DBMS Lab

Name: **Soham Das**

Section: **A1**

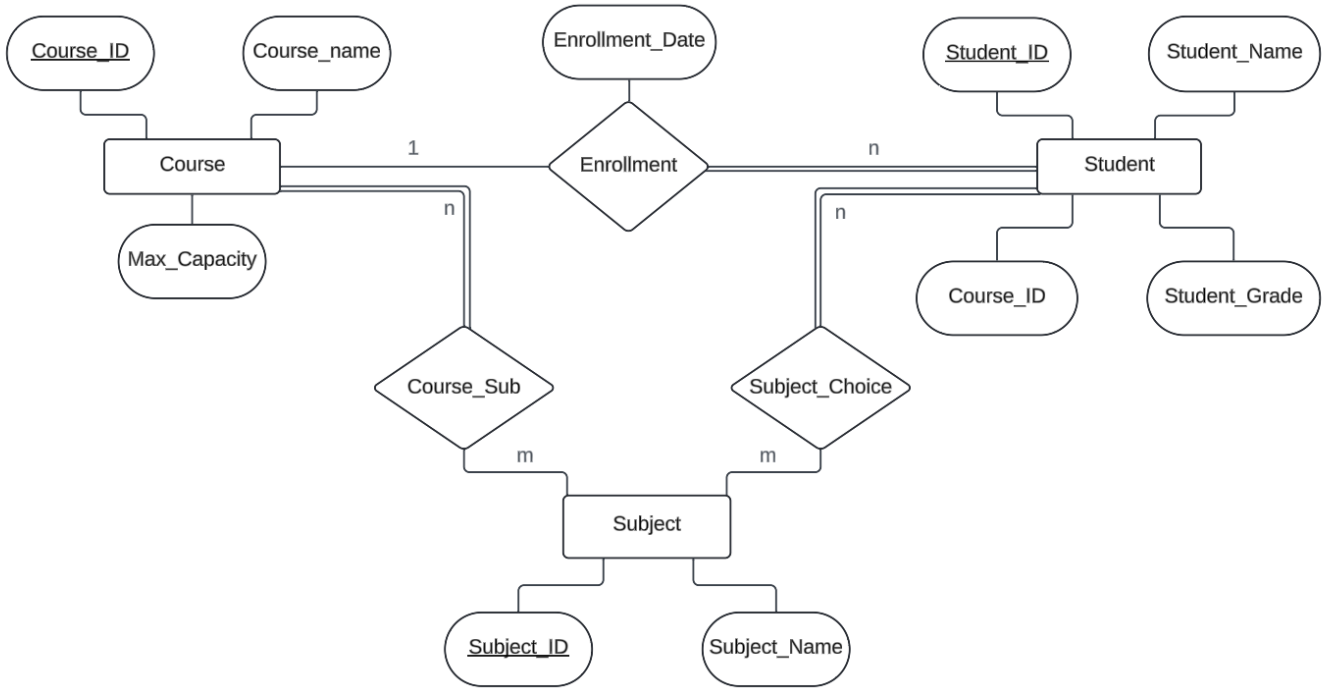
Roll No: **002311001004**

Assignment - 3

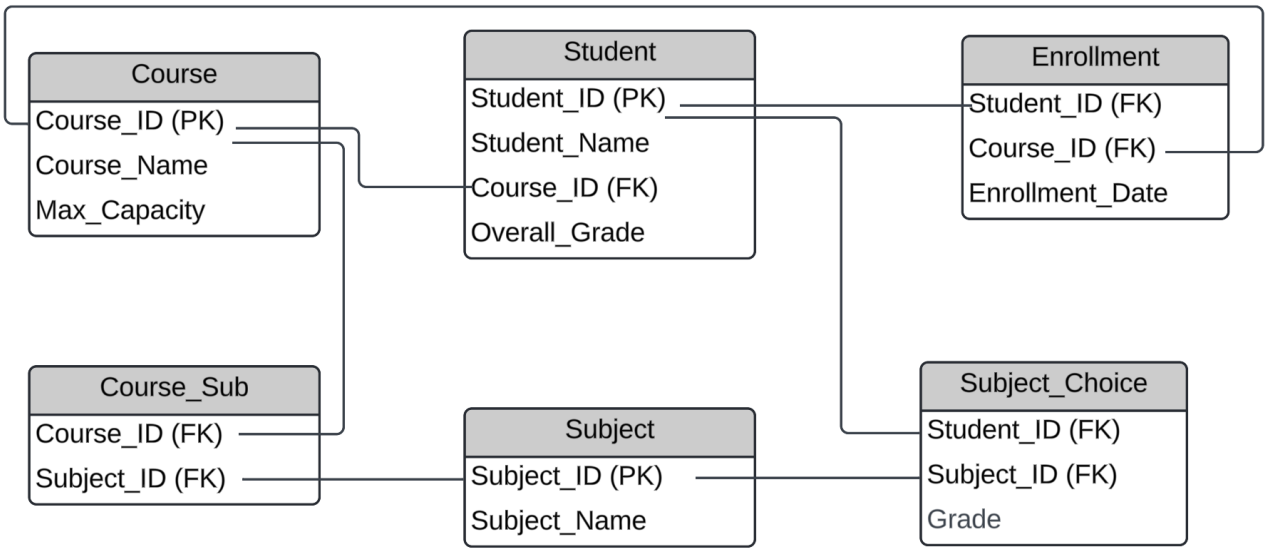
IT-UG2

Question: In an educational institute, various numbers of courses are offered. In each course, 7 numbers of subjects are taught. One student can select minimum 5 and maximum 6 numbers of subjects for that course. Each course has maximum intake capacity. The same subject may be taught in various courses. The system must be able to handle course, subject, student, marks grade and enrollment information. Assumptions also can be made. Design an ER diagram and database schema for the system. Specify the primary key, foreign key and other constraints for all required tables. Draw the ER diagram in MS Word.

ER Diagram



Database Schema



1. Insert at least five tuples in each table.

```
-- COURSE Table
Create Table COURSE (COURSE_ID number (2) PRIMARY KEY, COURSE_NAME
varchar2(10), MAX_CAPACITY number (3));

Insert into COURSE values (10, 'CS', 100);
Insert into COURSE values (20, 'IT', 90);
Insert into COURSE values (30, 'ECE', 125);
Insert into COURSE values (40, 'EE', 85);
Insert into COURSE values (50, 'IEE', 90);


--SUBJECT Table
Create Table SUBJECT (
SUBJECT_ID number (3) PRIMARY KEY, SUBJECT_NAME varchar2(15));

Insert into SUBJECT values (101, 'DBMS');
Insert into SUBJECT values (102, 'OOP');
Insert into SUBJECT values (103, 'DSA');
Insert into SUBJECT values (104, 'MATHS');
Insert into SUBJECT values (105, 'ELECTRONICS');
Insert into SUBJECT values (106, 'COA');
Insert into SUBJECT values (107, 'PHYSICS');
Insert into SUBJECT values (108, 'CHEMISTRY');

--STUDENT Table
Create Table STUDENT (
STUDENT_ID number (5) PRIMARY KEY, STUDENT_NAME varchar2(25), COURSE_ID
number (2), OVERALL_GRADE varchar2(1), FOREIGN KEY(COURSE_ID) REFERENCES
COURSE(COURSE_ID) ON DELETE CASCADE);
Insert into STUDENT values (18050, 'DEBODIT', 10, 'B');
Insert into STUDENT values (18051, 'SOHAM', 20, 'A');
Insert into STUDENT values (18052, 'ANUSKA', 30, 'C');
Insert into STUDENT values (18053, 'PRAMA', 20, 'B');
Insert into STUDENT values (18054, 'RISHIKA', 10, 'A');
Insert into STUDENT values (18055, 'ANISH', 40, 'A');
Insert into STUDENT values (18056, 'DANIAL', 20, 'C');
Insert into STUDENT values (18057, 'PRAGYA', 30, 'B');
Insert into STUDENT values (18058, 'SARBO', 20, 'A');
