RYERSON UNIVERSITY

CIND110
DATA ORGANIZATION FOR DATA ANALYSTS

ASSIGNMENT 2
DESIGN AND MAINTAIN A RELATIONAL DATABASE

SECTION: DK0
SUMBITTED BY: ANN SAM
STUDENT NUMBER: 501160843

Write an SQL statement(s) to find the following:

1. List the names, ids and major of students who are majored in Computer Engineering (CE).

```
255    -- Q1
256 •  SELECT DISTINCT St_Name AS 'StudentName', St_Id AS 'StudentId', Major
257    FROM STUDENT
258    WHERE Major = 'CE';
259
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

StudentName	StudentId	Major
Dara	15013	CE
Alice	16009	CE
Lary	19009	CE
Xue	20102	CE

2. List the names, Ids, major, and GPA of students who have GPA ≥ 3.5.

```
260    -- Q2
261 •  SELECT DISTINCT St_Name AS 'StudentName', St_Id AS 'StudentId', Major, GPA
262    FROM STUDENT
263    WHERE GPA >= 3.5;
264
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

StudentName	StudentId	Major	GPA
Kyle	15010	CS	3.5
Mina	15011	CS	3.6
Ali	15014	INE	3.7
Rona	16005	MATH	3.9
Mary	17011	CS	3.5
Peter	19003	MATH	3.9
Roy	19007	MATH	3.75
Lary	19009	CE	3.85
Arya	20101	Art	3.75

3. List the grades of student ‘David’ with this format:
<StudentName CourseName CourseID SectionID Term Year ProfName Grade>

```
265     -- Q3
266 •   SELECT DISTINCT S.St_Name AS 'StudentName', C.C_Name AS 'CourseName', K.C_Id AS 'CourseId', K.S_Id AS 'SectionId',
267             K.Term, K.Year, K.P_Name AS 'ProfName', G.Grade
268     FROM Section AS K
269     JOIN Course AS C ON C.C_Id = K.C_Id
270     JOIN Grade AS G ON G.S_Id = K.S_Id
271     JOIN Student AS S ON S.St_Id = G.St_Id
272     WHERE S.St_Name = 'David';
273
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

StudentName	CourseName	CourseId	SectionId	Term	Year	ProfName	Grade
David	Data Mining	CIND850	37	Summer	2021	Smith	3.45

4. List the grade of the students who passed the course ID “CPS510” with this format:
<StudentName StudentID SectionID CourseID Term Year ProfName Grade>

```
274     -- Q4
275 •   CREATE VIEW Q4 AS
276     SELECT S.St_Name AS 'StudentName', S.St_Id AS 'StudentId', K.S_Id AS 'SectionId', K.C_Id AS 'CourseId',
277             K.Term, K.Year, K.P_Name AS 'ProfName', G.Grade
278     FROM Section AS K
279     JOIN Course AS C ON C.C_Id = K.C_Id
280     JOIN Grade AS G ON G.S_Id = K.S_Id
281     JOIN Student AS S ON S.St_Id = G.St_Id
282     WHERE C.C_Id = 'CPS510' AND G.Grade >= 1.0;
283
284 •   SELECT * FROM Q4;
285
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

StudentName	StudentId	SectionId	CourseId	Term	Year	ProfName	Grade
Dara	15013	25	CPS510	Summer	2019	Smith	3.95

5. Update the grade of all the students who passed the course ID ‘CPS510’ section ‘25’ to ‘4’ and list the results.

```
286     -- Q5
287 •   UPDATE Q4
288     SET Grade = 4.0
289     WHERE CourseId = 'CPS510' AND SectionId = '25';
290
291 •   SELECT * FROM Q4;
292
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

StudentName	StudentId	SectionId	CourseId	Term	Year	ProfName	Grade
Dara	15013	25	CPS510	Summer	2019	Smith	4

6. List all attributes of students who are enrolled in course ID ‘CPS510’ with Professor ‘Smith’, grouped by the course section.

