ASSIGNMENT 11

1. **Create a database named "AdultLiteracy" on your RDBMS environment. Using Figure 7-5 above, write DDL commands to create table structures for each entity above. Name your tables the following names: Tutor, Student, MatchHistory, TutorReport**

To create database

```
IF EXISTS (SELECT name FROM sys.databases WHERE name = N'AdultLiteracy')
    DROP DATABASE AdultLiteracy
GO

CREATE DATABASE [AdultLiteracy]
go
```

To Use mentioned database

```
USE [AdultLiteracy]
GO
```

Create tables:

```
CREATE TABLE Tutor (
    TutorID INT PRIMARY KEY NOT NULL identity(100,1),
    cartDate date NOT NULL,
    Status VARCHAR(15) NOT NULL CHECK (Status IN('Active', 'Temp Stop', 'Dropped')),
);

CREATE TABLE Student (
    StudentID INT PRIMARY KEY NOT NULL identity(3000,1),
    [Read] INT NOT NULL,
);


CREATE TABLE MatchHistory (
    MatchID INT PRIMARY KEY NOT NULL identity(1,1),
    TutorID INT,
    StudentID INT,
    StartDate date,
    EndDate date,
    FOREIGN KEY (TutorID) REFERENCES Tutor(TutorID),
    FOREIGN KEY (StudentID) REFERENCES Student(StudentID)
);


CREATE TABLE TutorReport (
```

```
    MatchID INT,
    [Month] VARCHAR(10),
    [Hours] INT,
    Lessons INT,
    FOREIGN KEY (MatchID) REFERENCES MatchHistory(MatchID)
);
```

2. **Write SQL scripts to insert sample data from Fig 7-5 into the database.**

```
INSERT INTO Tutor (cartDate, [Status])
VALUES
    (CONVERT(DATE, '05/01/2008', 103), ‘Active'),
    (CONVERT(DATE, '05/01/2008', 103), 'Temp Stop'),
    (CONVERT(DATE, '05/01/2008', 103), 'Dropped'),
    (CONVERT(DATE, '22/05/2008', 103), 'Active'),
    (CONVERT(DATE, '22/05/2008', 103), 'Active'),
    (CONVERT(DATE, '22/05/2008', 103), 'Temp Stop'),
    (CONVERT(DATE, '22/05/2008', 103), 'Active');
```

```
[7]   1   select * from Tutor;
```

SQL

(7 rows affected)

Total execution time: 00:00:00.017

	TutorID	cartDate	Status
1	100	2008-01-05	Active
2	101	2008-01-05	Temp Stop
3	102	2008-01-05	Dropped
4	103	2008-05-22	Active
5	104	2008-05-22	Active
6	105	2008-05-22	Temp Stop
7	106	2008-05-22	Active

```sql
INSERT INTO Student ([Read])
VALUES
    (2.3),
    (5.6),
    (1.3),
    (3.3),
    (2.7),
    (4.8),
    (7.8),
    (1.5);
```

```sql
[9]   1   select * from Student;
```

SQL

(8 rows affected)

Total execution time: 00:00:00.025

	StudentID	Read
1	3000	2
2	300 3000	5
3	3002	1
4	3003	3
5	3004	2
6	3005	4
7	3006	7
8	3007	1

```sql
INSERT INTO MatchHistory (TutorID, StudentID, StartDate, EndDate)
VALUES
    (100, 3000, CONVERT(DATE, '10/01/2008'), null),
    (101, 3001, CONVERT(DATE, '15/01/2008', 103), CONVERT(DATE, '15/05/2008', 103)),
    (102, 3002, CONVERT(DATE, '10/02/2008', 103), CONVERT(DATE, '01/03/2008', 103)),
    (106, 3003, CONVERT(DATE, '28/05/2008', 103), null),
    (103, 3004, CONVERT(DATE, '01/06/2008', 103), CONVERT(DATE, '15/06/2008', 103)),
    (104, 3005, CONVERT(DATE, '01/06/2008', 103), CONVERT(DATE, '28/06/2008', 103)),
    (104, 3006, CONVERT(DATE, '01/06/2008', 103), null);
;
```

```
[13]    1    select * from MatchHistory;
```

(7 rows affected)