Insert into STUDENT values (18069, 'AKSHAT', 20, 'A');

-- ENROLLMENT Table
Create Table ENROLLMENT (
STUDENT_ID number (5) , COURSE_ID number (2), FOREIGN KEY(STUDENT_ID)
REFERENCES STUDENT(STUDENT_ID) ON DELETE CASCADE, FOREIGN
KEY(COURSE_ID) REFERENCES COURSE(COURSE_ID) ON DELETE CASCADE);
```

```
Insert into ENROLLMENT values (18050, 10);
Insert into ENROLLMENT values (18051, 20);
Insert into ENROLLMENT values (18052, 30);
Insert into ENROLLMENT values (18053, 20);
Insert into ENROLLMENT values (18054, 10);
Insert into ENROLLMENT values (18055, 40);
Insert into ENROLLMENT values (18056, 20);
Insert into ENROLLMENT values (18057, 30);
Insert into ENROLLMENT values (18058, 20);
Insert into ENROLLMENT values (18069, 20);

-- COURSE_SUB Table
Create Table COURSE_SUB (COURSE_ID number (2), SUBJECT_ID number (3), FOREIGN
KEY (COURSE_ID) REFERENCES COURSE(COURSE_ID) ON DELETE CASCADE,
FOREIGN KEY (SUBJECT_ID) REFERENCES SUBJECT(SUBJECT_ID) ON DELETE
CASCADE);
Insert into COURSE_SUB values (20, 101);
Insert into COURSE_SUB values (20, 102);
Insert into COURSE_SUB values (20, 103);
Insert into COURSE_SUB values (20, 104);
Insert into COURSE_SUB values (20, 105);
Insert into COURSE_SUB values (20, 106);
Insert into COURSE_SUB values (20, 107);
Insert into COURSE_SUB values (30, 102);
Insert into COURSE_SUB values (30, 103);
Insert into COURSE_SUB values (30, 104);
Insert into COURSE_SUB values (30, 105);
Insert into COURSE_SUB values (30, 106);
Insert into COURSE_SUB values (30, 107);
Insert into COURSE_SUB values (30, 108);

-- SUB_CHOICE Table
Create Table SUB_CHOICE (STUDENT_ID number (5), SUBJECT_ID number (3), GRADE
varchar2(1),FOREIGN KEY (STUDENT_ID) REFERENCES STUDENT(STUDENT_ID) ON
DELETE CASCADE,FOREIGN KEY (SUBJECT_ID) REFERENCES
SUBJECT(SUBJECT_ID) ON DELETE CASCADE);
Insert into SUB_CHOICE values (18053, 101, 'A');
Insert into SUB_CHOICE values (18053, 102, 'B');
Insert into SUB_CHOICE values (18053, 103, 'A');
Insert into SUB_CHOICE values (18053, 104, 'A');
Insert into SUB_CHOICE values (18053, 105, 'A');
Insert into SUB_CHOICE values (18053, 106, 'A');
Insert into SUB_CHOICE values (18055, 104, 'C');
Insert into SUB_CHOICE values (18055, 105, 'B');
Insert into SUB_CHOICE values (18055, 106, 'A');
Insert into SUB_CHOICE values (18055, 105, 'B');
Insert into SUB_CHOICE values (18055, 107, 'A');
```