```
293     -- Q6
294 •  SELECT DISTINCT S.St_Id AS 'StudentId', S.St_Name AS 'StudentName', S.Major, S.Degree, S.GPA, S.Email
295     FROM STUDENT AS S
296     JOIN GRADE AS G
297     ON G.St_Id = S.St_Id
298     JOIN SECTION AS K
299     ON K.S_Id = G.S_Id
300     WHERE K.C_Id = 'CPS510' AND K.P_Name = 'Smith'
301     GROUP BY K.S_Id;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

StudentId	StudentName	Major	Degree	GPA	Email
15013	Dara	CE	B.Eng.	3.1	Dara@

7. List the student ID, student name, course ID and course name for all students enrolled in Fall 2021; sort the output in ascending order by student ID.

```
303     -- Q7
304 •  SELECT S.St_Id AS 'StudentId', S.St_Name AS 'StudentName', C.C_Id AS 'CourseId', C.C_Name AS 'CourseName'
305     FROM STUDENT AS S
306     JOIN GRADE AS G ON G.St_Id = S.St_Id
307     JOIN SECTION AS K ON K.S_Id = G.S_Id
308     JOIN Course AS C ON C.C_Id = K.C_Id
309     WHERE K.Term = 'Fall' AND K.Year = '2021'
310     ORDER BY S.St_Id ASC;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

StudentId	StudentName	CourseId	CourseName
16007	Sam	CPS888	Adv Software Eng
17015	Amir	CIND001	Phyton

8. List all the professor names who teach a course in winter 2022, along with their course names.

```
312     -- Q8
313 •   SELECT K.P_Name AS 'ProfName', C.C_Name AS 'CourseName'
314     FROM Section AS K
315     JOIN Course AS C ON C.C_Id = K.C_Id
316     WHERE K.Term Like 'Winter%' and K.Year = '2022';
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

ProfName	CourseName
Green	Calculus Basic
Black	Intro to Java
White	Adv Web Prog
Black	Intro Data Analysts
Smith	Phyton
White	Phyton
Green	System Devlp

9. List the name, Id, major, degree, and GPA of all students who have a GPA ≥ 3.0 and studying towards degree 'B.Sc.' and 'B.A.'.

```
318     -- Q9
319 •   SELECT S.St_Name AS 'StudentName', S.St_Id AS 'StudentId', S.Major, S.Degree, S.GPA
320     FROM Student AS S
321     WHERE S.Degree = 'B.Sc.' OR S.Degree = 'B.A.'
322     HAVING S.GPA >= 3.0;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

StudentName	StudentId	Major	Degree	GPA
Kyle	15010	CS	B.Sc.	3.5
Mina	15011	CS	B.Sc.	3.6
Lida	15012	Art	B.A.	3.2
Rona	16005	MATH	B.Sc.	3.9
Mary	17011	CS	B.Sc.	3.5
David	17013	CS	B.Sc.	3.4
Amir	17015	Art	B.A.	3.45
Peter	19003	MATH	B.Sc.	3.9
Roy	19007	MATH	B.Sc.	3.75
Arya	20101	Art	B.A.	3.75
Sue	20103	CS	B.Sc.	3.1

10. List the name, Id, major, degree, and GPA of all students who have GPA between 3.0 and 4.0 inclusively.

```
-- Q10
SELECT S.St_Name AS 'StudentName', S.St_Id AS 'StudentId', S.Major, S.Degree, S.GPA
FROM Student AS S
WHERE S.GPA >= 3.0 AND S.GPA <= 4.0;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

StudentName	StudentId	Major	Degree	GPA
Kyle	15010	CS	B.Sc.	3.5
Mina	15011	CS	B.Sc.	3.6
Lida	15012	Art	B.A.	3.2
Dara	15013	CE	B.Eng.	3.1
Ali	15014	INE	B.Eng.	3.7
Rona	16005	MATH	B.Sc.	3.9
Sam	16007	INE	B.Eng.	3
Mary	17011	CS	B.Sc.	3.5
David	17013	CS	B.Sc.	3.4
Amir	17015	Art	B.A.	3.45
Peter	19003	MATH	B.Sc.	3.9
Roy	19007	MATH	B.Sc.	3.75
Kate	19008	INE	B.Eng.	3.3
Lary	19009	CE	B.Eng.	3.85
Arya	20101	Art	B.A.	3.75
Xue	20102	CE	B.Eng.	3
Sue	20103	CS	B.Sc.	3.1

11. List all attributes of students who are enrolled in course currently along with the term and year, grouped by the course name.

```
-- Q11
SELECT S.St_Id AS 'StudentId', S.St_Name AS 'StudentName', S.Major, S.Degree, S.GPA, S.Email, K.Term, K.Year
FROM Student AS S
JOIN Grade AS G ON G.St_Id = S.St_Id
JOIN Section AS K ON K.S_Id = G.S_Id
JOIN Course AS C ON C.C_Id = K.C_Id
WHERE K.Term LIKE 'Winter%' AND K.Year = '2022'
GROUP BY C.C_Name;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

StudentId	StudentName	Major	Degree	GPA	Email	Term	Year
20101	Arya	Art	B.A.	3.75	Arya@	Winter	2022
20100	Lili	Art	B.A.	2.9	Lili@	Winter	2022
16005	Rona	MATH	B.Sc.	3.9	Rona@	Winter	2022
17017	Tara	Art	B.A.	2.8	Tara@	Winter	2022
19003	Peter	MATH	B.Sc.	3.9	Peter@	Winter	2022

12. List the name, email, and phone of all Professors who tech in the current semester along with the course name, course id, term and year.

