Comsats University Islamabad

Lab – 06

Database Systems

Submitted to

Ms. Nimra Iqram

Submitted by

Muhammad Ahmad (FA23-BCE-113)

Computer Engineering

➔School Management System:

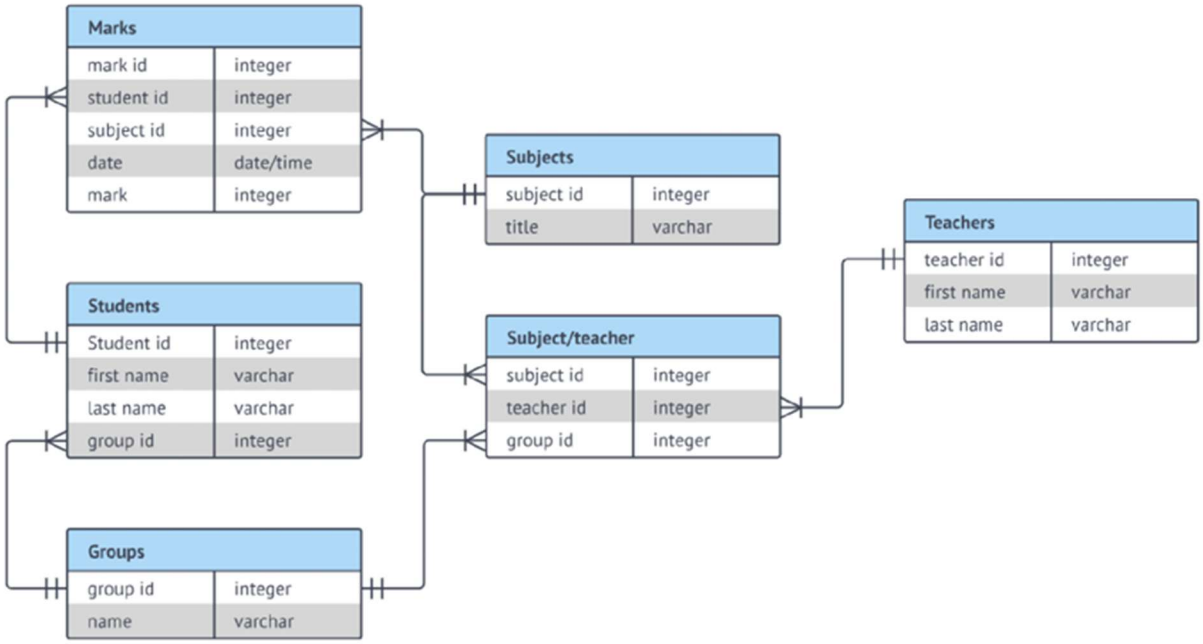


Table 1: A basic ER model of a school

➔TASKS:

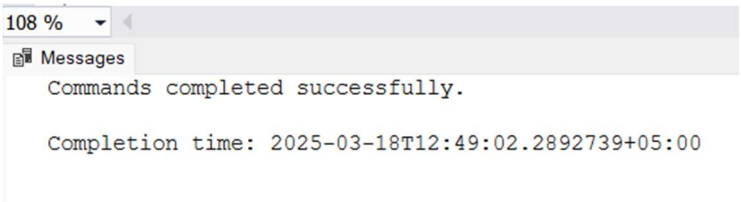
Task 1: To implement an ERD as per the given requirements, entity sets must be created as provided.

CODE:

