

```
SQL>
SQL> CREATE TABLE Student(RollNo varchar2(8), SName varchar2(20),
  2  Gender varchar2(1), City varchar2(20), PRIMARY KEY(RollNo));
```

Table created.

```
SQL>
SQL> INSERT INTO Student VALUES('1901CS99', 'Rahul', 'M', 'Kolkata');
```

1 row created.

```
SQL> INSERT INTO Student VALUES('1901CS98', 'Rajat', 'M', 'Bengalore');
```

1 row created.

```
SQL> INSERT INTO Student VALUES('1901CS97', 'Srishti', 'F', 'Chennai');
```

1 row created.

```
SQL> INSERT INTO Student VALUES('1901CS96', 'Ramesh', 'M', 'Hyderabad');
```

1 row created.

```
SQL> INSERT INTO Student VALUES('1901EC95', 'Rahul', 'M', 'Chennai');
```

1 row created.

```
SQL> INSERT INTO Student VALUES('1901CS94', 'Anyu', 'F', 'Delhi');
```

1 row created.

```
SQL> INSERT INTO Student VALUES('1901CS93', 'Thanya', 'F', 'Delhi');
```

```
1 row created.
```

```
SQL>
```

```
SQL> CREATE TABLE Course(CCode varchar2(5), CName varchar2(20), Credit number, PRIMARY KEY(CCode));
```

```
Table created.
```

```
SQL>
```

```
SQL> INSERT INTO Course VALUES('CS354', 'Database System', 4);
```

```
1 row created.
```

```
SQL> INSERT INTO Course VALUES('ME305', 'Eng. Mechanics', 3);
```

```
1 row created.
```

```
SQL> INSERT INTO Course VALUES('CS535', 'Operating System', 2);
```

```
1 row created.
```

```
SQL> INSERT INTO Course VALUES('CS356', 'Computer Networks', 3);
```

```
1 row created.
```

```
SQL> INSERT INTO Course VALUES('CS537', 'Compiler Design', 4);
```

```
1 row created.
```

```
SQL> INSERT INTO Course VALUES('CS538', 'Data Mining', 3);
```

```
1 row created.
```

```
SQL>
```

```
SQL> CREATE TABLE Enrolled(RollNo varchar2(8), CCode varchar2(5), YoE number);
```

```
Table created.
```

```
SQL>
```

```
SQL> INSERT INTO Enrolled VALUES('1901CS93', 'CS354', 2019);
```

```
1 row created.
```

```
SQL> INSERT INTO Enrolled VALUES('1901CS94', 'CS354', 2020);
```

```
1 row created.
```

```
SQL> INSERT INTO Enrolled VALUES('1901CS97', 'CS354', 2020);
```

```
1 row created.
```

```
SQL> INSERT INTO Enrolled VALUES('1901CS97', 'CS356', 2020);
```

```
1 row created.
```

```
SQL> INSERT INTO Enrolled VALUES('1901CS97', 'CS537', 2020);
```

```
1 row created.
```

```
SQL> INSERT INTO Enrolled VALUES('1901CS97', 'ME305', 2020);
```

```
1 row created.
```

```
SQL> INSERT INTO Enrolled VALUES('1901CS94', 'CS535', 2019);
```

```
1 row created.
```

```
SQL> INSERT INTO Enrolled VALUES('1901CS97', 'CS535', 2020);
```

```
1 row created.
```

```
SQL> INSERT INTO Enrolled VALUES('1901CS99', 'CS535', 2021);
```

```
1 row created.
```

```
SQL> INSERT INTO Enrolled VALUES('1901CS96', 'CS537', 2019);
```

```
1 row created.
```

```
SQL> INSERT INTO Enrolled VALUES('1901CS96', 'CS354', 2019);
```

```
1 row created.
```

```
SQL> INSERT INTO Enrolled VALUES('1901CS96', 'ME305', 2019);
```

```
1 row created.
```

```
SQL> INSERT INTO Enrolled VALUES('1901CS96', 'CS535', 2019);
```

1 row created.

```
SQL> INSERT INTO Enrolled VALUES('1901CS96', 'CS538', 2019);
```

1 row created.

```
SQL> INSERT INTO Enrolled VALUES('1901CS96', 'CS356', 2019);
```

1 row created.

```
SQL>
```

```
SQL> ALTER TABLE Enrolled ADD CONSTRAINT FK_Enrolled_Student FOREIGN KEY (RollNo) REFERENCES Student(RollNo);
```