```
338     -- Q12
339 •  SELECT P.P_Name AS 'ProfName', P.P_Email AS 'ProfEmail', P.P_Phone AS 'ProfPhone', C.C_Name AS 'CourseName',
340             C.C_Id AS 'CourseId', K.Term, K.Year
341     FROM Professor AS P
342     JOIN Section AS K ON K.P_Name = P.P_Name
343     JOIN Course AS C ON C.C_Id = K.C_Id
344     WHERE K.Term Like 'Winter%' AND K.Year = '2022';
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

ProfName	ProfEmail	ProfPhone	CourseName	CourseId	Term	Year
Green	Green@	8778	Calculus Basic	MATH101	Winter	2022
Black	Black@	3443	Intro to Java	CPS209	Winter	2022
White	White@	2332	Adv Web Prog	CPS630	Winter	2022
Black	Black@	3443	Intro Data Analysts	CIND110	Winter	2022
Smith	Smith@	9889	Phyton	CIND001	Winter	2022
White	White@	2332	Phyton	CIND001	Winter	2022
Green	Green@	8778	System Devlp	CITM430	Winter	2022

13. List the name of all Professors who can teach ‘CIND110’ (from the Teach-Pool) along with the course Id, but haven’t taught the course (from Section).

```
346     -- Q13
347 •  SELECT DISTINCT TP.P_Name AS 'ProfName', C.C_Name AS 'CourseName', C.C_Id AS 'CourseId'
348     FROM Teach_Pool AS TP
349     JOIN Course AS C ON C.C_Id = TP.C_Id
350     JOIN Section AS K ON K.C_Id = TP.C_Id
351     WHERE TP.C_Id = 'CIND110' AND NOT TP.P_Name = K.P_Name;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

ProfName	CourseName	CourseId
Adam	Intro Data Analysts	CIND110
Green	Intro Data Analysts	CIND110

14. List the name of all Professors who are available to teach in summer terms along with the course Id they can teach (from Teach-Pool and are not teaching the same course currently.

```
353     -- Q14
354 •  CREATE VIEW Q14a AS
355     SELECT DISTINCT TP.P_Name AS 'ProfName', TP.C_Id AS 'CourseId'
356     FROM Teach_Pool AS TP
357     WHERE TP.Term LIKE 'Summer%';
358
359 •  CREATE VIEW Q14b AS
360     SELECT DISTINCT K.P_Name AS 'ProfName', K.C_Id AS 'CourseID'
361     FROM Section AS K
362     WHERE K.Term LIKE 'Winter%' AND K.Year = '2022';
363
364 •  SELECT *
365     FROM Q14a
366     WHERE NOT EXISTS (SELECT * FROM Q14b
367                       WHERE Q14a.ProfName = Q14b.ProfName AND Q14a.CourseId = Q14b.CourseID);
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

ProfName	CourseId
Green	CIND110
Red	CIND800
Gray	CPS209
Gray	CPS406

15. List the name and Id of all courses that are introductory courses (their names begin with either ‘Intro’ or ‘Into’).

```
369     -- Q15
370 •  SELECT DISTINCT C.C_name AS 'CourseName', C.C_Id AS 'CourseId'
371     FROM Course AS C
372     WHERE C.C_Name LIKE 'Intro%' OR C.C_Name LIKE 'Into%';
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

CourseName	CourseId
Intro Data Analysts	CIND110
Intro to Java	CPS209
Intro Software Eng	CPS406
Into Database	CPS510
Intro to Web Prog	CPS530

16. List all course names and Ids which have a pre-requisite of “CPS209” along with the term and year they have been provided.