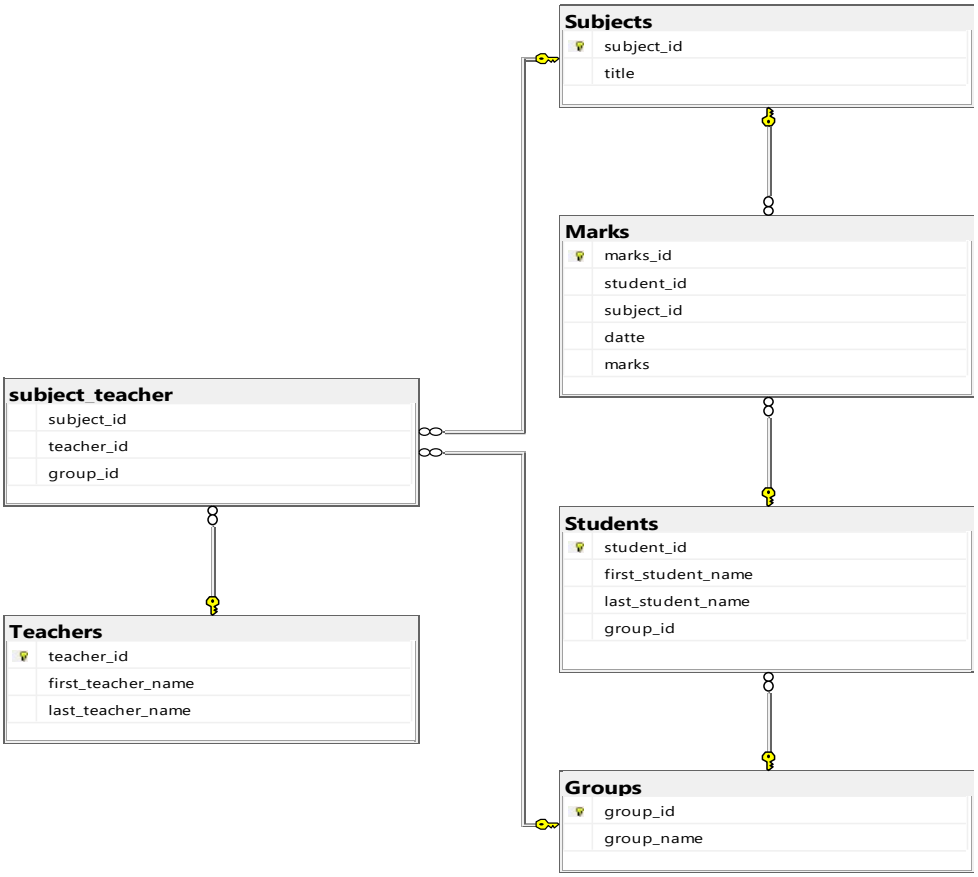
```
-- Subjets
create table Subjects(
    subject_id int primary key,
    title varchar(50) not null
);
-- Teacher
create table Teachers(
    teacher_id int primary key,
    first_teacher_name varchar(15) not null,
    last_teacher_name varchar(15)
);
-- Groups
create table Groups(
    group_id int primary key,
    group_name varchar(15) not null
);
-- teacher-subject
create table subject_teacher(
    subject_id int foreign key references Subjects(subject_id),
    teacher_id int foreign key references Teachers(teacher_id),
    group_id int foreign key references Groups(group_id)
    );
-- Students
```

```sql
create table Students(
    student_id int primary key,
    first_student_name varchar(15) not null,
    last_student_name varchar(15),
    group_id int foreign key references Groups(group_id)
);
-- Marks
create table Marks(
    marks_id int primary key,
    student_id int foreign key references Students(student_id),
    subject_id int foreign key references Subjects(subject_id),
    datte Date, -- date is misspelled to make difference with datatype(DATE)
    marks int not null
);
```

EXECUTION:



ER-Diagram generated by SSMS according to my data



Task 2: Constraints/rules must specified for all entities as per the relationships in Table. 1.

Constraints and rules are specified during the creation time (done above).

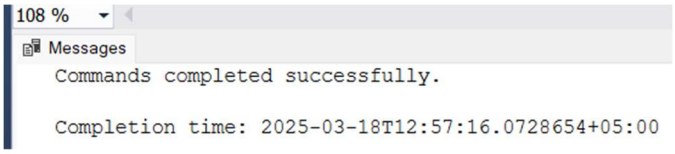
Task 3: Populate the data with instances.

I'm showing only 4 to 5 records to save space.

