

1. Employee

- i. CREATE TABLE command as follows
- ii. a. Primary Key is **empno**

```

1  -- q1. i. Create a Table EMPLOYEE to store employee details as shown below.
2  -- q1. ii. a Assign one column of your choice as the primary key.
3  DROP TABLE IF EXISTS Employee;
4
5  CREATE TABLE Employee(
6      empno INT NOT NULL,
7      ename VARCHAR(64) NOT NULL,
8      job VARCHAR(64) NOT NULL,
9      mgr INT,
10     hiredate DATE NOT NULL,
11     sal NUMERIC NOT NULL,
12     comm NUMERIC,
13     deptno INT NOT NULL,
14     CONSTRAINT PK_Employee PRIMARY KEY(empno)
15 );
16

```

b. I used sqlite browser for windows, PRAGMA is the command for this one. However, there are alternate ways to do this using Schema command.

```

48  -- q1.ii.b. Display the EMPLOYEE table structure.
49  PRAGMA TABLE_INFO(Employee);
50

```

	cid	name	type	notnull	dflt_value	pk
1	0	empno	INT	1	NULL	1
2	1	ename	VARCHAR(64)	1	NULL	0
3	2	job	VARCHAR(64)	1	NULL	0
4	3	mgr	INT	0	NULL	0
5	4	hiredate	DATE	1	NULL	0
6	5	sal	NUMERIC	1	NULL	0
7	6	comm	NUMERIC	0	NULL	0
8	7	deptno	INT	1	NULL	0

c. `SELECT * FROM Employee;`

```

17 -- q1. i.
18 INSERT INTO Employee (empno, ename, job, mgr, hiredate, sal, comm, deptno) VALUES
19 (8369, "Smith", "Clerk", 8902, "1990-12-18", 800.00, NULL, 20),
20 (8499, "Anya", "Salesman", 8698, "1991-02-20", 1600.00, 300.00, 30),
21 (8521, "Seth", "Salesman", 8698, "1991-02-22", 1250.00, 500.00, 30),
22 (8566, "Mahadevan", "Manager", 8839, "1991-04-02", 2985.00, NULL, 20),
23 (8654, "Momin", "Salesman", 8698, "1991-09-28", 1250.00, 1400.00, 30),
24 (8698, "Bina", "Manager", 8839, "1991-05-01", 2850.00, NULL, 30),
25 (8882, "Shivansh", "Manager", 8839, "1991-06-09", 2450.00, NULL, 10),
26 (8888, "Scott", "Analyst", 8566, "1992-12-09", 3000.00, NULL, 20),
27 (8839, "Amir", "President", NULL, "1991-11-18", 5000.00, NULL, 10),
28 (8844, "Kuldeep", "Salesman", 8698, "1991-09-08", 1500.00, 0.00, 30);
29
30 -- q1. ii. Display all the records the from EMPLOYEE table.
31 SELECT * FROM Employee;
32

```

	empno	ename	job	mgr	hiredate	sal	comm	deptno
1	8369	Smith	Clerk	8902	1990-12-18	800	NULL	20
2	8499	Anya	Salesman	8698	1991-02-20	1600	300	30
3	8521	Seth	Salesman	8698	1991-02-22	1250	500	30
4	8566	Mahadevan	Manager	8839	1991-04-02	2985	NULL	20
5	8654	Momin	Salesman	8698	1991-09-28	1250	1400	30
6	8698	Bina	Manager	8839	1991-05-01	2850	NULL	30
7	8882	Shivansh	Manager	8839	1991-06-09	2450	NULL	10
8	8888	Scott	Analyst	8566	1992-12-09	3000	NULL	20
9	8839	Amir	President	NULL	1991-11-18	5000	NULL	10
10	8844	Kuldeep	Salesman	8698	1991-09-08	1500	0	30

d. `SELECT ename, sal FROM Employee WHERE sal >= 2200;`

```

54 -- q1. ii. d. To display ename
55 SELECT ename, sal
56 FROM Employee
57 WHERE sal >= 2200;
58

```

	ename	sal
1	Mahadevan	2985
2	Bina	2850
3	Shivansh	2450
4	Scott	3000
5	Amir	5000

- e. `SELECT * FROM Employee WHERE comm IS NULL OR comm = 0;`

```

41  -- q1. ii. e. To display all details of employees who ar
42  SELECT *
43  FROM Employee
44  WHERE comm IS NULL
45  OR comm = 0;
46

```

	empno	ename	job	mgr	hiredate	sal	comm	deptno
1	8369	Smith	Clerk	8902	1990-12-18	800	NULL	20
2	8566	Mahadevan	Manager	8839	1991-04-02	2985	NULL	20
3	8698	Bina	Manager	8839	1991-05-01	2850	NULL	30
4	8882	Shivansh	Manager	8839	1991-06-09	2450	NULL	10
5	8888	Scott	Analyst	8566	1992-12-09	5000	NULL	20
6	8839	Amir	President	NULL	1991-11-18	5000	NULL	10
7	8844	Kuldeep	Salesman	8698	1991-09-08	1500	0	30

