```sql
-- Create Students table
CREATE TABLE Student (
    student_id INT PRIMARY KEY IDENTITY(1,1),
    first_name VARCHAR(50),
    last_name VARCHAR(50),
    email VARCHAR(100),
    date_of_birth DATE
);

CREATE TABLE Coursee (
    course_id INT PRIMARY KEY IDENTITY(1,1),
    course_name VARCHAR(100),
    course_description TEXT,
    instructor_id INT,
    FOREIGN KEY (instructor_id) REFERENCES Instructors(instructor_id)
);


-- Create Instructors table
CREATE TABLE Instructorss (
    instructor_id INT PRIMARY KEY IDENTITY(1,1),
    first_name VARCHAR(50),
    last_name VARCHAR(50),
    email VARCHAR(100)
);

-- Create Enrollments table
CREATE TABLE Enrollmentss (
    enrollment_id INT PRIMARY KEY IDENTITY(1,1),
    student_id INT,
    course_id INT,
    enrollment_date DATE,
    FOREIGN KEY (student_id) REFERENCES Students(student_id),
    FOREIGN KEY (course_id) REFERENCES Courses(course_id)
);
```

```sql
-- Insert data into Students table
INSERT INTO Students (first_name, last_name, email, date_of_birth) VALUES
('John', 'Doe', 'john.doe@example.com', '2000-01-15'),
('Jane', 'Smith', 'jane.smith@example.com', '2001-02-20'),
('Michael', 'Johnson', 'michael.johnson@example.com', '1999-03-25'),
('Emily', 'Davis', 'emily.davis@example.com', '2002-04-30'),
('David', 'Wilson', 'david.wilson@example.com', '2000-05-10'),
('Sarah', 'Brown', 'sarah.brown@example.com', '2001-06-15'),
('James', 'Jones', 'james.jones@example.com', '1999-07-20'),
('Linda', 'Garcia', 'linda.garcia@example.com', '2002-08-25'),
('Robert', 'Martinez', 'robert.martinez@example.com', '2000-09-10'),
('Patricia', 'Hernandez', 'patricia.hernandez@example.com', '2001-10-15');

-- Insert data into Courses table
INSERT INTO Course (course_name, course_description) VALUES
('Introduction to Computer Science', 'An introductory course to computer science.'),
('Data Structures', 'A course on the implementation and usage of data structures.'),
('Database Systems', 'An in-depth look at database design and SQL.'),
('Operating Systems', 'A course on the fundamentals of operating systems.'),
('Algorithms', 'An advanced course on algorithm design and analysis.');

-- Insert data into Instructors table
INSERT INTO Instructors (first_name, last_name, email) VALUES
('Alice', 'Taylor', 'alice.taylor@example.com'),
('Bob', 'Anderson', 'bob.anderson@example.com'),
('Carol', 'Thomas', 'carol.thomas@example.com');

-- Insert data into Enrollments table
INSERT INTO Enrollments (student_id, course_id, enrollment_date) VALUES
(1, 1, '2024-08-01'),
(2, 1, '2024-08-02'),
(3, 2, '2024-08-03'),
(4, 3, '2024-08-04'),
(5, 4, '2024-08-05'),
(6, 5, '2024-08-06'),
```

```
select * from Courses
```

100 %

Results | Messages

	course_id	course_name	course_description
1	1	Introduction to Computer Science	An introductory course to computer science.
2	2	Data Structures	A course on the implementation and usage of data ...
3	3	Database Systems	An in-depth look at database design and SQL.
4	4	Operating Systems	A course on the fundamentals of operating systems.
5	5	Algorithms	An advanced course on algorithm design and analy...

```
select * from Students
```

