

## Lab#2 CS4350 Database Systems

Given the following relational database schema:

Student = (SSN, Name, Major)

Course = ( CourseNumber, PrerequisiteCourseNumber, Course Title, NumberUnits)

Section = ( CourseNumber, Quarter, RoomNumber, DayTime), where DayTime is of the form MW 1:0-2:00PM.

Enrollment = (SSN, CourseNumber, Quarter, Grade)// Grade is either Null or a letter grade.

Express the following queries using appropriate SQL statements with a minimum number of operations:

### Course Table

	CourseNumber	PrerequisiteCourseNumber	CourseTitle	NumberUnits
▶	1	NULL	Formal Languages & Automata	3
	2	1	Database Systems	3
	3	NULL	Software Engineering	3
	4	2	Systems Programming	3
	5	NULL	Data Structures	3

### Enrollment Table

	SSN	CourseNumber	Quarter	Grade
▶	213784218	1	Fall2020	D
	214612384	1	Fall2020	A
	472934828	1	Fall2020	A
	213784218	1	Spring2020	F
	213784218	1	Summer2020	F
	303231274	2	Fall 2020	A
	214612384	2	Fall2020	C
	214612384	3	Spring 2021	NULL
	303231274	3	Summer2021	D
*	NULL	NULL	NULL	NULL

### Section Table

	CourseNumber	Quarter	RoomNumber	DayTime
▶	1	Fall 2020	704	MW12:00PM
	2	Fall2020	404	W4:50PM
	5	Spring 2021	615	T4:00PM
	4	Spring2021	603	M12:00PM
*	NULL	NULL	NULL	NULL

## Student Table

	SSN	Name	Major
▶	213784218	Charlie Smith	Art History
	214612384	Abraham Lincoln	Computer Science
	303231274	Bill Nye	Mechanical Engineering
	472934828	Billy Cobb	Computer Science
*	NULL	NULL	NULL

1. List the name, SSN and the number of courses the student has taken (courses with the same CourseNumber taken in different quarters are counted as different courses).

```

6      # 1
7      SELECT s.Name, s.SSN, COUNT(*) NumberOfCourses
8      FROM Student s, Enrollment e
9      WHERE s.SSN = e.SSN
10     GROUP BY s.Name, s.SSN;

```

Result Grid

	Name	SSN	NumberOfCourses
▶	Charlie Smith	213784218	3
	Abraham Lincoln	214612384	3
	Billy Cobb	472934828	1
	Bill Nye	303231274	2

2. Answer #1 assuming courses with the same CourseNumber taken in different quarters are considered as one course.

```

12     #2
13     SELECT s.Name, s.SSN, COUNT(DISTINCT e.CourseNumber) NumberOfCourses
14     FROM Student s LEFT OUTER JOIN Enrollment e
15     ON s.SSN = e.SSN
16     GROUP BY s.Name, s.SSN;
17

```

Result Grid

	Name	SSN	NumberOfCourses
▶	Abraham Lincoln	214612384	3
	Bill Nye	303231274	2
	Billy Cobb	472934828	1
	Charlie Smith	213784218	1

- List the name, SSN and number of courses the student has taken and completed, i.e., received a letter grade (courses with the same CourseNumber taken in different quarters are counted as different courses)

```

18      #3
19      SELECT s.Name, s.SSN, COUNT(*) CompletedCourses
20      FROM student s, enrollment e
21      WHERE s.SSN = e.SSN
22      AND e.GRADE is NOT NULL
23      GROUP BY s.Name, s.SSN;

```

Name	SSN	NumberOfCourses
Abraham Lincoln	214612384	3
Bill Nye	303231274	2
Billy Cobb	472934828	1
Charlie Smith	213784218	1

- List the name, SSN and number of courses the student has taken and completed with a letter grade C or better (courses with the same CourseNumber taken in different quarters are counted as different courses).

```

25      #4
26      SELECT s.Name, s.SSN, COUNT(*) NumberOfCourses
27      FROM student s, enrollment e
28      WHERE s.SSN = e.SSN
29      AND e.Grade IN (SELECT Grade
30                      FROM Enrollment
31                      WHERE Grade = 'A' OR Grade = 'B' OR Grade = 'C')
32      GROUP BY S.Name, S.SSN;

```

Name	SSN	NumberOfCourses
Abraham Lincoln	214612384	2
Billy Cobb	472934828	1
Bill Nye	303231274	1