- f. `SELECT ename, sal FROM Employee WHERE sal NOT BETWEEN 2500 AND 4000;`

```

64  -- q1. ii. f. To display employee name and salary of
65  SELECT ename, sal
66  FROM Employee
67  WHERE sal NOT BETWEEN 2500 AND 4000;
68

```

	ename	sal
1	Smith	800
2	Any	1600
3	Seth	1250
4	Momin	1250
5	Shivansh	2450
6	Amir	5000
7	Kuldeep	1500

- g. `SELECT ename, job, sal FROM Employee WHERE mgr IS NULL;`

```

69  -- q1. ii. g. To display the name, job title
70  SELECT ename, job, sal FROM Employee
71  WHERE mgr IS NULL;
72

```

	ename	job	sal
1	Amir	President	5000

- h. SELECT ename FROM Employee WHERE ename LIKE "__a%";
 SELECT ename FROM Employee WHERE ename REGEXP "^..[Aa].*";

```
56  -- q1. ii. h. To display the name of employee whose
57  SELECT ename
58  FROM Employee
59  WHERE ename LIKE "__a%";
60
61  SELECT ename
62  FROM Employee
63  WHERE ename REGEXP "^..[Aa].*";
64
```

- i. SELECT ename FROM Employee WHERE ename REGEXP "[Tt]\$";
 SELECT ename FROM Employee WHERE ename LIKE "%t";

```
65  -- q1. ii. i. To display the name of employee
66  SELECT ename
67  FROM Employee
68  WHERE ename REGEXP "[Tt]$";
69
70  SELECT ename
71  FROM Employee
72  WHERE ename LIKE "%t";
73
```

	ename
1	Scott

- j. SELECT ename FROM Employee WHERE ename LIKE "m_l%";
 SELECT ename FROM Employee WHERE ename REGEXP "^[Mm].[Ll].*";

```

73
74 -- q1. ii. j. To display the name of employee whose
75 SELECT ename
76 FROM Employee
77 WHERE ename LIKE "m_l%";
78
79 SELECT ename
80 FROM Employee
81 WHERE ename REGEXP "^[Mm].[Ll].*";

```

- k. UPDATE Employee SET sal = 5000.00 WHERE ename = "Scott";

```

83 -- q1. ii. k. There has been an error, input the correct s
84 UPDATE Employee
85 SET sal = 5000.00
86 WHERE ename = "Scott";
87
88 SELECT ename, sal FROM Employee WHERE ename = "Scott";

```

	ename	sal
1	Scott	5000

- l. SELECT DISTINCT job FROM Employee;

```

90 -- q1. ii. l. Display only the types of
91 SELECT DISTINCT job FROM Employee;

```

	job
1	Clerk
2	Salesman
3	Manager
4	Analyst
5	President

2.

i.

-- Student

DROP TABLE IF EXISTS Student;

```
CREATE TABLE Student (  
    StdID INT NOT NULL,  
    Fname VARCHAR(64) NOT NULL,  
    Lname VARCHAR(64),  
    Credits INT NOT NULL,  
    Dept VARCHAR(32) NOT NULL,  
    Gender VARCHAR(1) NOT NULL,  
    CONSTRAINT PK_Student PRIMARY KEY(StdID)  
);
```

```
INSERT INTO Student (StdID, Fname, Lname, Credits, Dept, Gender) VALUES  
(100, "Mary", "Cooper", 6000, "Drama", "F"),  
(101, "Mike", "Carpen", 5000, "Maths", "M"),  
(102, "Ryan", "Smith", 10000, "Drama", "M"),  
(103, "Tom", "Randall", 4800, "Maths", "M"),  
(104, "Ashley", "Brown", 5000, "Science", "F");
```

	StdID	Fname	Lname	Credits	Dept	Gender
1	100	Mary	Cooper	6000	Drama	F
2	101	Mike	Carpen	5000	Maths	M
3	102	Ryan	Smith	10000	Drama	M
4	103	Tom	Randall	4800	Maths	M
5	104	Ashley	Brown	5000	Science	F

```
-- Project
DROP TABLE IF EXISTS Project;

CREATE TABLE Project (
    ProjectID INT NOT NULL,
    ProjectName VARCHAR(64),
    StdID INT NOT NULL,
    CONSTRAINT PK_Project PRIMARY KEY(ProjectID)
);

INSERT INTO Project (ProjectID, StdID, ProjectName) VALUES
(1, 100, "Shakespeare"),
(2, 100, "Greek Tragedy"),
(3, 100, "Disaster"),
(4, 101, "Trigonometry"),
(5, 102, "Wizard of Oz"),
(6, 102, "Creative Dramatics"),
(7, 102, "Modern Art"),
(8, 106, "Natural Language Processing"),
(9, 104, "Gravity");
```