Total execution time: 00:00:00.030

	MatchID	TutorID	StudentID	StartDate	EndDate
1	1	100	3000	2008-10-01	*NULL*
2	2	101	3001	2008-01-15	2008-05-15
3	3	102	3002	2008-02-10	2008-03-01
4	4	106	3003	2008-05-28	*NULL*
5	5	103	3004	2008-06-01	2008-06-15
6	6	104	3005	2008-06-01	2008-06-28
7	7	104	3006	2008-06-01	*NULL*

Results grid

```
INSERT INTO TutorReport (MatchID, [Month], [Hours], Lessons)
VALUES
    (1, '06/08', 8, 4),
    (4, '06/08', 8, 6),
    (5, '06/08', 4, 4),
    (4, '07/08', 10, 5),
    (1, '07/08', 4, 2);
```

```
[15]    1    select * from TutorReport;
```

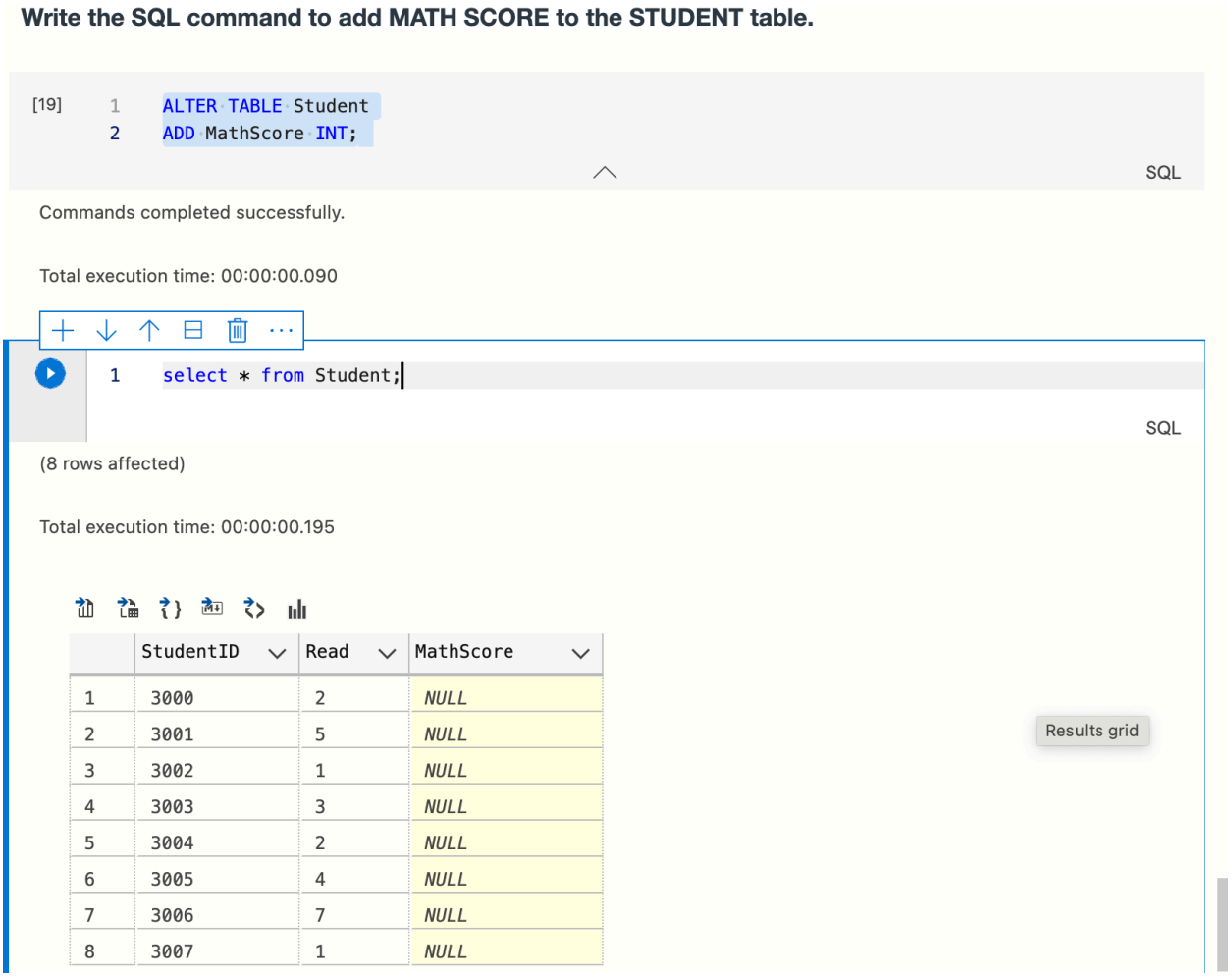
(5 rows affected)

