SET 3

STUDENT (rollno, name, class, birthdate)

COURSE (courseno, coursename, max_marks, pass_marks)

SC (rollno, courseno, marks)

---

1. **Create the above three tables along with key constraints.**
2. **Write an Insert script for insertion of rows with substitution variables and insert appropriate data.**

```
create table student(
                rollno int(3) primary key ,
                sname varchar(45),
                class varchar(23),
                birthday date
            );
insert into student values(5,'hjh','9 std','2018-12-9');
```

	Field	Type	Null	Key	Default	Extra
▶	rollno	int(3)	NO	PRI	NULL	
	sname	varchar(45)	YES		NULL	
	class	varchar(23)	YES		NULL	
	birthday	date	YES		NULL	

```
create table course(
                    courseno int(3) primary key,
                    cname varchar(45),
                    maxmarks int(3),
                    minmarks int(3)
                );
insert into course values(1,'ala',99,54);
```

	Field	Type	Null	Key	Default	Extra
▶	courseno	int(3)	NO	PRI	NULL	
	cname	varchar(45)	YES		NULL	
	maxmarks	int(3)	YES		NULL	
	minmarks	int(3)	YES		NULL	

```
create table sc(
                rollno int(3) ,
                courseno int(3),
                marks int(5),
                foreign key(rollno) references student(rollno)
            );
```

	Field	Type	Null	Key	Default	Extra
▶	rollno	int(3)	YES	MUL	NULL	
	courseno	int(3)	YES		NULL	
	marks	int(5)	YES		NULL	

3. **Add a constraint that the marks entered should strictly be between 0 and 100.**

```
ALTER TABLE sc ADD CHECK (marks>0 or marks<=100);
```

---

4. **While creating SC table, composite key constraint was forgotten. Add the composite key now.**

```
ALTER TABLE sc ADD FOREIGN KEY (courseno) REFERENCES course(courseno);
```

	Field	Type	Null	Key	Default	Extra
▶	rollno	int(3)	YES	MUL	NULL	
	courseno	int(3)	YES	MUL	NULL	
	marks	int(5)	YES		NULL	

---

5. **Display details of student who takes 'Database Management System' course.**

```
SELECT *
FROM student st, sc s
WHERE courseno = (
SELECT courseno
FROM course
WHERE coursename = 'dbms' )
LIMIT 0 , 30
```

rollno	name	class	birthdate	rollno	courseno	marks
101	paresh	sem3	1995-09-06	101	202	85
102	kalpesh	sem3	1995-09-07	101	202	85
101	paresh	sem3	1995-09-06	102	202	85
102	kalpesh	sem3	1995-09-07	102	202	85

---

6. **Display the names of students who have scored more than 70% in Computer Networks and have not failed in any subject.**

```
1 •  SELECT * FROM sc, student WHERE marks>70 order by sname;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

	rollno	courseno	marks	rollno	sname	class	birthday

---

7. **Display the average marks obtained by each student.**

```
1 •  SELECT * FROM sc, student where marks>30 order by sname;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

	rollno	courseno	marks	rollno	sname	class	birthday

---

8. **Select all courses where passing marks are more than 30% of average maximum mark.**

```
select * from course where maxmarks>30;
```

Result Grid | Filter Rows: | Edit:

courseno	cname	maxmarks	minmarks
1	ala	99	54
NULL	NULL	NULL	NULL

9. **Display details of students who are born in 1980 or 1982. 10. Create a view that displays student course no and its corresponding marks.**

```
SELECT course.courseno,sc.marks FROM course RIGHT JOIN sc ON course.courseno = sc.courseno ORDER BY course.courseno;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

courseno	marks

```
SELECT * FROM student where birthday between '2017-9-9' and '2018-9-8';
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Ce

rollno	sname	class	birthday
NULL	NULL	NULL	NULL

---

10. **Create a view that displays student course no and its corresponding marks.**