Table altered.

```
SQL> ALTER TABLE Enrolled ADD CONSTRAINT FK_Enrolled_Course FOREIGN KEY (CCode) REFERENCES Course(CCode);
```

Table altered.

```
SQL>
```

```
SQL> -- Question A
```

```
SQL> ALTER TABLE Course ADD CType varchar2(20);
```

Table altered.

```
SQL> ALTER TABLE Student ADD Email varchar2(30);
```

Table altered.

```
SQL> ALTER TABLE Student ADD CONSTRAINT UC_Email UNIQUE (Email);
```

```
Table altered.
```

```
SQL>
```

```
SQL> UPDATE Course SET CType = 'Core' WHERE CCode = 'CS354';
```

```
1 row updated.
```

```
SQL> UPDATE Course SET CType = 'Elective' WHERE CCode = 'ME305';
```

```
1 row updated.
```

```
SQL> UPDATE Course SET CType = 'Open Elective' WHERE CCode = 'CS535';
```

```
1 row updated.
```

```
SQL> UPDATE Course SET CType = 'Elective' WHERE CCode = 'CS356';
```

```
1 row updated.
```

```
SQL> UPDATE Course SET CType = 'Open Elective' WHERE CCode = 'CS537';
```

```
1 row updated.
```

```
SQL> UPDATE Course SET CType = 'Open Elective' WHERE CCode = 'CS538';
```

```
1 row updated.
```

```
SQL>
```

```
SQL> UPDATE Student SET Email = 'rahul@gmail.com' WHERE RollNo = '1901CS99';
```

```
1 row updated.
```

```
SQL> UPDATE Student SET Email = 'rajat@yahoo.com' WHERE RollNo = '1901CS98';
```

```
1 row updated.
```

```
SQL> UPDATE Student SET Email = 'srishti@outlook.com' WHERE RollNo = '1901CS97';
```

```
1 row updated.
```

```
SQL> UPDATE Student SET Email = 'ramesh@proton.me' WHERE RollNo = '1901CS96';
```

```
1 row updated.
```

```
SQL> UPDATE Student SET Email = 'rahul@yahoo.com' WHERE RollNo = '1901EC95';
```

```
1 row updated.
```

```
SQL> UPDATE Student SET Email = 'anya@gmail.com' WHERE RollNo = '1901CS94';
```

```
1 row updated.
```

```
SQL> UPDATE Student SET Email = 'thanya@outlook.com' WHERE RollNo = '1901CS93';
```

```
1 row updated.
```

```
SQL>
```

```
SQL> -- Question B
```

```
SQL> CREATE VIEW CSCourse AS
  2  SELECT CName FROM Course WHERE CCode LIKE 'CS%';
```

View created.

```
SQL> SELECT * FROM CSCourse;
```

```
CNAME
-----
Database System
Computer Networks
Operating System
Compiler Design
Data Mining
```

```
SQL>
SQL> -- Question C
SQL> CREATE VIEW MaleAASStudent AS
  2  SELECT SName FROM Student WHERE Gender = 'M' AND SName LIKE '%a%a%';
```

View created.

```
SQL> SELECT * FROM MaleAASStudent;
```

```
SNAME
-----
Rajat
```

```
SQL>
SQL> -- Question D
```



```
SQL> CREATE VIEW EnrolledNot1819 AS
  2  SELECT RollNo FROM Enrolled WHERE YoE NOT BETWEEN 2018 AND 2019;
```

View created.

```
SQL> SELECT * FROM EnrolledNot1819;
```

```
ROLLNO
-----
1901CS94
1901CS97
1901CS97
1901CS97
1901CS97
1901CS97
1901CS97
1901CS99
```

7 rows selected.

```
SQL>
SQL> -- Question E
SQL> CREATE VIEW MinCreditCourse AS
  2  SELECT * FROM Course WHERE Credit = (SELECT MIN(Credit) FROM Course);
```

View created.

```
SQL> SELECT * FROM MinCreditCourse;
```

```
CCODE CNAME                CREDIT CTYPE
-----
-----
```

SQL>

SQL> -- Question F

SQL> CREATE VIEW CourseNotEnrolled2020 AS

2 SELECT CName FROM Course WHERE CCode NOT IN (SELECT CCode FROM Enrolled WHERE YoE = 2020);

View created.

SQL> SELECT \* FROM CourseNotEnrolled2020;

CNAME

-----

Data Mining

SQL>

SQL> -- Question G

SQL> CREATE VIEW StudentCountCity AS

2 SELECT City, COUNT(\*) AS CityCount FROM Student GROUP BY City;

View created.

SQL> SELECT \* FROM StudentCountCity;

CITY

CITYCOUNT

-----

Bengalore 1

Delhi 2

Kolkata 1

Hyderabad 1

Chennai

2

SQL>

SQL> -- Question H

SQL> CREATE VIEW StudentCoursesCount5 AS

2 SELECT RollNo FROM Enrolled GROUP BY RollNo HAVING COUNT(CCode) = 5;

View created.

SQL> SELECT \* FROM StudentCoursesCount5;

ROLLNO

-----

1901CS97

SQL>

SQL> -- Question I

SQL> CREATE VIEW CourseCountCS99 AS

2 SELECT COUNT(CCode) AS CourseCount FROM Enrolled WHERE RollNo = '1901CS99';

View created.

SQL> SELECT \* FROM CourseCountCS99;

COURSECOUNT

-----

1

SQL>

SQL> --Question J

```
SQL> CREATE VIEW Level5Course AS
  2  SELECT * FROM Course WHERE CCode LIKE '__5%';
```

View created.

```
SQL> SELECT * FROM Level5Course;
```

CCODE	CNAME	CREDIT	CTYPE
CS535	Operating System	2	Open Elective
CS537	Compiler Design	4	Open Elective
CS538	Data Mining	3	Open Elective

```
SQL>
```

```
SQL> -- Question K
```

```
SQL> CREATE VIEW StudentCountCS AS
  2  SELECT COUNT(RollNo) AS StudCount FROM Student WHERE RollNo LIKE '%CS%';
```

View created.

```
SQL> SELECT * FROM StudentCountCS;
```

STUDCOUNT
6

```
SQL>
```

```
SQL> -- Question L
```

```
SQL> CREATE VIEW Enrolled2020MoreThan4 AS
  2  SELECT RollNo FROM Enrolled WHERE YoE = 2020 GROUP BY RollNo HAVING COUNT(CCode) > 4;
```

View created.

```
SQL> SELECT * FROM Enrolled2020MoreThan4;
```

```
ROLLNO  
-----  
1901CS97
```

```
SQL>
```

```
SQL> -- Question M
```

```
SQL> CREATE VIEW CourseEnrolledMaxFemaleDelhi AS  
 2 WITH FemaleDelhiStudents AS (  
 3     SELECT RollNo FROM Student WHERE Gender = 'F' AND City = 'Delhi'  
 4 ),  
 5 MaxEnrollment AS (  
 6     SELECT MAX(COUNT(*)) AS MaxCount  
 7     FROM Enrolled  
 8     WHERE RollNo IN (SELECT RollNo FROM FemaleDelhiStudents)  
 9     GROUP BY CCode  
10 )  
11 SELECT CCode  
12 FROM Enrolled  
13 WHERE RollNo IN (SELECT RollNo FROM FemaleDelhiStudents)  
14 GROUP BY CCode  
15 HAVING COUNT(*) = (SELECT MaxCount FROM MaxEnrollment);
```

View created.

```
SQL> SELECT * FROM CourseEnrolledMaxFemaleDelhi;
```

CCODE

-----

CS354

SQL>

SQL> -- Question N

SQL> CREATE VIEW StudentGmailYahoo AS

2 SELECT SName FROM Student WHERE Email LIKE '%@gmail.com' OR Email LIKE '%@yahoo.com';

View created.

SQL> SELECT \* FROM StudentGmailYahoo;

SNAME

-----

Rahul

Rajat

Rahul

Anyu

SQL>

SQL> -- Question 0

SQL> CREATE VIEW StudentEnrolledAllCourse AS

2 SELECT SName FROM Student WHERE RollNo IN (  
3       SELECT RollNo FROM Enrolled GROUP BY RollNo  
4       HAVING COUNT(CCode) = (SELECT COUNT(\*) FROM Course)  
5 );

View created.

```
SQL> SELECT * FROM StudentEnrolledAllCourse;
```

```
SNAME
```

```
-----
```

```
Ramesh
```

```
SQL>
```

```
SQL> |
```