```
CREATE TABLE EmployeeInfo_A (
  EmpId INT PRIMARY KEY,
  EmpFname VARCHAR(255),
  EmpLname VARCHAR(255),
  Department VARCHAR(255),
  Project VARCHAR(255),
  Address VARCHAR(255),
  DOB DATE,
  Gender VARCHAR(255)
);
INSERT INTO EmployeeInfo_A (EmpId, EmpFname, EmpLname, Department, Project, Address, DOB, Gender)
VALUES (1, 'Suraj', 'Patil', 'Admin', 'P1', 'Delhi', TO_DATE('02-12-1979', 'DD-MM-YYYY'), 'Male');

INSERT INTO EmployeeInfo_A (EmpId, EmpFname, EmpLname, Department, Project, Address, DOB, Gender)
VALUES (2, 'Sushant', 'Patil', 'Admin', 'P2', 'Mumbia', TO_DATE('10-10-1986', 'DD-MM-YYYY'), 'Male');

INSERT INTO EmployeeInfo_A (EmpId, EmpFname, EmpLname, Department, Project, Address, DOB, Gender)
VALUES (3, 'Sonia', 'Banerjee', 'HR', 'P3', 'Pune', TO_DATE('05-06-1983', 'DD-MM-YYYY'), 'Female');

INSERT INTO EmployeeInfo_A (EmpId, EmpFname, EmpLname, Department, Project, Address, DOB, Gender)
VALUES (4, 'Ankita', 'Kapoor', 'HR', 'P4', 'Chennai', TO_DATE('28-11-1983', 'DD-MM-YYYY'), 'Female');

INSERT INTO EmployeeInfo_A (EmpId, EmpFname, EmpLname, Department, Project, Address, DOB, Gender)
VALUES (5, 'Swati', 'Garg', 'HR', 'P5', 'Delhi', TO_DATE('06-04-1991', 'DD-MM-YYYY'), 'Female');
```

```
CREATE TABLE EmployeePosition_A (
  EmpId INT PRIMARY KEY,
  EmpPosition VARCHAR(255),
  DateOfJoining DATE,
  Salary INT,
  CONSTRAINT EmployeePosition FOREIGN KEY (EmpId) REFERENCES EmployeeInfo(EmpId)
);
insert into EmployeePosition_A Values(1,'Manager',TO_DATE('03-04-2020','DD-MM-YYYY'),500000);
insert into EmployeePosition_A Values('2','Manager',TO_DATE('03-04-2020','DD-MM-YYYY'),500000);
insert into EmployeePosition_A Values('3','Manager',TO_DATE('02-04-2020','DD-MM-YYYY'),150000);
insert into EmployeePosition_A Values('4','Officer',TO_DATE('02-04-2020','DD-MM-YYYY'),90000);
insert into EmployeePosition_A Values('5','Manager',TO_DATE('03-04-2020','DD-MM-YYYY'),300000);
```

```
create table EmpPerform(
Emp_id int Primary key,
Bones int
);
insert into EmpPerform values(1,56600);
insert into EmpPerform values(2,66600);
insert into EmpPerform values(3,60600);
insert into EmpPerform values(4,55600);
insert into EmpPerform values(5,45600);
insert into EmpPerform values(6,45000);
insert into EmpPerform values(7,40000);
insert into EmpPerform values(8,40000);
```

---

1. **Write a query to fetch the EmpFname from the EmployeeInfo table in upper case and use the ALIAS name as EmpName.**

   SELECT UPPER(EmpFname) AS EmpName FROM EmployeeInfo;

2. **Write a program of Database connectivity and insert records**

3. **Write q query to find all the employees whose salary is between 50000 to 100000.**

   SELECT * FROM EmployeePosition WHERE Salary BETWEEN '50000' AND '100000';

4. **Write a query to find the names of employees that begin with 'S'**

   SELECT * FROM EmployeeInfo WHERE EmpFname LIKE 'S%';

5. **Write a program of Database connectivity and update records**

6. **Write a query to fetch all the records from the EmployeeInfo table ordered by**

