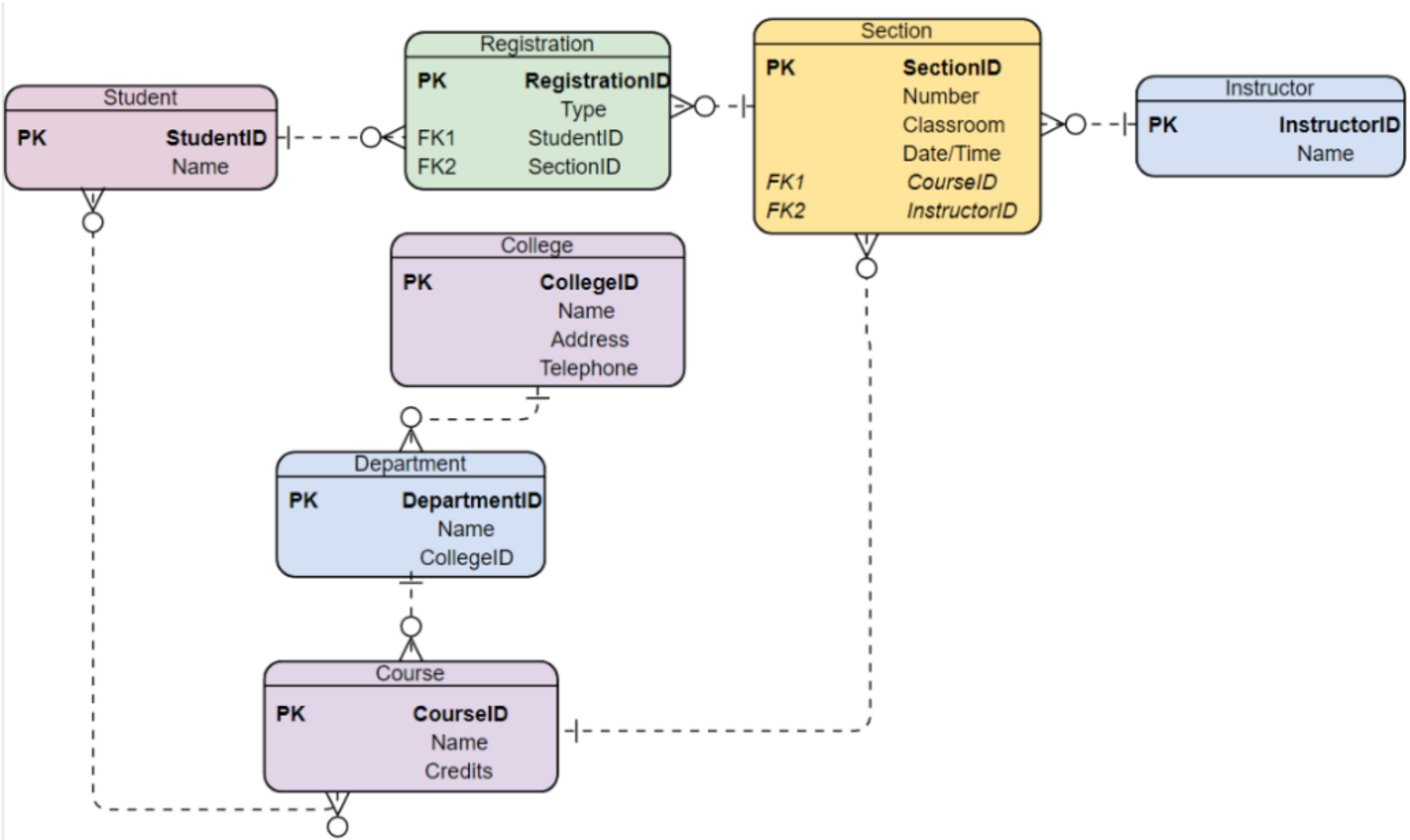
# course registration database.