- List the Course Title and CourseNumber which does not have any prerequisite.

```

34      #5
35      SELECT CourseTitle, CourseNumber
36      FROM Course
37      WHERE PrerequisiteCourseNumber is NULL;
38

```

Name	SSN	NumberOfCourses
Abraham Lincoln	214612384	2
Billy Cobb	472934828	1
Bill Nye	303231274	1

6. List the name of every student and SSN who earned an A in all courses he or she has completed, i.e., every grade is either Null or an A)

```

39      #6
40      SELECT s.Name, S.SSN
41      FROM Student s, Enrollment e
42      WHERE s.SSN = e.SSN AND
43      s.Name NOT IN (SELECT s1.Name FROM Enrollment e1, Student s1
44      WHERE e1.SSN = s1.SSN
45      AND e1.Grade NOT LIKE 'A');

```

<	Result Grid	Filter Rows:	Export:	Wrap Cell Content:
	Name	SSN	NumberOfCourses	
▶	Abraham Lincoln	214612384	2	
	Billy Cobb	472934828	1	
	Bill Nye	303231274	1	

7. List the name of every student, SSN and the CourseNumber who has taken the course at least three times.

```

47      #7
48      SELECT s.Name, E.SSN, E.CourseNumber
49      FROM Student s, Enrollment e
50      WHERE s.SSN = e.SSN
51      GROUP BY s.Name, E.SSN, e.CourseNumber HAVING COUNT(*) > 2;

```

<	Result Grid	Filter Rows:	Export:	Wrap Cell Content:
	Name	SSN	NumberOfCourses	
▶	Abraham Lincoln	214612384	2	
	Billy Cobb	472934828	1	
	Bill Nye	303231274	1	

8. List the name of every student, SSN, every quarter, and the number of courses he or she has taken in the listed quarter.

```

53      #8
54      SELECT s.Name, S.SSN, e.Quarter, COUNT(*)
55      FROM Student S, Enrollment e
56      WHERE s.SSN = e.SSN
57      GROUP BY s.Name, s.SSN, e.Quarter;

```

<	Result Grid	Filter Rows:	Export:
	Name	SSN	NumberOfCourses
▶	Abraham Lincoln	214612384	2
	Billy Cobb	472934828	1
	Bill Nye	303231274	1

9. List the name of every student and SSN who has not taken any course more than once.

```

59      #9
60      SELECT s.Name, s.SSN
61      FROM Student s, Enrollment e
62      WHERE s.SSN = e.SSN
63      AND e.CourseNumber HAVING COUNT(*) > 1;

```

Result Grid			
Filter Rows: <input type="text"/>			
Export:			
	Name	SSN	NumberOfCourses
▶	Abraham Lincoln	214612384	2
	Billy Cobb	472934828	1
	Bill Nye	303231274	1

10. List the name, SSN, the number of courses the student has taken, and the number of courses completed (courses with the same CourseNumber taken in different quarters are counted as different courses)

```

65      #10
66      SELECT s.Name, s.SSN, COUNT(*) NumberOfCourses, COUNT(Grade) NumberOfCoursesCompleted
67      FROM Student s, Enrollment e
68      WHERE s.SSN = e.SSN
69      GROUP BY s.Name, s.SSN;

```

Result Grid				
Filter Rows: <input type="text"/>				
Export:  Wrap Cell Content:				
	Name	SSN	NumberOfCourses	NumberOfCoursesCompleted
▶	Charlie Smith	213784218	3	3
	Abraham Lincoln	214612384	3	2
	Billy Cobb	472934828	1	1
	Bill Nye	303231274	2	2

11. List every two CourseNumber and their titles which have the same prerequisites.

```

71      #11
72      SELECT PrerequisiteCourseNumber, COUNT(*) SamePrerequisite
73      FROM Course
74      GROUP BY PrerequisiteCourseNumber HAVING SamePrerequisite > 1;

```

Result Grid		
Filter Rows: <input type="text"/>		
Export:  Wrap Cell Content:		
	PrerequisiteCourseNumber	SamePrerequisite
▶	NULL	3
	1	2

12. List the name of every student and SSN who has completed all the courses he/she has taken and earned an A in each course, i.e., every grade is an A and no NULL value

```

76      #12
77      SELECT s.SSN, S.Name, E.CourseNumber
78      FROM Student s INNER JOIN Enrollment e
79      ON s.SSN = e.SSN
80      WHERE Grade = 'A' AND Grade IS NOT NULL;

```

Result Grid		Filter Rows:	Export:
PrerequisiteCourseNumber	SamePrerequisite		
NULL	3		
1	2		