2. **At the time of creation if we forget to create a field enrollment date (ENROLL_DATE) in ENROLL table so add the field.**

```
Alter Table ENROLLMENT Add (ENROLL_DATE date);
Update ENROLLMENT set ENROLL_DATE='19-NOV-21' where STUDENT_ID=18050;
Update ENROLLMENT set ENROLL_DATE='17-JUL-20' where STUDENT_ID=18051;
Update ENROLLMENT set ENROLL_DATE='14-DEC-22' where STUDENT_ID=18052;
Update ENROLLMENT set ENROLL_DATE='21-JUL-21' where STUDENT_ID=18053;
Update ENROLLMENT set ENROLL_DATE='04-JUN-21' where STUDENT_ID=18054;
Update ENROLLMENT set ENROLL_DATE='08-AUG-20' where STUDENT_ID=18055;
Update ENROLLMENT set ENROLL_DATE='13-SEP-22' where STUDENT_ID=18056;
Update ENROLLMENT set ENROLL_DATE='02-NOV-21' where STUDENT_ID=18057;
Update ENROLLMENT set ENROLL_DATE='23-JUL-20' where STUDENT_ID=18058;
Update ENROLLMENT set ENROLL_DATE='18-OCT-22' where STUDENT_ID=18069;
```

STUDENT_ID	COURSE_ID	ENROLL_DATE
18050	10	19-NOV-21
18051	20	17-JUL-20
18052	30	14-DEC-22
18053	20	21-JUL-21
18054	10	04-JUN-21
18055	40	08-AUG-20
18056	20	13-SEP-22
18057	30	02-NOV-21
18058	20	23-JUL-20
18069	20	18-OCT-22

3. **Course name cannot be blank, therefore add the criteria in the specific table.**

```
Alter Table COURSE
Modify COURSE_NAME varchar2(10) NOT NULL;
```

4. **Find the Course which has more than 3 students.**

Select * from (Select COUNT(COURSE_ID) as STUDENT_COUNT, COURSE_ID, COURSE_NAME from (Select ENROLLMENT.COURSE_ID, COURSE.COURSE_NAME from ENROLLMENT, COURSE where ENROLLMENT.COURSE_ID=COURSE.COURSE_ID) group by COURSE_ID, COURSE_NAME) where STUDENT_COUNT>3;

```
Select * from (Select COUNT(COURSE_ID) as STUDENT_COUNT, COURSE_ID, COURSE_NAME from
(Select ENROLLMENT.COURSE_ID, COURSE.COURSE_NAME from ENROLLMENT, COURSE
where ENROLLMENT.COURSE_ID=COURSE.COURSE_ID)
group by COURSE_ID, COURSE_NAME)
where STUDENT_COUNT>3;
```

