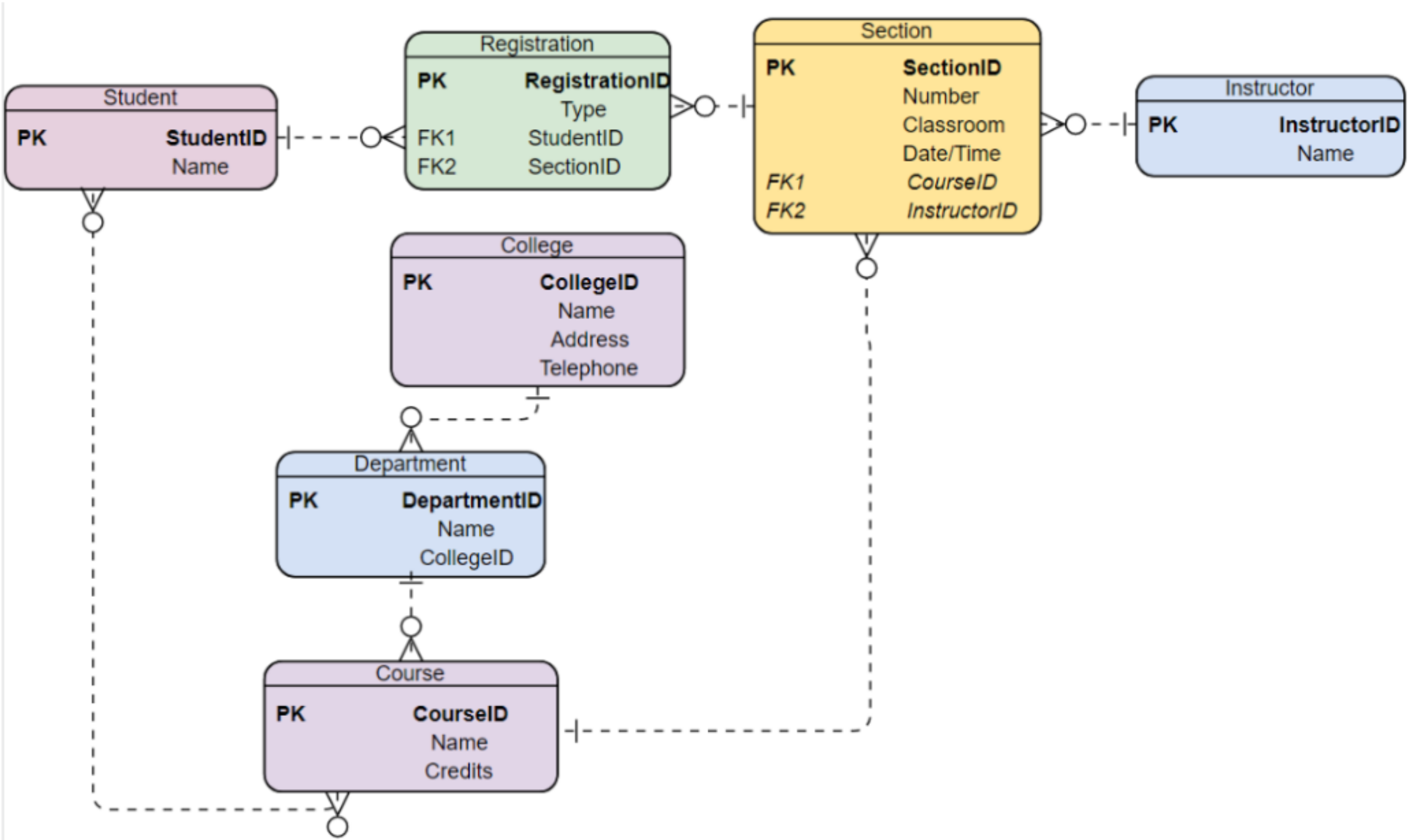
course registration database.

Shansi Dong

A20466369


Student
PK StudentID
Name
Registration
PK RegistrationID
Type
FK1 StudentID
FK2 SectionID
Section
PK SectionID
Number
Classroom
Date/Time
FK1 CourseID
FK2 InstructorID
Instructor
PK InstructorID
Name
College
PK CollegeID
Name
Address
Telephone
Department
PK DepartmentID
Name
CollegeID
Course
PK CourseID
Name
Credits

college

General | Columns | Advanced | Constraints | Parameters | Security | SQL

Inherited from table(s) Select to inherit from...

Columns +

		Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?
✎	🗑	collegeid	integer			Yes	Yes
✎	🗑	collegename	character varying			Yes	No
✎	🗑	address	character varying			No	No
✎	🗑	telephone	integer			No	No

course

General | Columns | Advanced | Constraints | Parameters | Security | SQL

Inherited from table(s) Select to inherit from...

