTINCT salary)
FROM employees  e2
WHERE e2.salary <= e1.salary);
```

Qno16:-

Write a query to select last 10 records from a table.

Answer:-

```
SELECT * FROM (
SELECT * FROM employees ORDER BY employee_id DESC LIMIT 10) sub
ORDER BY employee_id ASC;
```

Qno17:-

Write a query to list department number, name for all the departments in which there are no employees in the department

Answer:-

```
SELECT * FROM departments
WHERE department_id
```

```
NOT IN (select department_id FROM employees);
```

Qno18:-

Write a query to get 3 maximum salaries.

Answer:-

```
SELECT DISTINCT salary
FROM employees a
WHERE 3 >= (SELECT COUNT(DISTINCT salary)
FROM employees b
WHERE b.salary >= a.salary)
ORDER BY a.salary DESC;
```

Qno19:-

Write a query to get 3 minimum salaries

Answer:-

```
SELECT DISTINCT salary
FROM employees a
WHERE  3 >= (SELECT COUNT(DISTINCT salary)
FROM employees b
WHERE b.salary <= a.salary)
ORDER BY a.salary DESC;
```

Qno20:-

Write a query to get nth max salaries of employees.Further practice with nested queries

Answer:-

```
SELECT *
FROM employees emp1
```

```
WHERE (1) = (
SELECT COUNT(DISTINCT(emp2.salary))
FROM employees emp2
WHERE emp2.salary > emp1.salary);
```

LAB11

Qno1:-

Create a table tow columns for name and family_name respectively. Insert the names your three friends in lower case case caracters. Write a query to create columns aliased fullname by using the INITCAT() and CONCAT() functions.

Answer:-

```
SELECT customer_id, CONCAT(first_name,second_name,last_name)AS All_names from customer
```

LAB12

Qno1:-

Print countrycode and sum of percentage from countrylangauge, apply groupby on countrycode.

Answer:-

```
SELECT countrycode ,SUM(PER(countrylanguage) from country language Group By(countrycode)
```

Qno2:-

Find sum of any integer column from country table.

Answer:-

Select SUM(population) from country table;

Qno3:-

Count number of records in country table

Answer:-

SELECT COUNT(records) from countrytable;

Qno4:-

Count Distinct (languages) from countrylanguage

Answer:-

SELECT DISTINCT(languages) from countrylanguage;

LAB13

Qno1:-

Select customers name, number , phone from customers table, select checknumber from payments table. Display it for all customers. [either they have made payment or they haven't include all customers].

Answer:-

SELECT customer,phone from customertable;

SELECT checknumber from payment;

LAB14

Qno1:-

Update customer with any a particular order number (you can select any order number).

```
UPDATE customer
SET order_no = 5;
```

Qno2:-

Applying union print data of orders and order details table.

Answer:-

```
SELECT data.customerdata from customers
UNION ALL
SELECT order.orderdata FROM orders;
```

LAB15

Qno1:-

Apply update on any country name.

Answer:-

UPDATE country SET column1=”England” WHERE column=”Islamabad”;

Qno2:-

Delete Islamabad city by applying delete query on city table with it’s ID

Answer:-

```
DELETE FROM city
WHERE country.id = ANY (SELECT id FROM city WHERE id = 2);
```

Qno3:-

Try to update values for null column COMM column

Answer:-

```
UPDATE[city]

SET [COMM]=0

WHERE [COMM] is null;
```

Qno4:-

Try to update it for a specific employee whose salary is less than 1000

Answer:-

```
UPDATE employee   WHERE employee<1000;
```

Qno5:-

Answer:-

```
INSERT INTO city
VALUES (“Gujrat”, “Gujranwala”, “lahore”);
```

LAB16

Qno1:-

The first column is called supplier_id which is created as a number data type (maximum 10 digits in length) and cannot contain null values.

Answer:-

```
CREATE TABLE suppliers(supplier_id int(10) NOT NULL);
```

Qno2:-

The second column is called supplier_name which is a varchar2 datatype (50 maximum characters in length) and also can not contain null values

Answer:-

CREATE TABLE suppliers(supplier_id int(10) NOT NULL,supplier_name varchar2(50) NOT NULL);

Qno3:-

The third column is called address which is a varchar2 data type but can contain null values.

Answer:-

CREATE TABLE suppliers(supplier_id int(10) NOT NULL,supplier_name varchar(50) NOT NULL,address varchar2 NOT NULL);

Qno4:-

Define the supplier_id as the primary key

Answer:-

CREATE TABLE suppliers(supplier_id NOT NULL AUTO_INCREMENT,supplier_id int(10) NOT NULL,supplier_name varchar(50) NOT NULL,address varchar2 NOT NULL,PRIMARY KEY(supplier_id);

Qno5:-

Create a second table named as Item with columns:

Answer:-

CREATE TABLE Item();

Qno6:-

The first column itemname any length you want

Answer:-

CREATE TABLE Item(itemname varchar(255);

Qno7:-

The second column supplierId as foreignkey in item table

Answer:-

```
CREATE TABLE item (
    itemname varchar(255),
    FOREIGN KEY (supplier_id) REFERENCES Persons(supplier_id)
);
```

Qno8:-

The third column should be itemprice In INT

Answer:-

```
CREATE TABLE item (
    itemname varchar(255),

    Itemprice int NOT NULL,
    FOREIGN KEY (supplier_id) REFERENCES Persons(supplier_id)
);
```

LAB17

LAB20

Qno1:-

Write a SQL function to convert temperature from Fahrenheit to Celsius scale

Answer:-

```
DECLARE

  temp1    NUMBER := &input_a_temp;
  t_scale   CHAR := '&input_temp_scale';
  new_temp  NUMBER;
  new_scale CHAR;
 BEGIN
  IF t_scale != 'C'
   AND
   t_scale != 'F' THEN
   dbms_output.Put_line ('The scale you input is not a valid scale');
   new_temp := 0;
   new_scale := 'C';
  ELSE
   IF t_scale = 'C' THEN
    new_temp := ( ( 9 * temp1 ) / 5 ) + 32;
    new_scale := 'F';
   ELSE
    new_temp := ( ( temp1 - 32 ) * 5 ) / 9;
    new_scale := 'C';
   END IF;
  END IF;
  dbms_output.Put_line ('The new temperature in scale '
  ||new_scale
  ||' is: '
  ||new_temp);
 END;
/
```