﻿CREATE TABLE STUDENT

(

ROLLNO INT PRIMARY KEY,

NAME VARCHAR2(20) NOT NULL,

CLASS CHAR(2) CHECK(CLASS='FY' OR CLASS='SY' OR CLASS='TY'),

BIRTHDATE DATE

);

INSERT INTO STUDENT

VALUES(1,'OM','TY','22/03/2002');

INSERT INTO STUDENT

VALUES(2,'SAI','TY','18/09/2002');

INSERT INTO STUDENT

VALUES(3,'SHAHIL','TY','01/12/2002');

INSERT INTO STUDENT

VALUES(4,'RAM','SY','11/09/2003');

INSERT INTO STUDENT

VALUES(5,'KRISHNA','SY','15/04/2005');

INSERT INTO STUDENT

VALUES(6,'RADHA','SY','22/11/2003');

INSERT INTO STUDENT

VALUES(7,'JAY','SY','09/03/2002');

INSERT INTO STUDENT

VALUES(8,'SHIV','FY','10/12/2004');

INSERT INTO STUDENT

VALUES(9,'DEV','FY','12/10/2004');

INSERT INTO STUDENT

VALUES(10,'BRIJESH','FY','22/03/2004');

CREATE TABLE COURSE

(

COURSENO INT PRIMARY KEY,

COURSENAME VARCHAR2(30) NOT NULL,

MAX\_MARKS INT DEFAULT(100),

PASS\_MARKS INT DEFAULT(35)

);

INSERT INTO COURSE(COURSENO,COURSENAME)

VALUES(101,'DATABASE MANAGEMENT SYSTEM');

INSERT INTO COURSE(COURSENO,COURSENAME)

VALUES(102,'COMPUTER NETWORK');

INSERT INTO COURSE(COURSENO,COURSENAME)

VALUES(103,'DATA SCIENCE');

INSERT INTO COURSE(COURSENO,COURSENAME)

VALUES(104,'MACHINE LEARNING');

INSERT INTO COURSE(COURSENO,COURSENAME)

VALUES(105,'DIGITAL MARKETER');

INSERT INTO COURSE(COURSENO,COURSENAME)

VALUES(106,'AI ENGINEER');

INSERT INTO COURSE(COURSENO,COURSENAME)

VALUES(107,'SOFTWARE DEVELOPER');

INSERT INTO COURSE(COURSENO,COURSENAME)

VALUES(108,'BIG DATA');

INSERT INTO COURSE(COURSENO,COURSENAME)

VALUES(109,'CYBER SECURITY');

INSERT INTO COURSE(COURSENO,COURSENAME)

VALUES(11,'GRAPHIC DESIGNER');

CREATE TABLE SC

(

ROLLNO INT,

COURSENO INT,

MARKS INT,

PRIMARY KEY(ROLLNO,COURSENO),

FOREIGN KEY(ROLLNO) REFERENCES STUDENT(ROLLNO),

FOREIGN KEY(COURSENO) REFERENCES COURSE(COURSENO)

);

INSERT INTO SC

VALUES(1,101,90);

INSERT INTO SC

VALUES(2,101,93);

INSERT INTO SC

VALUES(3,101,60);

INSERT INTO SC

VALUES(4,101,30);

INSERT INTO SC

VALUES(10,102,60);

INSERT INTO SC

VALUES(10,103,80);

INSERT INTO SC

VALUES(5,102,90);

INSERT INTO SC

VALUES(7,102,90);

INSERT INTO SC

VALUES(2,103,90);

INSERT INTO SC

VALUES(8,104,90);

INSERT INTO SC

VALUES(5,105,90);

INSERT INTO SC

VALUES(9,105,90);

INSERT INTO SC

VALUES(10,106,90);

INSERT INTO SC

VALUES(2,107,90);

INSERT INTO SC

VALUES(3,106,90);

INSERT INTO SC

VALUES(7,107,90);

INSERT INTO SC

VALUES(4,109,90);

INSERT INTO SC

VALUES(9,102,30);

INSERT INTO SC

VALUES(5,101,30);

INSERT INTO SC

VALUES(5,109,110);

QUERISE:

1. Display details of student who takes ‘Database Management System’ course.

SELECT S.\* FROM STUDENT S,SC M

WHERE S.ROLLNO=M.ROLLNO AND M.COURSENO=(SELECT COURSENO FROM COURSE WHERE COURSENAME='DATABASE MANAGEMENT SYSTEM');

2. Display the names of students who have scored more than 70% in Computer

Networks and have not failed in any subject.