Columns +

		Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?
✎	🗑	courseid	integer			Yes	Yes
✎	🗑	credit	integer			Yes	No
✎	🗑	coursename	character varying	50		Yes	No

department

General | Columns | Advanced | Constraints | Parameters | Security | SQL

Inherited from table(s): Select to inherit from...

Columns

		Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?
✎	🗑	departmentid	character varying	20		Yes	Yes
✎	🗑	deptname	character varying	50		Yes	No
✎	🗑	collegeid	integer			Yes	No

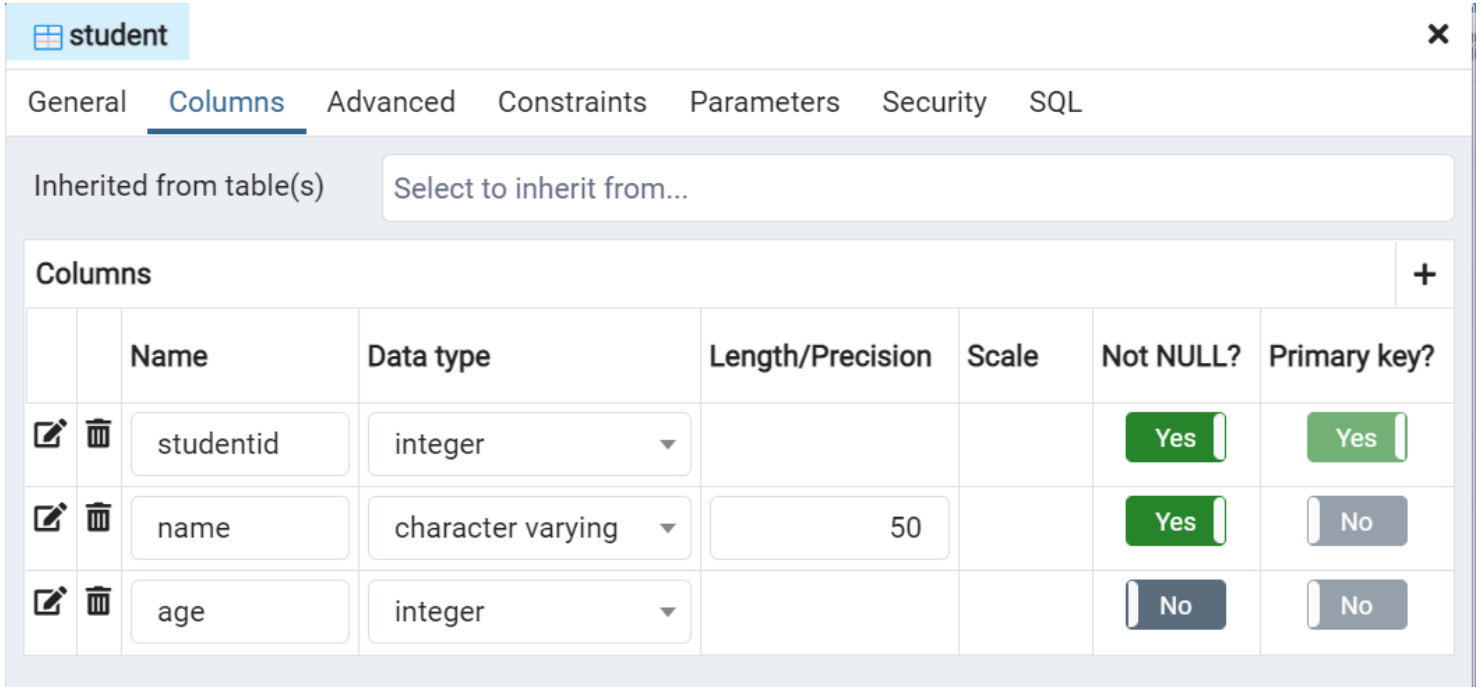
instructor

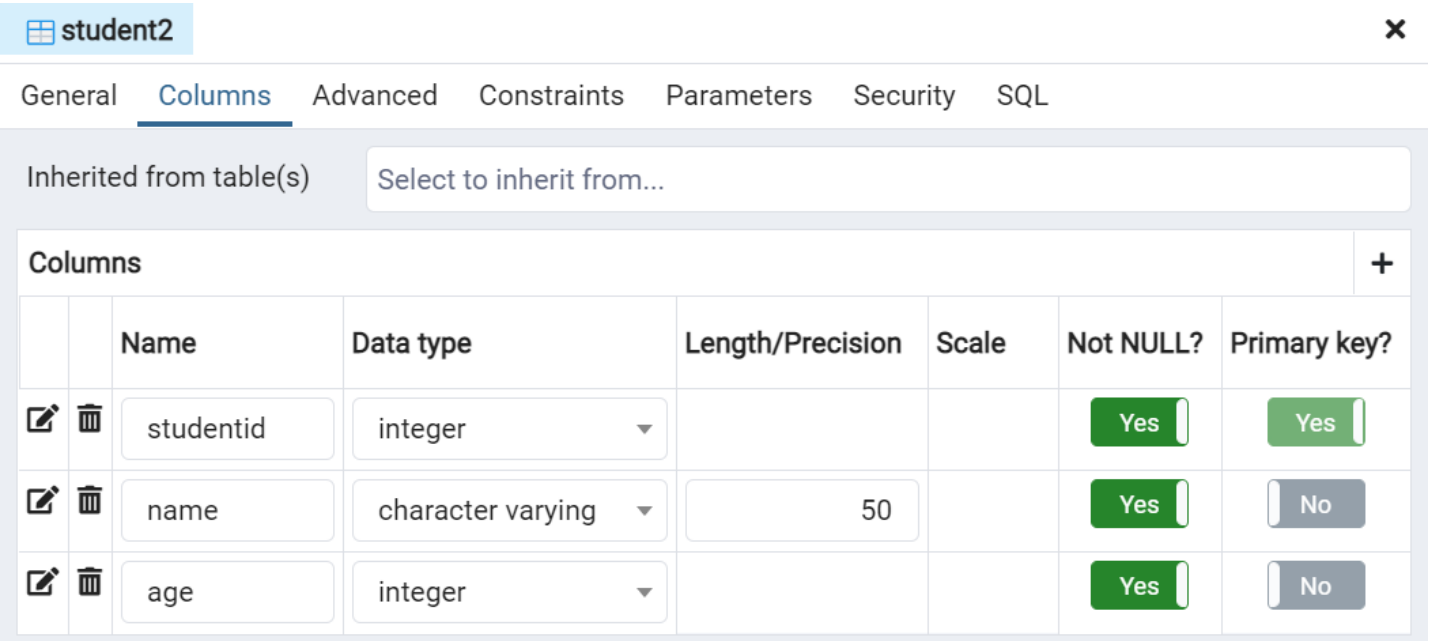
General | Columns | Advanced | Constraints | Parameters | Security | SQL

Inherited from table(s): Select to inherit from...

Columns

		Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?
✎	🗑	instructorid	integer			Yes	Yes
✎	🗑	name	character varying	20		Yes	No
✎	🗑	age	integer			Yes	No


student
General
Columns
Advanced
Constraints
Parameters
Security
SQL
Inherited from table(s)
Select to inherit from...
Columns
Name
Data type
Length/Precision
Scale
Not NULL?
Primary key?
studentid
integer
Yes
Yes
name
character varying
50
Yes
No
age
integer
No
No


student2
General
Columns
Advanced
Constraints
Parameters
Security
SQL
Inherited from table(s)
Select to inherit from...
Columns
Name
Data type
Length/Precision
Scale
Not NULL?
Primary key?
studentid
integer
Yes
Yes
name
character varying
50
Yes
No
age
integer
Yes
No

1. Create at least 10 different subqueries (with problem statement what to query) to search data using a table

1).What are all the departments from College of Computing?

Query Editor | Query History

```
1  SELECT*FROM department WHERE collegeid IN(SELECT collegeid FROM college