100 %

Results | Messages

	student_id	first_name	last_name	email	date_of_birth
1	1	John	Doe	john.doe@example.com	2000-01-15
2	2	Jane	Smith	jane.smith@example.com	2001-02-20
3	3	Michael	Johnson	michael.johnson@example.com	1999-03-25
4	4	Emily	Davis	emily.davis@example.com	2002-04-30
5	5	David	Wilson	david.wilson@example.com	2000-05-10
6	6	Sarah	Brown	sarah.brown@example.com	2001-06-15
7	7	James	Jones	james.jones@example.com	1999-07-20
8	8	Linda	Garcia	linda.garcia@example.com	2002-08-25
9	9	Robert	Martinez	robert.martinez@example.com	2000-09-10
10	10	Patricia	Hernand...	patricia.hernandez@example....	2001-10-15

```sql
select
    E.*,
    C.course_name,
    S.first_name,
    S.last_name
from Enrollments as E
JOIN Courses C
ON E.course_id = C.course_id
JOIN Students S
ON E.student_id = S.student_id;
```

100 %

Results | Messages

	enrollment_id	student_id	course_id	enrollment_date	course_name	first_name	last_name
1	1	1	1	2024-08-01	Introduction to Computer Science	John	Doe
2	2	2	1	2024-08-02	Introduction to Computer Science	Jane	Smith
3	3	3	2	2024-08-03	Data Structures	Michael	Johnson
4	4	4	3	2024-08-04	Database Systems	Emily	Davis
5	5	5	4	2024-08-05	Operating Systems	David	Wilson
6	6	6	5	2024-08-06	Algorithms	Sarah	Brown
7	7	7	1	2024-08-07	Introduction to Computer Science	James	Jones
8	8	8	2	2024-08-08	Data Structures	Linda	Garcia
9	9	9	3	2024-08-09	Database Systems	Robert	Martinez
10	10	10	4	2024-08-10	Operating Systems	Patricia	Hernandez
11	11	1	2	2024-08-11	Data Structures	John	Doe
12	12	2	3	2024-08-12	Database Systems	Jane	Smith
13	13	3	4	2024-08-13	Operating Systems	Michael	Johnson
14	14	4	5	2024-08-14	Algorithms	Emily	Davis
15	15	5	1	2024-08-15	Introduction to Computer Science	David	Wilson

```sql
--Advanced Queries
--1
select * from Students , Courses
where course_name = 'Operating Systems'
```

100 %

Results | Messages

	student_id	first_name	last_name	email	date_of_birth	course_id	course_name	course_description
1	1	John	Doe	john.doe@example.com	2000-01-15	4	Operating Systems	A course on the fundamentals of operating systems.
2	2	Jane	Smith	jane.smith@example.com	2001-02-20	4	Operating Systems	A course on the fundamentals of operating systems.
3	3	Michael	Johnson	michael.johnson@example.com	1999-03-25	4	Operating Systems	A course on the fundamentals of operating systems.
4	4	Emily	Davis	emily.davis@example.com	2002-04-30	4	Operating Systems	A course on the fundamentals of operating systems.
5	5	David	Wilson	david.wilson@example.com	2000-05-10	4	Operating Systems	A course on the fundamentals of operating systems.
6	6	Sarah	Brown	sarah.brown@example.com	2001-06-15	4	Operating Systems	A course on the fundamentals of operating systems.
7	7	James	Jones	james.jones@example.com	1999-07-20	4	Operating Systems	A course on the fundamentals of operating systems.
8	8	Linda	Garcia	linda.garcia@example.com	2002-08-25	4	Operating Systems	A course on the fundamentals of operating systems.
9	9	Robert	Martinez	robert.martinez@example.com	2000-09-10	4	Operating Systems	A course on the fundamentals of operating systems.
10	10	Patricia	Hernandez	patricia.hernandez@example.com	2001-10-15	4	Operating Systems	A course on the fundamentals of operating systems.

```sql
--2
SELECT COUNT(e.[course_id]) AS student_count,c.[course_name]
FROM Enrollments e, Courses c
WHERE e.[course_id] = c.[course_id]
GROUP BY e.[course_id], c.[course_name]
HAVING COUNT(e.[course_id]) > 3;
```