SELECT \*,M.MARKS=(M.MARKS\*100/100) FROM STUDENT S,SC M

WHERE S.ROLLNO=M.ROLLNO AND M.MARKS>70

AND M.COURSENO=(SELECT COURSENO FROM COURSE WHERE COURSENAME='COMPUTER NETWORK');

SELECT \*,M.MARKS=(M.MARKS\*100/100) FROM STUDENT S,SC M

WHERE S.ROLLNO=M.ROLLNO AND M.MARKS>70 AND MARKS>35

AND M.COURSENO=(SELECT COURSENO FROM COURSE WHERE COURSENAME='COMPUTER NETWORK');

SELECT S.\*,C.\*,M.\* FROM STUDENT S,COURSE C, SC M

WHERE M.MARKS>70 AND C.COURSENAME="COMPUTER NETWORK" AND M.ROLLNO IN(SELECT ROLLNO FROM SC WHERE ROLLNO NOT IN (SELECT ROLLNO FROM SC,COURSE WHERE MARKS<C.PASS\_MARKS))

AND S.ROLLNO=M.ROLLNO;

SELECT S.\*,C.\*,M.\* FROM STUDENT S,COURSE C, SC M

WHERE M.MARKS>70 AND C.COURSENAME="COMPUTER NETWORK" AND M.ROLLNO IN(SELECT ROLLNO FROM SC WHERE ROLLNO NOT IN (SELECT ROLLNO FROM SC,COURSE WHERE MARKS<C.PASS\_MARKS))

AND S.ROLLNO=M.ROLLNO AND C.COURSENO=M.COURSENO;

SELECT S.\*,C.\*,M.\*,M.MARKS=(M.MARKS\*100/100) FROM STUDENT S,COURSE C, SC M

WHERE M.MARKS>70 AND C.COURSENAME="COMPUTER NETWORK" AND M.ROLLNO IN(SELECT ROLLNO FROM SC WHERE ROLLNO NOT IN (SELECT ROLLNO FROM SC,COURSE WHERE MARKS<C.PASS\_MARKS))

AND S.ROLLNO=M.ROLLNO AND C.COURSENO=M.COURSENO;

SELECT S.NAME,M.MARKS=(M.MARKS\*100/100) FROM STUDENT S,COURSE C, SC M

WHERE M.MARKS>70 AND C.COURSENAME="COMPUTER NETWORK" AND M.ROLLNO IN(SELECT ROLLNO FROM SC WHERE ROLLNO NOT IN (SELECT ROLLNO FROM SC,COURSE WHERE MARKS<C.PASS\_MARKS))

AND S.ROLLNO=M.ROLLNO AND C.COURSENO=M.COURSENO;

3. Display the average marks obtained by each student.

SELECT ROLLNO,AVG(MARKS) "AVERAGE MARKS SCORED BY STUDENT" FROM SC

GROUP BY ROLLNO;

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

mark.

SELECT \*,AVG(PASS\_MARKS) FROM COURSE

GROUP BY COURSENAME

HAVING PASS\_MARKS>30;

5. Display all course name.

SELECT COURSENAME FROM COURSE;

6. Display the student details who have secure 1st rank in ‘Computer Network’

course.

SELECT \*,MAX(MARKS) FROM SC M,STUDENT S

WHERE S.ROLLNO=M.ROLLNO AND M.COURSENO=(SELECT COURSENO FROM COURSE WHERE COURSENAME='COMPUTER NETWORK');

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

SELECT S.\*,C.COURSENAME FROM STUDENT S,SC M,COURSE C

WHERE S.ROLLNO=M.ROLLNO AND C.COURSENO=M.COURSENO AND S.CLASS='SY';

8. Display the average marks obtained by each student.

SELECT \*,AVG(MARKS) FROM SC

GROUP BY ROLLNO;

9. Write a trigger which does not allow deletion of student whose pass\_mark is

greater than 35.

CREATE TRIGGER DEL\_TRG

BEFORE DELETE ON STUDENT

BEGIN

SELECT CASE

WHEN (SELECT S.\*,C.\*,M.\* FROM STUDENT S, COURSE C,SC M WHERE S.ROLLNO=M.ROLLNO AND C.COURSENO=M.COURSENO AND C.PASS\_MARKS>35)

THEN RAISE(ABORT,'WHEN THE SUBJECT HAVE PASSING MARKS MORE THEN 35 ')

END;

END;

10. Write a trigger which does not allow insertion / updating student whose max marks more than 100 and less than 0

CREATE TRIGGER INSR\_TRG

BEFORE INSERT ON SC

BEGIN

SELECT

CASE

WHEN NEW.MARKS>100 OR NEW.MARKS<0

THEN RAISE (ABORT,'PLEASE ENTER MARKS BETWEEEN 0 TO 100')

END;

END;

CREATE TRIGGER UPD\_TRG

BEFORE UPDATE ON SC

BEGIN

SELECT

CASE

WHEN NEW.MARKS>100 OR NEW.MARKS<0

THEN RAISE (ABORT,'PLEASE ENTER MARKS BETWEEEN 0 TO 100')

END;

END: