Create the below tables along with key constraints and write and insert script for insertion of rows with subsitution variables and insert appropriate data.

```
Create table department
(
Dept_no nemeric,
Dept_name varchar(10),
location varchar(10)
);

insert into department values
(1, 'sales', 'Delhi'),
(2, 'IT', 'Mumbai'),
(3, 'Production', 'Mumbai'),
(4, 'Marketing', 'Ahmadabad'),
(5, 'Analysis', 'surat'),
(6, 'BCA', 'MP'),
(7, 'BBA', 'Baroda');
```

* Queries:

1. Display all Department belonging to location 'surat'.

→ select * from department where location = "surat";

2.

2. list all department name starting with 'A'.
→ select * From department where dept_name like "a%";

3. list all department whose number is between 1 and 100.
→ select * From department where dept_no between 1 and 100;

4. Delete 'TRG' Department.
→ Delete From department where dept_name = "TRG";

5. change department name 'BBA' to 'IT'.
→ update department set dept_name = "BBA". where dept_name = "IT";

6. update the location whose dept_name second letter is "a"
→ update department set location = "Goa" where dept_name like "_a%";

7. Display whose location is 'Baroda', 'surat', and 'Ahemabadad'.
→ select * From department where location = "surat" or "Baroda" or "Ahemdabad";

8. Display whose are not From "sales" and "marketing" department.

vision

→ select * from department where dept_name not like "sales" and "marketing";

9. list all records of each table in as order.

→ select * from department order by department;
select * from employee order by emp_name;

Employee (emp_id, emp_name, gender, dept_no, address, designation, salary, experience, email);

```
create table employee
(
emp_id numeric,
emp_name varchar(10),
gender varchar(10),
dept_no int references department(Dept_no),
address varchar(10),
designation varchar(10),
Salary numeric,
experience numeric,
email varchar(20)
);

insert into employee values
(1, 'Diya', 'Female', '1', 'Orna', 'CEO', '5000',
'3 Yrs', 'Diya@gmail.com'),
(2, 'Prachi', 'Female', '2', 'Bardoli', 'Manager',
8000, '2 Yrs', 'Prachi@gmail.com'),
```

(3, 'Pankaj', 'Male', 'Navsari', 'CEO', 10,000, 3 'Yrs', 'Panki@gmail.com'),
(4, 'Parth', 'Male', 'Kadod', 'clerk', 5000, '1 Yrs', 'Parth@gmail.com'),
(5, 'Alka', 'Female', 'Kamrej', 'Manager', 9000, '2 Yrs', 'Alka@gmail.com');

* Queries:

10. Display female employee list
→ select * From employee where gender = "Female";

11. Display all record by order emp_name
→ select * From employee order by emp_name Asc;

12. Find name of employee whose salary less than 5000 and greater than 2000.
→ select emp_name From employee where salary < 5000 and salary > 2000;

13. Display names and the designation of all female employee in designation order.
→ select emp_name, designation From employee where gender = "Female" order by emp_no desc;

14. Display name of all the employee whose name start with 'A' ends with 'A';

→ select emp_name From employee where emp_name like 'a%';

15. Find the name of employee and salary for these who had obtain manimum salary.

→ select emp_name, min(salary) as "lowest salary" From employee;

16. Add 10% raise in salary of all employee whose department is 'IT'.

→ update employee set salary = (salary * 100)/100 where designation = "manager";

17. list name of employee who are fresher's class than 1 yer of experience.

→ select emp_name From employee where experience < 1 Yrs;

18. list department wise name of employee who has more than 5 years of experiance.

→ select emp_name experience From employee where experience < 5;

→ select employee emp_name From department inner join employee on department_dept_no = employee.dept_no where experience > 5 order by dept_name ASC;

19. list department having no employee
→ select * from department where dept_no not in (select dept_no from employee);

20. Delete the employee whose salary is less than 10,000.
→ Delete from employee where salary < 10,000;

Create the below three tables along with key constraints and write and insert script for insertions of rows with subsitution variables and insert appropriate data.

Student (Roll no, name, class, birthdate)

Course (course No, coursename, max. mark, Pass

SC (Roll No, Course No, Marks)

```
Create table student
(
Roll no int primary key,
name varchar(10),
class varchar(10),
Birthdate date
);

insert into student values
(1, 'Diya', 'FY', '9-Oct-2003'),
(2, 'Pankaj', 'SY', '6-April-2001'),
(3, 'Prachi', 'FY', '1-Jun-2003'),
(4, 'Jemmy', 'SY', '10-March-2002'),
(5, 'Alka', 'FY', '1-Jan-2003'),
(6, 'Kiran', 'SY', '18-Dec-2002');
```

```
Create table course
(
CourseNo int primary key,
Course name varchar(10),
Max marks int,
Pass marks int
);

insert into course values
(101, 'Maths', 100, 35),
(102, 'DBMS', 100, 35),
(103, 'CN', 100, 35),
(104, 'CPPM', 100, 35),
(105, 'OS', 100, 35),
(106, 'PS', 100, 35),
(107, 'Practical', '300, 150);

Pragma Foreign_keys=on;

Create table SC2
(
Roll No int references students (Roll No),
Course No int references Course (course No),
Marks int
);

insert into SC2 values
(1, 101, 60), (1, 102, 55),
(2, 101, 90), (2, 102, 88),
(3, 101, 40), (5, 102, 34);
```

(4, 201, 72), (4, 202, 70),
(5, 201, 55), (5, 202, 38),
(6, 201, 88), (8, 202, 74),
(7, 301, 41), (7, 302, 44),
(8, 301, 65), (8, 302, 23),
(9, 301, 50), (9, 302, 61);

1. Display details of students who takes 'DBMS' course.

→ select students, rollno, name, class, birthdate, course coursename from (students inner join SC_2 on students.rollno = SC_2 roll no) inner join course on SC_2 course no = course no where coursename = "DBMS";

2. Display the name of student who have scored more than 70% in computer Networks and have not failed in any subject.

→ select name from (students inner joint SC_2 on students.rollno - SC_2.rollno) inner joint course on SC_2 coursend = course = courseno = where course = coursename = "Computer Network" and SC_2.marks > (70 * 100) / 100 and SC_2.marks >= 35;

3. Display the average marks obtained by each student.

→ select students name, avg (SC_2.marks) as "Avg Marks" from students inner joints SC_2 on students.rollno = SC_2 rollno group by student name.

4. Select all courses where passing marks are more than 30% of average maximum marks.

→ Select coursename From course where pass_marks = (select avg(max_marks) * 30/100 From course);

5. Display all course name.

→ select coursename From course;

6. Display the student detail who have secure 1st rank in 'computer network' course.

→ select students rollno, students name, students class, students birthdate, max(sc2.marks) as "max_marks" From students inner joint sc2 on students rollno = sc2 roll no where sc2.courseno = 201;

7. Display all sy student list along with course name.

→ select distint students name, students class, st course. coursename from (students inner joint course on sc2.courseno = course.courseno) where students class = "sy" group by students name;

8. Display the average Marks obtained by each student.

→ select students name, avg(sc2.marks) as "Avg marks" From students inner joint sc2 on students rollno = sc2. roll no group by students. name;