```
create database Employee;
use  Employee;
## QUERY 1
create table data_science_team(
EMP_ID varchar(50),
FIRST_NAME char(50),
LAST_NAME char(50),
GENDER char(50),
ROLE_NAME char(50),
DEPT CHAR(50),
EXP INT,
COUNTRY CHAR(100),
CONTINENT CHAR(100));
SELECT * FROM proj_table;
insert into
proj_table(PROJECT_ID,PROJ_NAME,DOMAIN,START_DATE,CLOSURE_DATE,DEV_QTR,STATUS)
values
('P103','Drug Discovery','HEALTHCARE','2021-04-06','2021-06-20','Q1','DONE'),
('P105','Fraud Detection','FINANCE','2021-11-06','2021-06-25','Q1','DONE'),
('P109','Market Basket
Analysis','RETAIL','2021-04-12','2021-06-30','Q1','DELAYED'),
('P204','Supply Chain
Management','AUTOMOTIVE','2021-07-21','2021-09-28','Q2','WIP'),
('P302','Early Detection of Lung
Cancer','HEALTHCARE','2021-10-08','2021-12-08','Q3','YTS'),
('P406','Customer Sentiment
Analysis','RETAIL','2021-07-09','2021-09-24','Q2','WIP');
select * from emp_record_table;
## QUERY 2data_science_teamproj_tableemp_record_tabledata_science_team
## Create ER diagramdata_science_team
## QUERY 3
Select EMP_ID, FIRST_NAME, LAST_NAME, GENDER,DEPT from emp_record_table;
# QUERY 4
SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT,EMP_RATING
FROM emp_record_table
WHERE EMP_RATING<2;
SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT,EMP_RATING
FROM emp_record_table
WHERE EMP_RATING>4;
SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT,EMP_RATING
FROM emp_record_table
WHERE EMP_RATING BETWEEN 2 AND 4;
## QUERY  4 Write a query to concatenate the FIRST_NAME and the LAST_NAME
## of employees in the Finance department from the employee table and then give the
resultant column alias as NAME.
SELECT concat(FIRST_NAME,' ',LAST_NAME) AS NAME,DEPT
FROM emp_record_table
WHERE DEPT='Finance';
## QUERY 6 Write a query to list only those employees who have someone reporting to
them. Also
```

```
## show the number of reporters (including the TPresident).
SELECT EMP_ID,FIRST_NAME,MANAGER_ID,ROLE FROM emp_record_table
WHERE MANAGER_ID IS NOT NULL;
# QUERY 7 Write a query to list down all the employees from the healthcare and
finance departments using union. Take data from the employee record table.
SELECT FIRST_NAME,LAST_NAME , DEPT
FROM emp_record_table
WHERE DEPT='HEALTHCARE'
UNION
SELECT FIRST_NAME,LAST_NAME , DEPT
FROM emp_record_table
WHERE DEPT='FINANCE';
## QUERY 8 Write a query to list down employee details such as EMP_ID, FIRST_NAME,
LAST_NAME, ROLE, DEPARTMENT, and EMP_RATING grouped by dept.
##Also include the respective employee rating along with the max emp rating for the
department.
SELECT  EMP_ID, FIRST_NAME, LAST_NAME,ROLE, DEPT, EMP_RATING,max(EMP_RATING)
FROM emp_record_table
group by DEPT;
# QUERY 9 Write a query to calculate the minimum and the maximum salary of the
employees in each role.
 ##Take data from the employee record table
 SELECT EMP_ID,ROLE,MAX(SALARY),MIN(SALARY) FROM emp_record_table
 GROUP BY ROLE ;
 # QUERY 10 Write a query to assign ranks to each employee based on their
experience.
 ## Take data from the employee record table.
 SELECT EMP_ID, EXP, RANK() OVER ( ORDER BY EXP) AS EMP_RANK
 FROM emp_record_table;
 # QUERY 11 Write a query to create a view that displays employees in various
countries whose salary is more than six thousand.
 ##Take data from the employee record table.
 SELECT EMP_ID , FIRST_NAME,COUNTRY,SALARY
 FROM emp_record_table
 WHERE SALARY >6000;
 ## QUERY 12 Write a nested query to find employees with experience of more than
ten years.
SELECT * FROM emp_record_table WHERE EXP IN
(SELECT EXP FROM emp_record_table WHERE EXP>10);
 # QUERY 13 Write a query to create a stored procedure to retrieve the details of
the employees whose experience is more than three years.
 ##Take data from the employee record table.
DELIMITER $$
 CREATE PROCEDURE EmployeeRecord(IN EMP_EXP INT)
 BEGIN
 SELECT * FROM emp_record_table
 WHERE EXP>EMP_EXP;
 END$$
 DELIMITER ;
 CALL EmployeeRecord('3');
```

```
## QUERY 14 Write a query using stored functions in the project table
## to check whether the job profile assigned to each employee in the data science
## team matches the organization’s set standard.
DELIMITER $$
CREATE FUNCTION `JOBPROFILE` (PROFILE INT)
RETURNS varchar(100)
DETERMINISTIC
BEGIN
DECLARE JobPROFILE varchar(50);
if PROFILE<=2 THEN
               SET JobPROFILE ='JUNIOR DATA SCIENTIST';
ELSEIF (PROFILE>=2 AND PROFILE<=5 ) THEN
                                SET JobPROFILE='ASSOCIATE DATA SCIENTIST';
ELSEIF (PROFILE>=5 AND PROFILE<=10) THEN
                                SET JobPROFILE= 'SENIOR DATA SCIENTIST';
ELSEIF (PROFILE>=10 AND PROFILE<=12) THEN
                                SET  JobPROFILE='LEAD SENIOR DATA SCIENTIST';
ELSEIF (PROFILE>=12 AND PROFILE<=16) THEN
                                SET JobPROFILE= 'MANAGER';
END IF;
                RETURN (JobPROFILE);
END $$
DELIMITER ;
SELECT EMP_ID,EXP,JOBPROFILE(EXP) FROM data_science_team;
# QUERY 15 Create an index to improve the cost and performance of the query to find
the employee whose
## FIRST_NAME is ‘Eric’ in the employee table after checking the execution plan.
alter TABLE emp_record_table ADD INDEX emp_index (FIRST_NAME(10));
EXPLAIN SELECT * FROM emp_record_table WHERE FIRST_NAME='Eric';
## QUERY 16 Write a query to calculate the bonus for all the employees, based on
their ratings and salaries
##(Use the formula: 5% of salary * employee rating).
SELECT EMP_ID,FIRST_NAME,SALARY,(0.05 * SALARY)*(EMP_RATING) AS BONUS ,EMP_RATING
FROM emp_record_table;
# QUERY 17 Write a query to calculate the average salary distribution based on the
continent and country.
## Take data from the employee record table.
select COUNTRY,SALARY,CONTINENT,SUM(distinct SALARY)/COUNT(DISTINCT COUNTRY) AS
AVG_SALARY_COUNTRY ,
SUM(distinct SALARY)/COUNT(DISTINCT CONTINENT) AS AVG_SALARY_CONTINENT
FROM emp_record_table
GROUP BY COUNTRY;
```