   SELECT *
   FROM EmployeeInfo_A
   ORDER BY empid DESC;

7. **EmpLname in descending order and Department in the ascending order.**

```
1  SELECT * FROM EmployeeInfo ORDER BY EmpFname desc, Department asc;
```

8. **Write a query to fetch details of all employees excluding the employees with first names, "Sanjay" and "Sonia" from the EmployeeInfo table.**

```
9.  1    SELECT * FROM EmployeeInfo WHERE EmpFname NOT IN ('Sanjay','Sonia');
```

10. **Draw an E-R Diagram of Hospital organization.**

11. Write a program of Database connectivity and display records

12. Use DDL Queries to create, alter (add, modify, rename, drop) & drop Tables.

```
SELECT *FROM TABLE_NAME;

Create table Person(
person_id int primary key,
person_name varchar(100),
address varchar(100));

ALTER TABLE table_name

 ADD column_name datatype;

Update Person

Set address = 'Gadhinglaj'

Where person_Roll_NO = 7;
```

13. Use DML Queries to insert, delete, update & display records of the tables.

```
Delete Person
Where Person_roll_no = '6';
```

14. Create table with integrity constraints like primary key, check, not null and unique.

15. Write a program of Database connectivity and delete records

16. Display the records using Aggregate functions like min, max, avg, sum & count. Also use group by, having clauses.

```
Select count(Emp_id)
from EmpPerform;
```

```
Select Sum(Bones)
from EmpPerform;
Select Avg(bones)
from EmpPerform;

Select Max(bones)
from EmpPerform;

Select Min(bones)
from EmpPerform;

Select EmpPosition, AVG(salary) as avg_salary
From EmployeePosition_A
GROUP BY EmpPosition
HAVING AVG(salary)>50000;
```

17. Display the results of Join Operations like inner join, left outer join, right outer join and full outer join.

```
SELECT * FROM EmployeePosition_A
INNER JOIN EmpPerform
ON EmpPerform.Emp_id=EmployeePosition_A.Empid;

SELECT
 EmployeePosition_A.Empid,EmployeePosition_A.EmpPosition,EmpPerform.bones
from EmployeePosition_A
LEFT JOIN EmpPerform
ON EmpPerform.Emp_id=EmployeePosition_A.Empid;

SELECT EmployeePosition_A.Empid,EmployeePosition_A.EmpPosition,EmpPerform.bones
from EmployeePosition_A
RIGHT JOIN EmpPerform
ON EmpPerform.Emp_id=EmployeePosition_A.Empid;
```

```
SELECT EmployeePosition_A.Empid,EmployeePosition_A.EmpPosition,EmpPerform.bones
from EmployeePosition_A
FULL OUTER JOIN EmpPerform
ON EmpPerform.Emp_id=EmployeePosition_A.Empid;
```

18. Display the results of set operations like union, intersections & set difference.

```
SELECT * FROM Civil_student
UNION
SELECT * FROM Cse_student;

SELECT Student_id, Full_name FROM Civil_Student
INTERSECT
SELECT Student_id, Full_name FROM Cse_Student;

SELECT Student_id,Batch
FROM Civil_Student
MINUS
SELECT Student_id,Batch
FROM Cse_Student;
```

19. Draw an E-R Diagram of E-Commerce Website.

20. Write a query to fetch details of all employees with first names, “Sanjay” and “Sonia” from the EmployeeInfo table.

```
SELECT *
FROM EmployeeInfo_A
WHERE EmpFname IN ('Sanjay', 'Sonia');
```

21. Display the all records in descending order, where country is "India" AND city is "kolhapur"

```
SELECT *
FROM EmployeeInfo_A
WHERE country = 'India' AND city = 'Kolhapur'
```

```
ORDER BY Empid DESC;
```

22. Display the all records in acceding order, where country is "India" OR “Jarman”

```
SELECT *
FROM EmployeeInfo
WHERE country IN ('India', 'Jarman')
ORDER BY record_id ASC;
```