STUDENT_COUNT	COURSE_ID	COURSE_NAME
5	20	IT

5. **Give the details of a STUDENT with all Subjects and Grade where he/she enrolls (Enter the sid value as input).**

Select
STUDENT.STUDENT_ID,
STUDENT.STUDENT_NAME,
SUBJECT.SUBJECT_ID,
SUBJECT.SUBJECT_NAME,
SUB_CHOICE.GRADE
from STUDENT
INNER JOIN SUB_CHOICE on STUDENT.STUDENT_ID = SUB_CHOICE.STUDENT_ID
INNER JOIN SUBJECT on SUB_CHOICE.SUBJECT_ID = SUBJECT.SUBJECT_ID
where STUDENT.STUDENT_ID = 18053

```
Select
STUDENT.STUDENT_ID,
STUDENT.STUDENT_NAME,
SUBJECT.SUBJECT_ID,
SUBJECT.SUBJECT_NAME,
SUB_CHOICE.GRADE
from STUDENT
INNER JOIN SUB_CHOICE on STUDENT.STUDENT_ID = SUB_CHOICE.STUDENT_ID
INNER JOIN SUBJECT on SUB_CHOICE.SUBJECT_ID = SUBJECT.SUBJECT_ID
where STUDENT.STUDENT_ID = 18053;
```

STUDENT_ID	STUDENT_NAME	SUBJECT_ID	SUBJECT_NAME	GRADE
18053	PRAMA	101	DBMS	A
18053	PRAMA	102	OOP	B
18053	PRAMA	103	DSA	A
18053	PRAMA	104	MATHS	A
18053	PRAMA	105	ELECTRONICS	A
18053	PRAMA	106	COA	A

6. **Display the course where the maximum number of students enrolls.**

```
Select STUDENT_COUNT, COURSE_NAME from (
select COUNT(COURSE_ID) as STUDENT_COUNT, COURSE_ID, COURSE_NAME
from (select ENROLLMENT.COURSE_ID, COURSE.COURSE_NAME
from ENROLLMENT, COURSE where
ENROLLMENT.COURSE_ID=COURSE.COURSE_ID) group by COURSE_ID,
COURSE_NAME)
where STUDENT_COUNT = (select MAX(STUDENT_COUNT) from (
select COUNT(COURSE_ID) as STUDENT_COUNT, COURSE_ID, COURSE_NAME
from (select ENROLLMENT.COURSE_ID, COURSE.COURSE_NAME from ENROLLMENT,
COURSE where ENROLLMENT.COURSE_ID=COURSE.COURSE_ID) group by
COURSE_ID, COURSE_NAME));
```

STUDENT_COUNT	COURSE_NAME
5	IT

7. **Find out the course where no student is enrolled.**

```
Select COURSE.COURSE_ID, COURSE_NAME from COURSE left join STUDENT on
COURSE.COURSE_ID = STUDENT.COURSE_ID where STUDENT_ID is NULL;
```

COURSE_ID	COURSE_NAME
50	IEE

8. **Delete Course no 30 from COURSE table.**

```
Delete from COURSE where COURSE_ID = 30;
```

```
Statement processed.
```

9. **Rename the COURSE table as DEPARTMENT.**

```
Rename COURSE to DEPARTMENT;
```

```
Statement processed.
```

10. Change the Marks Grade of Student “A” to “B” who is Enroll in the subject DBMS.

Update SUB_CHOICE set GRADE='B' where SUBJECT_ID =103 and GRADE='A';

```
1 row(s) updated.
```

11. Delete the record of the student who is enrolled in the course ‘IT’.

Delete from STUDENT where COURSE_ID = 20;

```
5 row(s) deleted.
```

12. Change the enroll date to ‘16-08-2018’ whose student id is 18069 (first convert the date into the default format).

Update ENROLLMENT set ENROLL_DATE = '16-AUG-18' where STUDENT_ID = 18069;

STUDENT_ID	COURSE_ID	ENROLL_DATE
18050	10	19-NOV-21
18051	20	17-JUL-20
18052	30	14-DEC-22
18053	20	21-JUL-21
18054	10	04-JUN-21
18055	40	08-AUG-20
18056	20	13-SEP-22
18057	30	02-NOV-21
18058	20	23-JUL-20
18069	20	16-AUG-18