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

## 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 |

## 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 |
| ✎ | 🗑 | credits | integer | | | Yes | No |
| ✎ | 🗑 | coursename | character varying | 50 | | 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? |
|---|---|---|---|---|---|---|---|
| ✎ | 🗑 | Instructor ID | integer | | | Yes | Yes |
| ✎ | 🗑 | Name | character varying | 20 | | Yes | No |
| ✎ | 🗑 | Age | integer | | | Yes | No |

# 1. Create at least one or two tables of the HW2 design

```
CREATE TABLE student(
   studentid INT PRIMARY KEY      NOT NULL,
    name           VARCHAR(50) NOT NULL
);
CREATE TABLE course(
   courseid INT PRIMARY KEY      NOT NULL,
   coursename           VARCHAR(50) NOT NULL,
   credits        INT      NOT NULL
);
```

# 2. Insert at least 10 records (rows) into each table using SQL queries.

Query Editor | Query History

```
1  INSERT INTO public.student(
2      studentid, name)
3      VALUES (1,'Fried'),
4      (2,'Tim'),
5      (3,'Tom'),
6      (4,'Peter'),
7      (5,'Reid'),
8      (6,'Creighton'),
9      (7,'Ryan'),
10     (8,'Martin'),
11     (9,'Linda'),
12     (10,'Anna');
```

Query Editor | Query History

```
1  select * from student
2  ORDER BY studentid ASC
```

Data Output | Explain | Messages | Notifications

| | studentid [PK] integer | name character varying |
|---|---|---|
| 1 | 1 | Fried |
| 2 | 2 | Tim |
| 3 | 3 | Tom |
| 4 | 4 | Peter |
| 5 | 5 | Reid |
| 6 | 6 | Creighton |
| 7 | 7 | Ryan |
| 8 | 8 | Martin |
| 9 | 9 | Linda |
| 10 | 10 | Anna |

# 2. Insert at least 10 records (rows) into each table using SQL queries.

Query Editor | Query History

```
1  INSERT INTO public.course(
2      courseid, coursename, credits)
3      VALUES (1, 'Introduction to Algorithms' , 3),
4      (2, 'Data Structure' , 3),
5      (3, 'Database organization' , 3),
6      (4, 'Computer Architecture' , 3),
7      (5, 'Operating Systems' , 3),
8      (6, 'Software Engineering' , 3),
9      (7, 'Data Integration' , 3),
10     (8, 'Theory of Computation' , 3),
11     (9, 'Advanced OS' , 3),
12     (10, 'Advanced Database Organization', 3);
13
```

Query Editor | Query History

```
1  select*from course
2  order by courseid ASC
3
```

Data Output | Explain | Messages | Notifications

| | courseid [PK] integer | coursename character varying (50) | credits integer |
|---|---|---|---|
| 1 | 1 | Introduction to Algorithms | 3 |
| 2 | 2 | Data Structure | 3 |
| 3 | 3 | Database organization | 3 |
| 4 | 4 | Computer Architecture | 3 |
| 5 | 5 | Operating Systems | 3 |
| 6 | 6 | Software Engineering | 3 |
| 7 | 7 | Data Integration | 3 |
| 8 | 8 | Theory of Computation | 3 |
| 9 | 9 | Advanced OS | 3 |
| 10 | 10 | Advanced Database Organiz... | 3 |

✔ Successfully

# 3. Modify (add and delete) columns of a table using SQL queries

Query Editor | Query History

```
1  alter table course add enrolldate character varying(50);
2  select*from course
3  order by courseid ASC
```

Data Output | Explain | Messages | Notifications

| | courseid [PK] integer | coursename character varying (50) | credits integer | enrolldate character varying (50) |
|---|---|---|---|---|
| 1 | 1 | Introduction to Algorithms | 3 | [null] |
| 2 | 2 | Data Structure | 3 | [null] |
| 3 | 3 | Database organization | 3 | [null] |
| 4 | 4 | Computer Architecture | 3 | [null] |
| 5 | 5 | Operating Systems | 3 | [null] |
| 6 | 6 | Software Engineering | 3 | [null] |
| 7 | 7 | Data Integration | 3 | [null] |
| 8 | 8 | Theory of Computation | 3 | [null] |
| 9 | 9 | Advanced OS | 3 | [null] |
| 10 | 10 | Advanced Database Organiz... | 3 | [null] |

Query Editor | Query History

```
1  alter table course drop enrolldate;
2  select*from course
3  order by courseid ASC
```

Data Output | Explain | Messages | Notifications

| | courseid [PK] integer | coursename character varying (50) | credits integer |
|---|---|---|---|
| 1 | 1 | Introduction to Algorithms | 3 |
| 2 | 2 | Data Structure | 3 |
| 3 | 3 | Database organization | 3 |
| 4 | 4 | Computer Architecture | 3 |
| 5 | 5 | Operating Systems | 3 |
| 6 | 6 | Software Engineering | 3 |
| 7 | 7 | Data Integration | 3 |
| 8 | 8 | Theory of Computation | 3 |
| 9 | 9 | Advanced OS | 3 |
| 10 | 10 | Advanced Database Organiz... | 3 |

✔ Successfully run. Total

4. Delete a column of a table using SQL queries.
5. Restore the deleted column to the table by item 4 using SQL queries.

Query Editor | Query History

```
1  create table courseidname(
2      courseid Integer, coursename character varying(50));
3  insert into courseidname(
4      select courseid, coursename from course);
5  alter table course drop coursename;
6  select*from course;
```

Data Output | Explain | Messages | Notifications

| | courseid [PK] integer | credits integer |
|---|---|---|
| 1 | 1 | 3 |
| 2 | 2 | 3 |
| 3 | 3 | 3 |
| 4 | 4 | 3 |
| 5 | 5 | 3 |
| 6 | 6 | 3 |
| 7 | 7 | 3 |
| 8 | 8 | 3 |
| 9 | 9 | 3 |
| 10 | 10 | 3 |

4. Delete a column of a table using SQL queries.
5. Restore the deleted column to the table by item 4 using SQL queries.

Query Editor | Query History

```
1  alter table course add coursename character varying(50);
2  update course
3  set coursename =(select courseidname.coursename from courseidname
4      where courseidname.courseid= course.courseid);
5  select*from course;
```

Data Output | Explain | Messages | Notifications

| | courseid [PK] integer | credits integer | coursename character varying (50) |
|---|---|---|---|
| 1 | 1 | 3 | Introduction to Algorithms |
| 2 | 2 | 3 | Data Structure |
| 3 | 3 | 3 | Database organization |
| 4 | 4 | 3 | Computer Architecture |
| 5 | 5 | 3 | Operating Systems |
| 6 | 6 | 3 | Software Engineering |
| 7 | 7 | 3 | Data Integration |
| 8 | 8 | 3 | Theory of Computation |
| 9 | 9 | 3 | Advanced OS |
| 10 | 10 | 3 | Advanced Database Organiz... |

6. Remove one table using SQL queries.
7. Restore the removed table by item 6 using SQL queries.

Query Editor | Query History

```
1  create table student2(studentid INT PRIMARY KEY  NOT NULL,
2                       name  character varying(20) NOT NULL);
3  insert into student2(select studentid, name from student);
4  DROP TABLE student;
5  select*from student2
```

```
CREATE TABLE student(
   studentid INT PRIMARY KEY      NOT NULL,
    name           VARCHAR(50) NOT NULL);
```

Query Editor | Query History

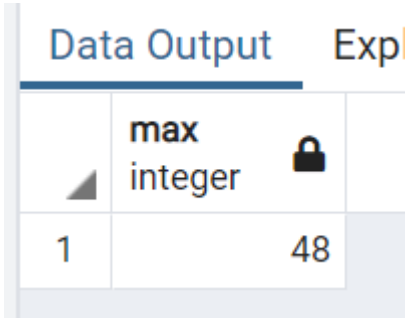
```
1  insert into student(studentid,name)
2  select studentid as studentid, name as name from student2
3  SELECT*FROM student
```

Data Output | Explain | Messages | Notifications

| | studentid [PK] integer | name character varying (50) |
|---|---|---|
| 1 | 1 | Fried |
| 2 | 2 | Tim |
| 3 | 3 | Tom |
| 4 | 4 | Peter |
| 5 | 5 | Reid |
| 6 | 6 | Creighton |
| 7 | 7 | Ryan |
| 8 | 8 | Martin |
| 9 | 9 | Linda |
| 10 | 10 | Anna |

# 8. Create at least 10 different queries to search data using a table.

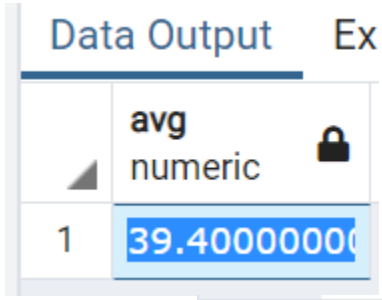
1. SELECT* FROM student;
2. SELECT *FROM course WHERE credit >= 2;
3. SELECT DISTINCT courseid，coursename FROM course;
4. SELECT name,COUNT(name) FROM STUDENT GROUP BY name;
5. SELECT name,age FROM instructor WHERE ORDER BY GRADE DESC LIMIT 1;
6. SELECT AVG(age) FROM instructor;
7. SELECT MAX(age) FROM instructor;

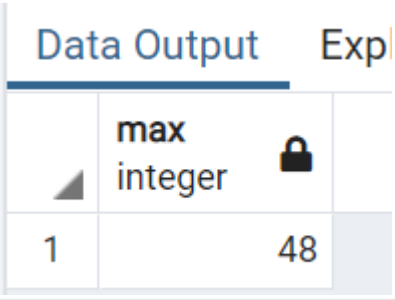


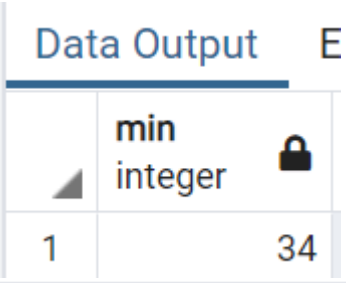

8. SELECT MIN(age) FROM instructor;
9. select length(name) from student;
10. SELECT *FROM instructor WHERE age BETWEEN 35 AND 45;

# 8. Create at least 10 different queries to search data using a table.

1. SELECT* FROM student;
2. SELECT *FROM course WHERE credit >= 2;
3. SELECT DISTINCT courseid，coursename FROM course;
4. SELECT name,COUNT(name) FROM STUDENT GROUP BY name;
5. SELECT name,age FROM instructor WHERE ORDER BY GRADE DESC LIMIT 1;
6. SELECT AVG(age) FROM instructor;
7. SELECT MAX(age) FROM instructor;




8. SELECT MIN(age) FROM instructor;

9. select length(name) from student;
10. SELECT *FROM instructor WHERE age BETWEEN 35 AI

Data Output | Explain | Messages | Notifications

| | instructorid [PK] integer | name character varying (20) | age integer |
|---|---|---|---|
| 1 | 2 | Alice | 36 |
| 2 | 3 | Eric | 42 |
| 3 | 4 | Jonathan | 37 |

# 8. Create at least 10 different queries to search data using a table.

1. SELECT* FROM student;

Data Output | Explain | Messages | Notifications

| | studentid [PK] integer | name character varying (50) |
|---|---|---|
| 1 | 1 | Fried |
| 2 | 2 | Tim |
| 3 | 3 | Tom |
| 4 | 4 | Peter |
| 5 | 5 | Reid |
| 6 | 6 | Creighton |
| 7 | 7 | Ryan |
| 8 | 8 | Martin |
| 9 | 9 | Linda |
| 10 | 10 | Anna |

# 8. Create at least 10 different queries to search data using a table.

2. SELECT *FROM course WHERE credits >= 2;

Data Output | Explain | Messages | Notifications

| | courseid [PK] integer | credits integer | coursename character varying (50) |
|---|---|---|---|
| 1 | 1 | 3 | Introduction to Algorithms |
| 2 | 2 | 3 | Data Structure |
| 3 | 3 | 3 | Database organization |
| 4 | 4 | 3 | Computer Architecture |
| 5 | 5 | 3 | Operating Systems |
| 6 | 6 | 3 | Software Engineering |
| 7 | 7 | 3 | Data Integration |
| 8 | 8 | 3 | Theory of Computation |
| 9 | 9 | 3 | Advanced OS |
| 10 | 10 | 3 | Advanced Database Organiz... |

