MIS 4330 Video Activity – Week 2

1. **Show StudentID and the Description of the activities he or she is involved with. Arrange by Description from Z to A.**

```
SELECT
    sa.StudentID,
    a.Description
FROM Student_Activity AS sa
INNER JOIN Activity AS a
ON sa.ActivityID = a.ActivityID
ORDER BY Description DESC;
```

100 %

Results | Messages

	StudentID	Description
1	3	Karate
2	1	Basketball
3	2	Basketball
4	1	Baseball
5	2	Baseball
6	3	Baseball

2. **Show me the LockerNumber for the student that won the AwardID one.**

```
SELECT
    LockerNumber
FROM Student_Award AS sa
INNER JOIN Gym_locker AS ga
ON sa.StudentID = ga.StudentID
WHERE sa.AwardID = 1;
```

100 %

Results | Messages

	LockerNumber
1	224
2	225
3	228

3.

```
SELECT s.LastName,  count(gl.StudentID) as 'numlocker'
FROM STUDENT as s
left outer join gym_locker as gl
ON s.studentID = gl.studentID
Group By s.LastName;
```

100 %

Results | Messages

	LastName	numlocker
1	Beldon	1
2	Bell	1
3	Benson	1
4	Brymerski	1
5	Davis	0
6	Dewer	1
7	Hall	1
8	Henry	0
9	Hinz	1
10	Holland	0
11	Johnson	0
12	Kavlie	1

4. Show me students' last and first name that received an award from 2010 to 2017.

```
SELECT
    LastName,
    FirstName
FROM Student
WHERE StudentID IN (SELECT StudentID
                    FROM Student_Award
                    WHERE AwardYear BETWEEN 2010 AND 2017);
```

100 %

Results | Messages

	LastName	FirstName
1	Hinz	Sam
2	Ostrem	Pete
3	Olsen	Russ
4	Johnson	Scott
5	Bell	Keith

5. Write an SQL query that shows the Risk and List Price of activities that the student with the last name 'Hinz' participates in.

```
SELECT
    Risk,
    ListPrice
FROM Activity
WHERE ActivityID IN (SELECT ActivityID
                     FROM Student_Activity
                     WHERE StudentID IN (SELECT StudentID
                     FROM Student
                     WHERE LastName = 'Hinz'));
```

100 %

Results | Messages

	Risk	ListPrice
1	3	35.00
2	3	50.00

6. After reading pages 106 and 107, write the code needed to create a table of numbers from 1 to 10,000.

```
CREATE TABLE dbo.Digits(digit INT NOT NULL PRIMARY KEY);
INSERT INTO dbo.Digits(digit)
VALUES (0), (1), (2), (3), (4), (5), (6), (7), (8), (9);


SELECT D4.digit * 1000 + D3.digit * 100 + D2.digit * 10 + D1.digit + 1 AS Number
FROM dbo.Digits AS D1
CROSS JOIN dbo.Digits AS D2
CROSS JOIN dbo.Digits AS D3
CROSS JOIN dbo.Digits AS D4
ORDER BY Number;
```