0 %

Results | Messages

student_count	course_name
4	Introduction to Computer Science

```sql
--3
update [dbo].[Students]
set [email] = 'omarnouh@example.com'
where [student_id] = 3

select email from [dbo].[Students]
```

100 %

Results | Messages

	email
1	john.doe@example.com
2	jane.smith@example.com
3	omarnouh@example.com
4	emily.davis@example.com
5	david.wilson@example.com
6	sarah.brown@example.com
7	james.jones@example.com
8	linda.garcia@example.com
9	robert.martinez@example.com
10	patricia.hernandez@example.com

```sql
--4
DELETE FROM Course
WHERE course_id NOT IN (SELECT DISTINCT course_id FROM Enrollments);
```

0 %

Messages

```
(0 rows affected)

Completion time: 2024-08-27T21:27:56.9872176+03:00
```

```sql
--5
SELECT AVG(DATEDIFF(YEAR, date_of_birth, GETDATE())) AS average_age
FROM Students;
```

100 %

Results | Messages

	average_age
1	23

```sql
select count([course_id]) as courses_count
from [dbo].[Enrollments]
group by [course_id]
```

100 %

Results | Messages

	courses_count
1	4
2	3
3	3
4	3
5	2

```sql
--Join Queries
--1
select s.*,c.[course_name] from [dbo].[Students] as s
join [dbo].[Enrollments] as e on s.[student_id] = e.[student_id]
join [dbo].[Courses] as c on e.[course_id] = c.[course_id]
--2

--3
```

00 %

Results | Messages

	student_id	first_name	last_name	email	date_of_birth	course_name
1	1	John	Doe	john.doe@example.com	2000-01-15	Introduction to Computer Science
2	2	J		xample.com	2001-02-20	Introduction to Computer Science
3	3	Michael	Johnson	omarnouh@example.com	1999-03-25	Data Structures
4	4	Emily	Davis	emily.davis@example.com	2002-04-30	Database Systems
5	5	David	Wilson	david.wilson@example.com	2000-05-10	Operating Systems
6	6	Sarah	Brown	sarah.brown@example.com	2001-06-15	Algorithms
7	7	James	Jones	james.jones@example.com	1999-07-20	Introduction to Computer Science
8	8	Linda	Garcia	linda.garcia@example.com	2002-08-25	Data Structures
9	9	Robert	Martinez	robert.martinez@example.com	2000-09-10	Database Systems
10	10	Patricia	Hernandez	patricia.hernandez@example.com	2001-10-15	Operating Systems
11	1	John	Doe	john.doe@example.com	2000-01-15	Data Structures
12	2	Jane	Smith	jane.smith@example.com	2001-02-20	Database Systems
13	3	Michael	Johnson	omarnouh@example.com	1999-03-25	Operating Systems
14	4	Emily	Davis	emily.davis@example.com	2002-04-30	Algorithms
15	5	David	Wilson	david.wilson@example.com	2000-05-10	Introduction to Computer Science

Click to select the whole column

```sql
select distinct i.* ,  c.[course_name] from  [dbo].[Instructors] as i
join [dbo].[Course] as c  on i.[instructor_id] = c.[instructor_id]
join [dbo].[Enrollments] as e on c.[course_id] = e.[course_id]

--3
```

00 %

Results | Messages

	instructor_id	first_name	last_name	email	course_name
1	3	Carol	Thomas	carol.thomas@example.com	Algorithms
2	2	Bob	Anderson	bob.anderson@example.com	Data Structures
3	3	Carol	Thomas	carol.thomas@example.com	Database Systems
4	1	Alice	Taylor	alice.taylor@example.com	Introduction to Computer Science
5	1	Alice	Taylor	alice.taylor@example.com	Operating Systems

```sql
--3
select [first_name],[last_name] from [dbo].[Students] as s
left join [dbo].[Enrollments] as e on s.student_id = e.student_i
where e.student_id is null
```