CODE:

```
INSERT INTO Groups VALUES
(1, 'BCE'),
(2, 'BEE-P'),
(3, 'BEE-E'),
(4, 'BEE-T');

INSERT INTO Teachers VALUES
(1, 'Usman', 'Rafiq'),
(2, 'Jehangir', 'Arshad'),
(3, 'Modassir', 'Ishfaq'),
(4, 'Imran', 'Ahmed'),
(17, 'Sobia', 'Baig');

INSERT INTO Subjects VALUES
(1, 'Islamic Studies'),
(2, 'Discrete Mathematics'),
(3, 'Linear Algebra'),
(4, 'Digital Logic Design'),
(24, 'Entrepreneurship');

INSERT INTO Students VALUES
(1, 'Ali', 'Khan', 1),
(2, 'Ahmed', 'Raza', 2),
(3, 'Sara', 'Ali', 3),
(4, 'Fatima', 'Ahmed', 4),
(5, 'Usman', 'Khan', 1);

INSERT INTO Subject_Teacher VALUES
(1, 1, 1),
(2, 2, 1),
(3, 3, 2),
(4, 4, 3),
(17, 17, 4);

INSERT INTO Marks VALUES
(1, 1, 1, '2023-10-01', 85),
(2, 1, 2, '2023-10-02', 90),
(3, 2, 3, '2023-10-03', 78),
(6, 5, 6, '2023-10-06', 80);
```

EXECUTION:



Task 4: Analyze your implementation and see if the rules are implemented appropriately for all entities.

CODE:

```
SELECT * FROM Subjects;
SELECT * FROM Teachers;
SELECT * FROM Groups;
SELECT * FROM Subject_Teacher;
SELECT * FROM Students;
SELECT * FROM Marks;
```

EXECUTION:

108 %

Results | Messages

	subject_id	title
1	1	Islamic Studies
2	2	Discrete Mathematics
3	3	Linear Algebra

	teacher_id	first_teacher_name	last_teacher_name
1	1	Usman	Rafiq
2	2	Jehangir	Arshad

	group_id	group_name
1	1	BCE
2	2	BEE-P

	subject_id	teacher_id	group_id
4	4	4	3
5	5	5	4

	student_id	first_student_name	last_student_name	group_id
1	1	Ali	Khan	1
2	2	Ahmed	Raza	2

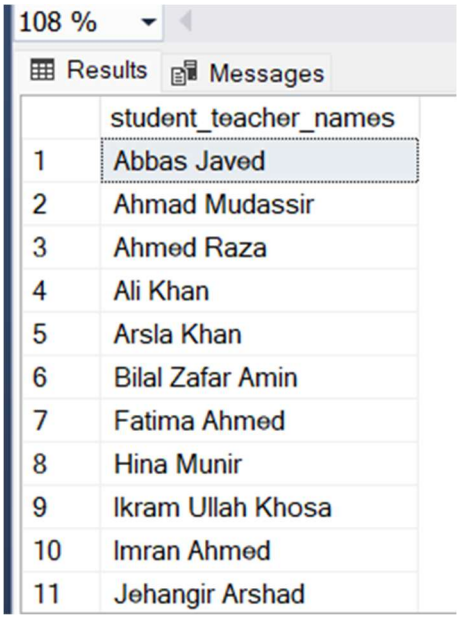
	marks_id	student_id	subject_id	datte	marks
1	1	1	1	2023-10-01	85
2	2	1	2	2023-10-02	90

Task 5: Retrieve the name of all students and teachers (Hint: Union operator)

CODE:

```
select first_student_name + ' ' + last_student_name as student_teacher_names
from Students
UNION
select first_teacher_name + ' ' + last_teacher_name from Teachers
```

EXECUTION:



108 %

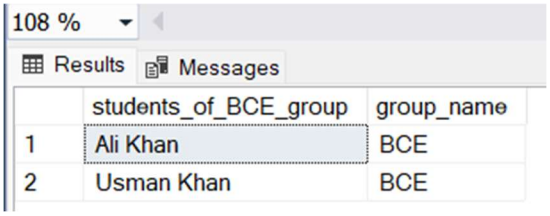
Results | Messages

	student_teacher_names
1	Abbas Javed
2	Ahmad Mudassir
3	Ahmed Raza
4	Ali Khan
5	Arsla Khan
6	Bilal Zafar Amin
7	Fatima Ahmed
8	Hina Munir
9	Ikram Ullah Khosa
10	Imran Ahmed
11	Jehangir Arshad