```
374     -- Q16
375 •   SELECT C.C_Name AS 'CourseName', C.C_Id AS 'CourseId', K.Term, K.Year
376     FROM Course AS C
377     JOIN Section AS K ON K.C_Id = C.C_Id
378     WHERE C.C_Prerequisite = 'CPS209';
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

CourseName	CourseId	Term	Year
System Design	CITM305	Summer	2021
Intro Software Eng	CPS406	Fall	2019
Intro Software Eng	CPS406	Summer	2019
Into Database	CPS510	Fall	2019
Into Database	CPS510	Summer	2019
Intro to Web Prog	CPS530	Fall	2020
Intro to Web Prog	CPS530	Summer	2020

17. Create a View as ‘Report1’ from all attributes of students who were enrolled in the Summer Term of 2021, along with their course Id, term and year; Display Report1.

```
380     -- Q17
381 •   CREATE VIEW Report1 AS
382     SELECT  S.St_Id AS 'StudentId', S.St_Name AS 'StudentName', S.Major, S.Degree, S.GPA,
383             S.Email, K.C_Id AS 'CourseId', K.Term, K.Year
384     FROM Student AS S
385     JOIN Grade AS G ON G.St_Id = S.St_Id
386     JOIN Section AS K ON K.S_Id = G.S_Id
387     WHERE K.Term = 'Summer' AND K.Year = '2021';
388
389 •   SELECT * FROM Report1;
390
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

StudentId	StudentName	Major	Degree	GPA	Email	CourseId	Term	Year
15011	Mina	CS	B.Sc.	3.6	Mina@	MATH300	Summer	2021
20102	Xue	CE	B.Eng.	3	Xue@	CITM305	Summer	2021
17013	David	CS	B.Sc.	3.4	David@	CIND850	Summer	2021

18. Create a view as ‘Report_S2022’ from all attributes of potential Professors who will be available to teach in Summer 2022 along with their course Ids, sort the output in ascending order based on names; Display Report_S2022.

```
391     -- Q18
392 •  CREATE VIEW Report_S2022 AS
393     SELECT  P.P_Name AS 'ProfName', P.P_Phone AS 'ProfNumber', P.P_Email AS 'ProfEmail',
394             TP.C_Id AS 'CourseId'
395     FROM Professor AS P
396     JOIN Teach_Pool AS TP ON TP.P_Name = P.P_Name
397     WHERE TP.Term = 'Summer'
398     ORDER BY P.P_Name ASC;
399
400 •  SELECT * FROM Report_S2022;
401
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

ProfName	ProfNumber	ProfEmail	CourseId
Gray	5665	Gray@	CPS209
Gray	5665	Gray@	CPS406
Green	8778	Green@	CIND110
Red	3773	Red@	CIND800

19. Make a list containing all courses’ attributes from ‘Course’ table and their relevant exam types from ‘Teach’ table based on the same course Id.

```
402     -- Q19
403 •  SELECT C.C_Id AS 'CourseId', C.C_name AS 'CourseName', C.C_Credit AS 'CourseCredit',
404                     C.C_Prerequisite AS 'Prerequisite', C.Department AS 'Department', T.ExamType
405     FROM Course AS C
406     JOIN Teach AS T ON T.C_Id = C.C_Id
407     GROUP BY C.C_Id;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

CourseId	CourseName	CourseCredit	Prerequisite	Department	ExamType
CIND001	Phyton	2	NO	IND	Labs
CIND110	Intro Data Analysts	3	CIND001	IND	Labs
CIND800	Adv Data Analysts	3	CIND110	IND	WTest
CIND850	Data Mining	4	CIND800	IND	Proj
CITM305	System Design	3	CPS209	TRSM	WTest
CITM430	System Devlp	4	CITM305	TRSM	WTest
CPS209	Intro to Java	3	NO	CS	Proj
CPS406	Intro Software Eng	3	CPS209	CS	WTest
CPS510	Into Database	3	CPS209	CS	Proj
CPS530	Intro to Web Prog	3	CPS209	CS	Labs
CPS630	Adv Web Prog	4	CPS530	CS	Proj
CPS888	Adv Software Eng	4	CPS406	CS	Proj
MATH101	Calculus Basic	3	NO	MATH	WTest
MATH300	Calculus Inter	3	MATH101	MATH	Labs
MATH400	Calculus Adv	4	MATH300	MATH	Proj

20. Find the Minimum, Maximum, Average, Variance, and Standard Deviation of the GPA of all Students and display them.

```
409     -- 20
410 •  SELECT  MIN(S.GPA) AS 'Minimum GPA', MAX(S.GPA) AS 'Maximum GPA', AVG(S.GPA) AS 'Average GPA',
411            VARIANCE(S.GPA) AS 'Variance GPA', STDDEV(S.GPA) AS 'Standard Deviation GPA'
412     FROM Student AS S;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Minimum GPA	Maximum GPA	Average GPA	Variance GPA	Standard Deviation GPA
2.8	3.9	3.38400000333786	0.12156400765037953	0.3486603040932241

EER Diagram for RegistrarDB