```
#1
CREATE DATABASE employee;
USE employee;

#3
SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT AS DEPARTMENT
FROM emp_record_table;

#4
SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT AS DEPARTMENT, EMP_RATING
FROM emp_record_table
WHERE EMP_RATING < 2;

SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT AS DEPARTMENT, EMP_RATING
FROM emp_record_table
WHERE EMP_RATING > 4;

SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT AS DEPARTMENT, EMP_RATING
FROM emp_record_table
WHERE EMP_RATING BETWEEN 2 AND 4;

#5
SELECT EMP_ID, FIRST_NAME, LAST_NAME, CONCAT(FIRST_NAME, " ", LAST_NAME) AS
NAME
FROM emp_record_table
WHERE DEPT="FINANCE";

#6
SELECT DISTINCT(t2.MANAGER_ID), CONCAT(t1.FIRST_NAME, " ", t1.LAST_NAME) AS NAME
FROM emp_record_table t1 RIGHT JOIN   emp_record_table t2
ON t1.EMP_ID=t2.MANAGER_ID ORDER BY t2.MANAGER_ID;

#7
SELECT EMP_ID, CONCAT(FIRST_NAME, " ", LAST_NAME) AS NAME FROM emp_record_table
WHERE DEPT="FINANCE"
UNION
SELECT EMP_ID, CONCAT(FIRST_NAME, " ", LAST_NAME) AS NAME FROM emp_record_table
WHERE DEPT="HEALTHCARE";

#8
SELECT t1.EMP_ID, t1.FIRST_NAME, t1.LAST_NAME, t1.ROLE, t1.DEPT, t1.EMP_RATING,
t2.Max_rating_in_dept
FROM emp_record_table t1 LEFT JOIN (SELECT DEPT, MAX(EMP_RATING) AS
Max_rating_in_dept FROM emp_record_table GROUP BY DEPT) AS t2
```

```
ON t1.DEPT=t2.DEPT
ORDER BY DEPT;

#9
SELECT ROLE, MAX(SALARY), MIN(SALARY) FROM emp_record_table GROUP BY ROLE;

#10
SELECT EMP_ID, FIRST_NAME, LAST_NAME, DENSE_RANK() OVER(ORDER BY EXP) FROM emp_record_table;

#11
CREATE VIEW v AS SELECT EMP_ID, FIRST_NAME, LAST_NAME, COUNTRY FROM emp_record_table WHERE SALARY > 6000;
SELECT * FROM v;

#12
SELECT EMP_ID, FIRST_NAME, LAST_NAME, EXP FROM emp_record_table WHERE EXP >10;

#13
DELIMITER //
CREATE PROCEDURE ret()
BEGIN
   SELECT EMP_ID, FIRST_NAME, LAST_NAME, EXP FROM emp_record_table WHERE EXP >3;
END //
CALL ret;

#14
DELIMITER //
CREATE FUNCTION EMPLOYEE_ROLE_MATCH(EXP INT)
RETURNS VARCHAR(50)
DETERMINISTIC
BEGIN
DECLARE STANDARD_ROLE VARCHAR(50);
IF EXP < 2 THEN SET STANDARD_ROLE ='JUNIOR DATA SCIENTIST';
ELSEIF ( EXP >=2 and EXP <5) THEN SET STANDARD_ROLE ='ASSOCIATE DATA SCIENTIST';
ELSEIF ( EXP >=5 and EXP <10) THEN SET STANDARD_ROLE ='SENIOR DATA SCIENTIST';
ELSEIF ( EXP >=10 and EXP <12) THEN SET STANDARD_ROLE ='LEAD DATA SCIENTIST';
ELSEIF ( EXP >=12 and EXP <16) THEN SET STANDARD_ROLE ='MANAGER';
END IF;
RETURN (STANDARD_ROLE);
END //

select EMP.EMP_ID,DS.ROLE,EMP.EXP,EMPLOYEE_ROLE_MATCH(EMP.EXP)
from emp_record_table EMP
```

```
JOIN data_science_team DS
ON EMP.EMP_ID=DS.EMP_ID
;

#15
SELECT EMP_ID, FIRST_NAME, LAST_NAME FROM emp_record_table WHERE
FIRST_NAME="Eric";
EXPLAIN SELECT EMP_ID, FIRST_NAME, LAST_NAME FROM emp_record_table WHERE
FIRST_NAME="Eric";
SHOW INDEX FROM emp_record_table;
CREATE INDEX iDX ON emp_record_table(FIRST_NAME(100));
SELECT EMP_ID, FIRST_NAME, LAST_NAME FROM emp_record_table WHERE
FIRST_NAME="Eric";
EXPLAIN SELECT EMP_ID, FIRST_NAME, LAST_NAME FROM emp_record_table WHERE
FIRST_NAME="Eric";

#16
SELECT EMP_ID, FIRST_NAME, LAST_NAME, 0.05*SALARY*EMP_RATING AS BONUS FROM
emp_record_table;
#17
SELECT Continent, Country, AVG(Salary) AS Average_Salary FROM emp_record_table GROUP
BY
    Continent, Country;
```