Task 6: Retrieve the name of all students of BCE group.

```
--inner join
SELECT s.first_student_name + ' ' + s.last_student_name  as
students_of_BCE_group, g.group_name
FROM Students s
JOIN Groups g ON s.group_id = g.group_id
WHERE g.group_name = 'BCE';
--sub-query
select first_student_name + ' ' + last_student_name students_of_BCE_group
from Students
where group_id = (select group_id from Groups where group_name = 'BCE')
```

EXECUTION:



108 %

Results | Messages

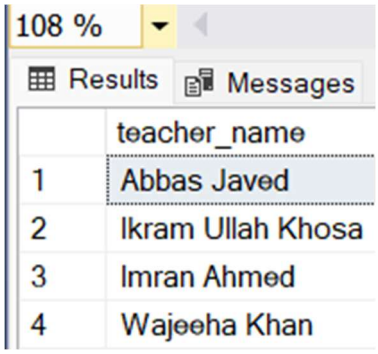
	students_of_BCE_group	group_name
1	Ali Khan	BCE
2	Usman Khan	BCE

Task 7: Retrieve the name of all teachers who taught the students of BEE-E group.

CODE:

```
SELECT DISTINCT t.first_teacher_name + ' ' + t.last_teacher_name AS
teacher_name
FROM Teachers t
JOIN Subject_Teacher st ON t.teacher_id = st.teacher_id
JOIN Groups g ON st.group_id = g.group_id
WHERE g.group_name = 'BEE-E';
```

EXECUTION:

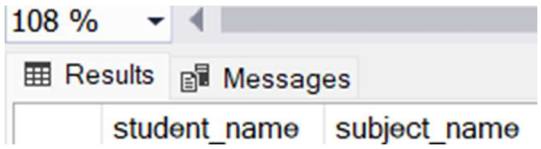


Task 8: Retrieve the name of all students & subject name who were registered in any subject taught by Dr. Abbas Javed

CODE:

```
SELECT DISTINCT s.first_student_name + ' ' + s.last_student_name AS
student_name,
                sub.title AS subject_name
FROM Students s
JOIN Marks m ON s.student_id = m.student_id
JOIN Subjects sub ON m.subject_id = sub.subject_id
JOIN Subject_Teacher st ON sub.subject_id = st.subject_id
JOIN Teachers t ON st.teacher_id = t.teacher_id
WHERE t.first_teacher_name = 'Abbas' AND t.last_teacher_name = 'Javed';
```

EXECUTION:



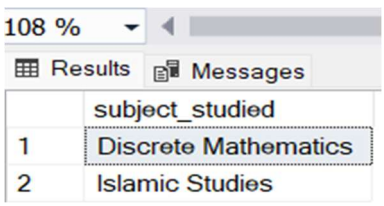
No records

Task 9: Supposedly, there is a student named Ali. Retrieve all subjects studied by Ali.

CODE:

```
SELECT DISTINCT sub.title AS subject_studied
FROM Subjects sub
JOIN Marks m ON sub.subject_id = m.subject_id
JOIN Students s ON m.student_id = s.student_id
WHERE s.first_student_name = 'Ali';
```

EXECUTION:

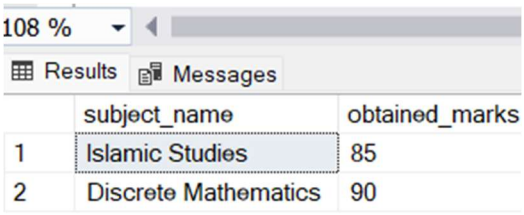


Task 10: Building upon the previous task, retrieve the obtained marks too

CODE

```
SELECT sub.title AS subject_name, m.marks AS obtained_marks
FROM Subjects sub
JOIN Marks m ON sub.subject_id = m.subject_id
JOIN Students s ON m.student_id = s.student_id
WHERE s.first_student_name = 'Ali';
```

EXECUTION:



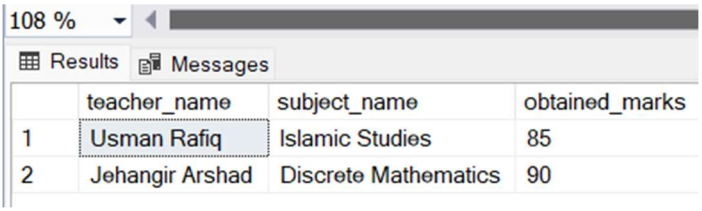
	subject_name	obtained_marks
1	Islamic Studies	85
2	Discrete Mathematics	90

Task 11: Continuing from the previous task, retrieve the teachers along with the previous details as well.