Total execution time: 00:00:00.032

	MatchID	Month	Hours	Lessons
1	1	06/08	8	4
2	4	06/08	8	6
3	5	06/08	4	4
4	4	07/08	10	5
5	1	07/08	4	2

3. **Write the SQL command to add MATH SCORE to the STUDENT table.**

```
ALTER TABLE Student
ADD MathScore INT;
```



	StudentID	Read	MathScore
1	3000	2	NULL
2	3001	5	NULL
3	3002	1	NULL
4	3003	3	NULL
5	3004	2	NULL
6	3005	4	NULL
7	3006	7	NULL
8	3007	1	NULL

4. **Write the SQL command to add SUBJECT to TUTOR. The only values allowed for SUBJECT will be Reading, Math, and ESL.**

```
ALTER TABLE Tutor
ADD [Subject] VARCHAR(15) CHECK ([Subject] IN('Reading', 'Math', 'ESL'));
```

```
[21]  1  ALTER TABLE Tutor
      2  ADD [Subject] VARCHAR(15) CHECK ([Subject] IN('Reading', 'Math', 'ESL'));
```

SQL

Commands completed successfully.

Total execution time: 00:00:00.171

```
[22]  1  select * from tutor;
```

SQL

(7 rows affected)

Total execution time: 00:00:00.046

	TutorID	cartDate	Status	Subject
1	100	2008-01-05	Active	*NULL*
2	101	2008-01-05	Temp Stop	*NULL*
3	102	2008-01-05	Dropped	*NULL*
4	103	2008-05-22	Active	*NULL*
5	104	2008-05-22	Active	*NULL*
6	105	2008-05-22	Temp Stop	*NULL*
7	106	2008-05-22	Active	*NULL*

5. **What do you need to do if a tutor signs up and wants to tutor in both reading and math? (Don't need to write SQL).**

To allow tutors to teach multiple subjects, we need to modify the database schema. Creating a new table, "TutorSubject," which links tutors to the subjects they are teaching.

The "TutorSubject" table should have columns for "TutorID" and "Subject." "TutorID" connects to the "Tutor" table, while "Subject" lists the subjects each tutor can teach. This structure enables a tutor to be assigned multiple subjects.

For example, if a tutor can teach both reading and math, two records are added to the "TutorSubject" table: one for reading and one for math, both linked to the same tutor's "TutorID."

6. Write the SQL command to find any tutors who have not submitted a report for July.

```sql
SELECT DISTINCT t.TutorID FROM Tutor t
LEFT JOIN MatchHistory mh ON t.TutorID = mh.TutorID
LEFT JOIN TutorReport tr ON mh.MatchID = tr.MatchID AND tr.month = '07/08'
WHERE tr.MatchID IS NULL ;
```

Write the SQL command to find any tutors who have not submitted a report for July.

```sql
select * from MatchHistory;
select * from TutorReport;
```

SQL

(7 rows affected)

(5 rows affected)

Total execution time: 00:00:00.030

	MatchID	TutorID	StudentID	StartDate	EndDate
1	1	100	3000	2008-10-01	NULL
2	2	101	3001	2008-01-15	2008-05-15
3	3	102	3002	2008-02-10	2008-03-01
4	4	106	3003	2008-05-28	NULL
5	5	103	3004	2008-06-01	2008-06-15
6	6	104	3005	2008-06-01	2008-06-28
7	7	104	3006	2008-06-01	NULL

	MatchID	Month	Hours	Lessons
1	1	06/08	8	4
2	4	06/08	8	6
3	5	06/08	4	4
4	4	07/08	10	5
5	1	07/08	4	2

There are two potential scenarios for tutors who have not submitted their report in July:

1. The tutor has not submitted any report yet.
2. The tutor may have submitted a report, but not for the month of July.

We can filter the data based on either of these conditions.

```sql
SELECT DISTINCT t.TutorID
FROM Tutor t
LEFT JOIN MatchHistory mh ON t.TutorID = mh.TutorID
LEFT JOIN TutorReport tr ON mh.MatchID = tr.MatchID
AND tr.month = '07/08'
WHERE tr.MatchID IS NULL ;
```

SQL

(5 rows affected)

Collapse code cell contents

Total execution time: 00:00:00.035

	TutorID
1	101
2	102
3	103
4	104
5	105