%

Results | Messages

first_name	last_name
omar	nouh

```sql
--1
select s.[student_id] ,s.[first_name],s.[last_name],count(e.course_id) AS course_count
from [dbo].[Students] as s
join [dbo].[Enrollments] as e on s.student_id = e.student_id
GROUP BY
    s.student_id,
    s.first_name,
    s.last_name
having count(e.course_id)>1
--2
```

00 %

Results | Messages

	student_id	first_name	last_name	course_count
1	1	John	Doe	2
2	2	Jane	Smith	2
3	3	Michael	Johnson	2
4	4	Emily	Davis	2
5	5	David	Wilson	2

```sql
--2
select c.[course_name] , i.[first_name] from [dbo].[Course] as c
join [dbo].[Instructors] as i on c.instructor_id = i.instructor_id
where i.[first_name] = 'Carol'
--3

--4
```

100 %

Results | Messages

	course_name	first_name
1	Database Systems	Carol
2	Algorithms	Carol

```sql
--3
select top 3 s.[student_id] ,s.[first_name],s.[last_name],count(e.course_id) AS enrollment_count  from [dbo].[Students] as s
join [dbo].[Enrollments] as e on s.student_id = e.student_id
GROUP BY
    s.student_id,
    s.first_name,
    s.last_name
order by count(e.course_id) desc

--4
```

100 %

Results | Messages

	student_id	first_name	last_name	enrollment_count
1	4	Emily	Davis	2
2	3	Michael	Johnson	2
3	2	Jane	Smith	2

```sql
--4
SELECT s.student_id, s.first_name, s.last_name, c.course_name
FROM [dbo].[Students] AS s
JOIN [dbo].[Enrollments] AS e ON s.student_id = e.student_id
JOIN [dbo].[Courses] AS c ON e.course_id = c.course_id
WHERE c.course_name = 'Introduction to Computer Science'

UNION

SELECT s.student_id, s.first_name, s.last_name, c.course_name
FROM [dbo].[Students] AS s
JOIN [dbo].[Enrollments] AS e ON s.student_id = e.student_id
JOIN [dbo].[Courses] AS c ON e.course_id = c.course_id
WHERE c.course_name = 'Data Structures';
--Functions and Stored Procedures
--1
```

0 %

Results | Messages

student_id	first_name	last_name	course_name
1	John	Doe	Data Structures
1	John	Doe	Introduction to Computer Science
2	Jane	Smith	Introduction to Computer Science
3	Michael	Johnson	Data Structures
5	David	Wilson	Introduction to Computer Science
7	James	Jones	Introduction to Computer Science
8	Linda	Garcia	Data Structures

SQLQuery1.sql - O...OUH34\omarn (63))*

```
    @FirstName VARCHAR(100),
    @LastName VARCHAR(100),
    @Email VARCHAR(255),
    @DateOfBirth DATE
AS
BEGIN
    -- Insert the new student into the Students table
    INSERT INTO Students (first_name, last_name, email, date_of_birth)
    VALUES (@FirstName, @LastName, @Email, @DateOfBirth);
END;

EXEC AddNewStudent
    @FirstName = 'omar',
    @LastName = 'nouh',
    @Email = 'omar.nouh@example.com',
    @DateOfBirth = '2004-03-1';

select * from  [dbo].[Students]
```