CODE

```
SELECT t.first_teacher_name + ' ' + t.last_teacher_name AS teacher_name,
       sub.title AS subject_name,
       m.marks AS obtained_marks
FROM Teachers t
JOIN Subject_Teacher st ON t.teacher_id = st.teacher_id
JOIN Subjects sub ON st.subject_id = sub.subject_id
JOIN Marks m ON sub.subject_id = m.subject_id
JOIN Students s ON m.student_id = s.student_id
WHERE s.first_student_name = 'Ali';
```

EXECUTION:



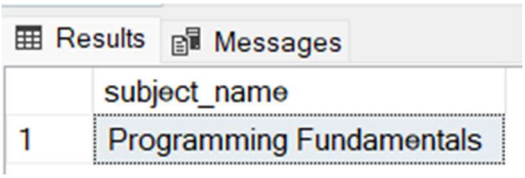
	teacher_name	subject_name	obtained_marks
1	Usman Rafiq	Islamic Studies	85
2	Jehangir Arshad	Discrete Mathematics	90

Task 12: Retrieve the name of all subjects taught by Dr. Arsla Khan

CODE

```
SELECT DISTINCT sub.title AS subject_name
FROM Subjects sub
JOIN Subject_Teacher st ON sub.subject_id = st.subject_id
JOIN Teachers t ON st.teacher_id = t.teacher_id
WHERE t.first_teacher_name = 'Arsla' AND t.last_teacher_name = 'Khan';
```

EXECUTION:

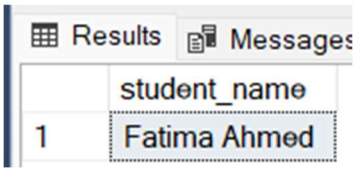


Task 13: Retrieve the name of students who are part of BEE-T group

CODE

```
SELECT s.first_student_name + ' ' + s.last_student_name AS student_name
FROM Students s
JOIN Groups g ON s.group_id = g.group_id
WHERE g.group_name = 'BEE-T';
```

EXECUTION:

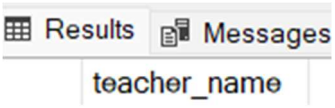


Task 14: Retrieve the name of teachers who are not teaching any subject

CODE

```
SELECT t.first_teacher_name + ' ' + t.last_teacher_name AS teacher_name
FROM Teachers t
LEFT JOIN Subject_Teacher st ON t.teacher_id = st.teacher_id
WHERE st.subject_id IS NULL;
```

EXECUTION:



No records

Task 15: Retrieve the name of students along with their grades for the subjects that were taught by Dr. Zaid Ahmed

CODE

```
SELECT s.first_student_name + ' ' + s.last_student_name AS student_name,
       sub.title AS subject_name,
       m.marks AS obtained_marks,
       CASE
           WHEN m.marks >= 90 THEN 'A'
           WHEN m.marks >= 80 THEN 'B'
           WHEN m.marks >= 70 THEN 'C'
           WHEN m.marks >= 60 THEN 'D'
```

```
            ELSE 'F'
        END AS grade
FROM Students s
JOIN Marks m ON s.student_id = m.student_id
JOIN Subjects sub ON m.subject_id = sub.subject_id
JOIN Subject_Teacher st ON sub.subject_id = st.subject_id
JOIN Teachers t ON st.teacher_id = t.teacher_id
WHERE t.first_teacher_name = 'Zaid' AND t.last_teacher_name = 'Ahmed';
```

EXECUTION:

Results | Messages

	student_name	subject_name	obtained_marks	grade
1	Sara Ali	Probability Methods in Engineering	85	B