2  WHERE collegename = 'College of Computing');
```

Data Output | Explain | Messages | Notifications

	departmentid [PK] character varying (20)	deptname character varying (50)	collegeid integer
1	CS	Computer Science	101
2	IT	Information Technology	101
3	AM	Applied Mathematics	101

2). Using a subquery, list all the students with ages above the average.

Query Editor | Query History

```
1  SELECT*FROM student WHERE age > (SELECT AVG(age) FROM student);
2
3
```

Data Output | Explain | Messages | Notifications

	studentid [PK] integer	name character varying (50)	age integer
1	4	Peter	21
2	5	Reid	23
3	6	Creighton	24
4	7	Ryan	22

3). Using a subquery, list courses with credits larger than Database Organization.

Query Editor | Query History

```
1  select * from course where
2  credit > (select credit from course where coursename = 'Database Organization');
3
```

Data Output | Explain | Messages | Notifications

	courseid [PK] integer	credit integer	coursename character varying (50)
1	2	3	Data Structure
2	6	3	Software Engineering
3	9	3	Advanced OS
4	4	4	Computer Architecture
5	5	5	Operating Systems
6	8	4	Theory of Computation

4). Using a subquery, to see if a student ages 29 exists within the student table:

Query Editor | Query History

```
1  SELECT 29 IN (SELECT age FROM student) as result;
```

Data Output | Explain | Messages | Notifications

	result boolean
1	false

5). Using a subquery, to see if a student ages 24 exists within the student table:

Query Editor | Query History

```
1  SELECT 24 IN (SELECT age FROM student) as result;
```

Data Output | Explain | Messages | Notifications

	result boolean
1	true

6). Using a subquery, to check if 2 or more students have the same name.