100 %

Results | Messages

	student_id	first_name	last_name	email	date_of_birth
1	1	John	Doe	john.doe@example.com	2000-01-15
2	2	Jane	Smith	jane.smith@example.com	2001-02-20
3	3	Michael	Johnson	omarnouh@example.com	1999-03-25
4	4	Emily	Davis	emily.davis@example.com	2002-04-30
5	5	David	Wilson	david.wilson@example.com	2000-05-10
6	6	Sarah	Brown	sarah.brown@example.com	2001-06-15
7	7	James	Jones	james.jones@example.com	1999-07-20
8	8	Linda	Garcia	linda.garcia@example.com	2002-08-25
9	9	Robert	Martinez	robert.martinez@example.com	2000-09-10
10	10	Patricia	Hernandez	patricia.hernandez@example.com	2001-10-15
11	11	omar	nouh	omar.nouh@example.com	2004-03-01

```sql
--2
CREATE PROCEDURE CalculateStudentAge
    @StudentID INT,
    @Age INT OUTPUT
AS
BEGIN
    -- Calculate the age based on the student's date of birth
    SELECT @Age = DATEDIFF(YEAR, date_of_birth, GETDATE())
                - CASE WHEN DATEADD(YEAR, DATEDIFF(YEAR, date_of_birth, GETDATE()), date_of_birth) > GETDATE()
                       THEN 1
                       ELSE 0
                  END
    FROM Students
    WHERE student_id = @StudentID;
END;

DECLARE @StudentAge INT;

EXEC CalculateStudentAge
    @StudentID = 1,
    @Age = @StudentAge OUTPUT;

SELECT @StudentAge AS StudentAge;
```

00 %

Results | Messages

StudentAge
24

```sql
--Aggregate Functions and Grouping
--1
select count([student_id]) as sum_of_students from [dbo].[Students]

--2
--Additional Tasks
--1


--2
```

00 %

Results | Messages

sum_of_students
11

```sql
--2
SELECT
    AVG(num_of_enrollment) AS avg_Enrollments,
    MAX(num_of_enrollment) AS max_Enrollments,
    MIN(num_of_enrollment) AS min_Enrollments
FROM (
    SELECT course_id,COUNT(*) AS num_of_enrollment
    FROM[dbo].[Enrollments]
    GROUP BY course_id
)   AS EnrollmentCounts;


--Additional Tasks
```

00 %

Results | Messages

avg_Enrollments	max_Enrollments	min_Enrollments
3	4	2

```sql
--Additional Tasks
--1
SELECT * ,
    DATEDIFF(YEAR, date_of_birth, GETDATE()) AS age,
    CASE
        WHEN DATEDIFF(YEAR, date_of_birth, GETDATE()) < 20 THEN 'Under 20'
        WHEN DATEDIFF(YEAR, date_of_birth, GETDATE()) BETWEEN 20 AND 29 THEN '20-29'
        WHEN DATEDIFF(YEAR, date_of_birth, GETDATE()) BETWEEN 30 AND 39 THEN '30-39'
        WHEN DATEDIFF(YEAR, date_of_birth, GETDATE()) >= 40 THEN '40 and above'
        ELSE 'Unknown'
    END AS age_category
FROM
    [dbo].[Students];

--2
```

00 %

Results | Messages

	student_id	first_name	last_name	email	date_of_birth	age	age_category
3	3	Michael	Johnson	omarnouh@example.com	1999-03-25	25	20-29
4	4	Emily	Davis	emily.davis@example.com	2002-04-30	22	20-29
5	5	David	Wilson	david.wilson@example.com	2000-05-10	24	20-29
6	6	Sarah	Brown	sarah.brown@example.com	2001-06-15	23	20-29
7	7	James	Jones	james.jones@example.com	1999-07-20	25	20-29
8	8	Linda	Garcia	linda.garcia@example.com	2002-08-25	22	20-29
9	9	Robert	Martinez	robert.martinez@example.com	2000-09-10	24	20-29
10	10	Patricia	Hernandez	patricia.hernandez@example.com	2001-10-15	23	20-29
11	11	omar	nouh	omar.nouh@example.com	2004-03-01	20	20-29

```sql
--2
SELECT c.course_id,c.course_name
FROM[dbo].[Courses] as c
WHERE
    EXISTS (SELECT 1
            FROM [dbo].[Enrollments] as e
            WHERE e.course_id = c.course_id);
```

00 %

Results | Messages

	course_id	course_name
1	1	Introduction to Computer Science
2	2	Data Structures
3	3	Database Systems
4	4	Operating Systems
5	5	Algorithms