	ProjectID	ProjectName	StdID
1	1	Shakespeare	100
2	2	Greek Tragedy	100
3	3	Disaster	100
4	4	Trigonometry	101
5	5	Wizard of Oz	102
6	6	Creative Dramatics	102
7	7	Modern Art	102
8	8	Natural Language Processing	106
9	9	Gravity	104

ii.

a.

```

SELECT Student.Fname, Student.Lname, Project.ProjectName
FROM Student JOIN Project
ON Student.StdID = Project.StdID
ORDER BY Student.Fname, Project.ProjectName;

```

```

137
138 -- q2. ii.
139 -- a. Get name, project name order by firstname from "student"
140 SELECT Student.Fname, Student.Lname, Project.ProjectName
141 FROM Student JOIN Project
142 ON Student.StdID = Project.StdID
143 ORDER BY Student.Fname, Project.ProjectName;

```

	Fname	Lname	ProjectName
1	Ashley	Brown	Gravity
2	Mary	Cooper	Disaster
3	Mary	Cooper	Greek Tragedy
4	Mary	Cooper	Shakespeare
5	Mike	Carpen	Trigonometry
6	Ryan	Smith	Creative Dramatics
7	Ryan	Smith	Modern Art
8	Ryan	Smith	Wizard of Oz

b.

```

SELECT Student.Fname, Student.Lname, Project.ProjectName
FROM Student LEFT JOIN Project
ON Student.StdID = Project.StdID
WHERE ProjectName IS NULL
ORDER BY Student.Lname;

```

```

145 -- b. Get name, project name order by lastname from "student"
146 SELECT Student.Fname, Student.Lname, Project.ProjectName
147 FROM Student LEFT JOIN Project
148 ON Student.StdID = Project.StdID
149 WHERE ProjectName IS NULL
150 ORDER BY Student.Lname;
151

```

	Fname	Lname	ProjectName
1	Tom	Randall	NULL

- c.
 SELECT Project.ProjectName
 FROM Project LEFT JOIN Student
 ON Student.StdID = Project.StdID
 ORDER BY Student.Fname;

```

152  -- c. Get all project name even if they do not have
153  SELECT Project.ProjectName
154  FROM Project LEFT JOIN Student
155  ON Student.StdID = Project.StdID
156  ORDER BY Student.Fname;
157

```

	ProjectName
1	Natural Language Processing
2	Gravity
3	Shakespeare
4	Greek Tragedy
5	Disaster
6	Trigonometry
7	Wizard of Oz
8	Creative Dramatics
9	Modern Art

- d.
 SELECT Student.Fname, Student.Lname, Student.Credits, Student.Dept,
 Student.Gender, Project.ProjectName
 FROM Project LEFT JOIN Student ON Student.StdID = Project.StdID
 UNION
 SELECT Student.Fname, Student.Lname, Student.Credits, Student.Dept,
 Student.Gender, Project.ProjectName
 FROM Student LEFT JOIN Project ON Student.StdID = Project.StdID;

```

158  -- d. Get complete records from both tables
159  SELECT Student.Fname, Student.Lname, Student.Credits, Student.Dept, Student.Gender, Project.ProjectName
160  FROM Project LEFT JOIN Student
161  ON Student.StdID = Project.StdID
162  UNION
163  SELECT Student.Fname, Student.Lname, Student.Credits, Student.Dept, Student.Gender, Project.ProjectName
164  FROM Student LEFT JOIN Project
165  ON Student.StdID = Project.StdID
166  ;
167

```

	Fname	Lname	Credits	Dept	Gender	ProjectName
1	NULL	NULL	NULL	NULL	NULL	Natural Language Processing
2	Ashley	Brown	5000	Science	F	Gravity
3	Mary	Cooper	6000	Drama	F	Disaster
4	Mary	Cooper	6000	Drama	F	Greek Tragedy
5	Mary	Cooper	6000	Drama	F	Shakespeare
6	Mike	Carpenter	5000	Maths	M	Trigonometry
7	Ryan	Smith	10000	Drama	M	Creative Dramatics
8	Ryan	Smith	10000	Drama	M	Modern Art
9	Ryan	Smith	10000	Drama	M	Wizard of Oz
10	Tom	Randall	4800	Maths	M	NULL

3.

- Student (ID, NAME, SUBJECT, AVERAGE, DIV, CREDITS)
- Teacher (SUBJECT, PNAME)

a. Display the student, professor name and the subject.

```
SELECT Student.NAME, Student.SUBJECT, Teacher.PNAME
FROM Student JOIN Teacher
ON Student.SUBJECT = Teacher.SUBJECT;
```

b. Display all the student and professor name who are offering subjects Maths and Science.

```
SELECT Student.NAME, Teacher.PNAME
FROM Teacher LEFT JOIN Student
ON Teacher.SUBJECT = Student.SUBJECT
WHERE Teacher.SUBJECT IN ("Maths", "Science");
```

c. Correct the following query if you find any errors.

```
Select NAME from Student where CREDITS = NULL; (NULL All Uppercase)
Select NAME, CREDITS From Student where CREDITS BETWEEN 10 AND 20;
(No error)
```