13. List the name of every student and SSN who earned no A in any course.

```

82      #13
83      SELECT s.SSN, S.Name, e.CourseNumber
84      FROM Student s INNER JOIN Enrollment e
85      ON s.SSN = e.SSN
86      WHERE Grade != 'A';
87

```

Result Grid		Filter Rows:	Export:
SSN	Name	CourseNumber	
213784218	Charlie Smith	1	
213784218	Charlie Smith	1	
213784218	Charlie Smith	1	
214612384	Abraham Lincoln	2	
303231274	Bill Nye	3	

14. List the name and major of every student who has only taken courses that meet MW afternoon (12 or after).

```

88      #14
89      SELECT s.Name, s.Major, sec.DayTime
90      FROM Student s INNER JOIN Section sec
91      WHERE sec.DayTime >= 'MW12:00PM'
92      AND sec.DayTime NOT LIKE 'T_____'
93      AND sec.DayTime NOT LIKE 'W_____'
94      AND sec.DayTime NOT LIKE 'F_____' ;
95

```

Result Grid		Filter Rows:	Export:
Name	Major	DayTime	
Charlie Smith	Art History	MW 12:00PM	
Abraham Lincoln	Computer Science	MW 12:00PM	
Bill Nye	Mechanical Engineering	MW 12:00PM	
Billy Cobb	Computer Science	MW 12:00PM	

15. List the name and major of every student who has taken all the courses that meet MW afternoon.

```

96      #15
97 •    SELECT s.Name, s.Major, sec.DayTime
98      FROM Student s INNER JOIN Section sec
99      WHERE sec.DayTime >= 'MW12:00PM'
100     AND sec.DayTime NOT LIKE 'T_____'
101     AND sec.DayTime NOT LIKE 'W_____'
102     AND sec.DayTime NOT LIKE 'F_____' ;
103

```

Name	Major	DayTime
Charlie Smith	Art History	MW12:00PM
Abraham Lincoln	Computer Science	MW12:00PM
Bill Nye	Mechanical Engineering	MW12:00PM
Billy Cobb	Computer Science	MW12:00PM

16. List the name and major of every student who has not taken any course that meets MW afternoon.

```

104      #16
105 •    SELECT s.Name, s.Major, sec.DayTime
106      FROM Student s INNER JOIN Section sec
107      WHERE DayTime < 'MW12:00PM';

```

Name	Major	DayTime
Charlie Smith	Art History	M12:00PM
Abraham Lincoln	Computer Science	M12:00PM
Bill Nye	Mechanical Engineering	M12:00PM
Billy Cobb	Computer Science	M12:00PM

17. List every CourseNumber and Quarter which has the highest enrollment.

```





109      #17
110 •    SELECT CourseNumber, max(Quarter) Highest
111      FROM Enrollment;

```

CourseNumber	Highest
1	Summer2021

18. List every CourseNumber and CourseTitle which has the highest enrollment based on all quarters.





```
113      #18
114 •    SELECT e.CourseNumber, e.Quarter, c.CourseTitle
115      FROM Enrollment e INNER JOIN Course c
116      ON e.CourseNumber = c.CourseNumber
117      WHERE e.Quarter = (SELECT MAX(NumberUnits) FROM Course);
```

<   Filter Rows:  | Export:  | Wrap Cell Content: 

	CourseNumber	Quarter	CourseTitle
--	--------------	---------	-------------

19. List the name and major of every student who has completed the highest number of units.





```
119      #19
120 •    SELECT s.Name, s.Major
121      FROM Student s INNER JOIN Course c
122      WHERE c.NumberUnits = (SELECT MAX(NumberUnits) FROM Course);
```

<   Filter Rows:  | Export:  | Wrap Cell Content: 

	Name	Major
▶	Charlie Smith	Art History
	Charlie Smith	Art History
	Charlie Smith	Art History
	Charlie Smith	Art History
	Charlie Smith	Art History
	Charlie Smith	Art History
	Abraham Lincoln	Computer Science
	Abraham Lincoln	Computer Science
	Abraham Lincoln	Computer Science
	Abraham Lincoln	Computer Science

20. List every Course tile which is a prerequisite for the largest number of courses.

```
124      #20
125 •    SELECT CourseNumber, PrerequisiteCourseNumber, MAX(NumberUnits) LargestNumberOfCourses
126      FROM Course;
```

<   Filter Rows:  | Export:  | Wrap Cell Content: 

	CourseNumber	PrerequisiteCourseNumber	LargestNumberOfCourses
▶	1	NULL	3