# 8. Create at least 10 different queries to search data using a table.

3.SELECT DISTINCT courseid,coursename FROM course;

Data Output | Explain | Messages | Notifications

| | courseid [PK] integer | coursename character varying (50) |
|---|---|---|
| 1 | 10 | Advanced Database Organiz... |
| 2 | 5 | Operating Systems |
| 3 | 8 | Theory of Computation |
| 4 | 3 | Database organization |
| 5 | 6 | Software Engineering |
| 6 | 7 | Data Integration |
| 7 | 2 | Data Structure |
| 8 | 9 | Advanced OS |
| 9 | 1 | Introduction to Algorithms |
| 10 | 4 | Computer Architecture |

# 8. Create at least 10 different queries to search data using a table.

4.SELECT name,COUNT(name) FROM student GROUP BY name;

Data Output | Explain | Messages | Notifica

| | name<br>character varying (50) | count<br>bigint |
|---|---|---|
| 1 | Tim | 1 |
| 2 | Tom | 1 |
| 3 | Anna | 1 |
| 4 | Peter | 1 |
| 5 | Martin | 1 |
| 6 | Creighton | 1 |
| 7 | Linda | 1 |
| 8 | Fried | 1 |
| 9 | Ryan | 1 |
| 10 | Reid | 1 |

# 8. Create at least 10 different queries to search data using a table.

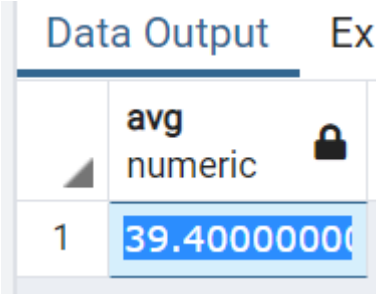
5.SELECT name,age FROM instructor ORDER BY instructorid DESC;

Data Output | Explain | Messages | Notificatio

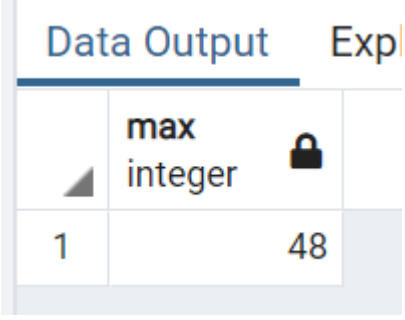
| | name<br>character varying (20) | age<br>integer |
|---|---|---|
| 1 | Ray | 48 |
| 2 | Jonathan | 37 |
| 3 | Eric | 42 |
| 4 | Alice | 36 |
| 5 | George | 34 |

# 8. Create at least 10 different queries to search data using a table.

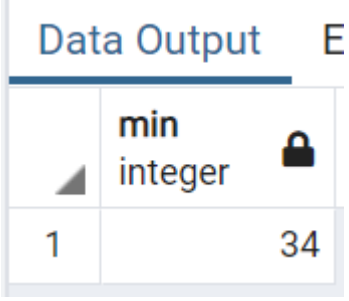
6.SELECT AVG(age) FROM instructor;




7.SELECT MAX(age) FROM instructor;




8.SELECT MIN(age) FROM instructor;



# 8. Create at least 10 different queries to search data using a table.

9.select length(name) from student;

Data Output | Expl

| | length integer |
|---|---|
| 1 | 5 |
| 2 | 3 |
| 3 | 3 |
| 4 | 5 |
| 5 | 4 |
| 6 | 9 |
| 7 | 4 |
| 8 | 6 |
| 9 | 5 |
| 10 | 4 |

10.SELECT *FROM instructor WHERE age BETWEEN 35 AND 45;

Data Output | Explain | Messages | Notifications

| | instructorid [PK] integer | name character varying (20) | age integer |
|---|---|---|---|
| 1 | 2 | Alice | 36 |
| 2 | 3 | Eric | 42 |
| 3 | 4 | Jonathan | 37 |
| | | | |