Query Editor | Query History

```
1  SELECT*FROM student E WHERE 1 <(SELECT COUNT(*) FROM student where name=E.name);
```

Data Output | Explain | Messages | Notifications

	studentid [PK] integer	name character varying (50)	age integer
1	4	Peter	21
2	10	Peter	20

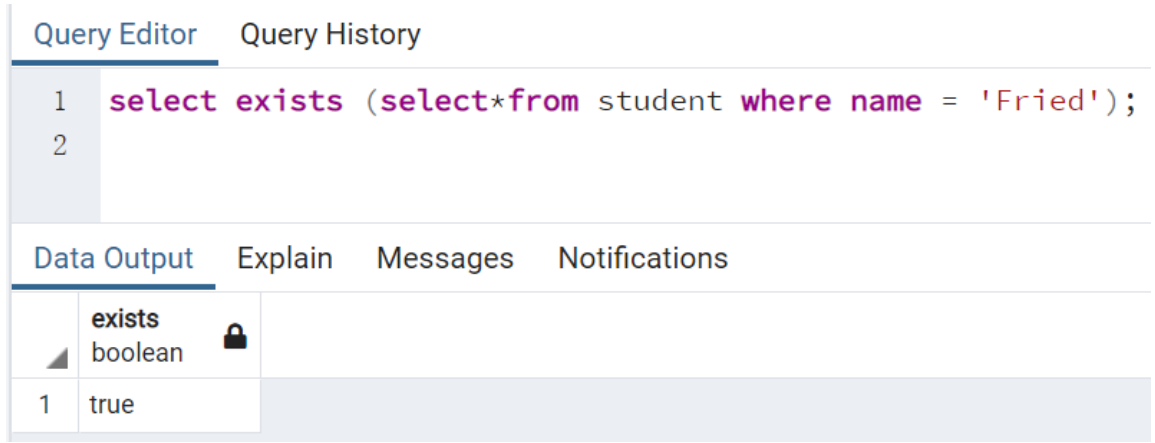
7). Using a subquery, to see if there’re any instructors with ages above 55 exist.

```
1  SELECT EXISTS(SELECT*FROM instructor WHERE age>55) as result;
2
3
```

Data Output | Explain | Messages | Notifications

	result boolean
1	false

8). Using a subquery, to see if there’re any instructors with name Fried exist.



9). Using a subquery, list all the instructors whose ages above all students' ages.

Query Editor | Query History

```
1  select * from instructor
2  where age >= all(select age From  student)
```

Data Output | Explain | Messages | Notifications

	instructorid [PK] integer	name character varying (20)	age integer
1	1	George	34
2	2	Alice	36
3	3	Eric	42
4	4	Jonathan	37
5	5	Ray	48
6	6	Stevens	51
7	7	Leo	48
8	8	Daniel	39
9	9	Cathie	33
10	10	Jacob	41

10). Using a subquery, delete all the students with ages above 20 from student2(which is a copy of table student)

```
1  DELETE FROM STUDENT2 WHERE AGE IN (SELECT AGE FROM STUDENT
2     WHERE AGE > 20 );
3  SELECT*FROM STUDENT2;
```

Data Output | Explain | Messages | Notifications

	studentid [PK] integer	name character varying (50)	age integer
1	1	Fried	18
2	2	Tim	19
3	3	Tom	20
4	8	Martin	20
5	9	Linda	19
6	10	Peter	20

2. If you can get the same result without using subqueries as item1, use the alternative queries and results.

1).What are all the departments from College of Computing?

Query Editor | Query History

```
1  SELECT department.* FROM department INNER JOIN college
2  ON department.collegeid = college.collegeid
3  WHERE college.collegename = 'College of Computing';
```

Data Output | Explain | Messages | Notifications

	departmentid [PK] character varying (20)	deptname character varying (50)	collegeid integer
1	CS	Computer Science	101
2	IT	Information Technology	101
3	AM	Applied Mathematics	101

2. If you can get the same result without using subqueries as item1, use the alternative queries and results.

1).What are all the departments from College of Computing?

Query Editor | Query History

```
1  SELECT department.* FROM department INNER JOIN college
2  ON department.collegeid = college.collegeid
3  WHERE college.collegename = 'College of Computing';
```

Data Output | Explain | Messages | Notifications

	departmentid [PK] character varying (20)	deptname character varying (50)	collegeid integer
1	CS	Computer Science	101
2	IT	Information Technology	101
3	AM	Applied Mathematics	101