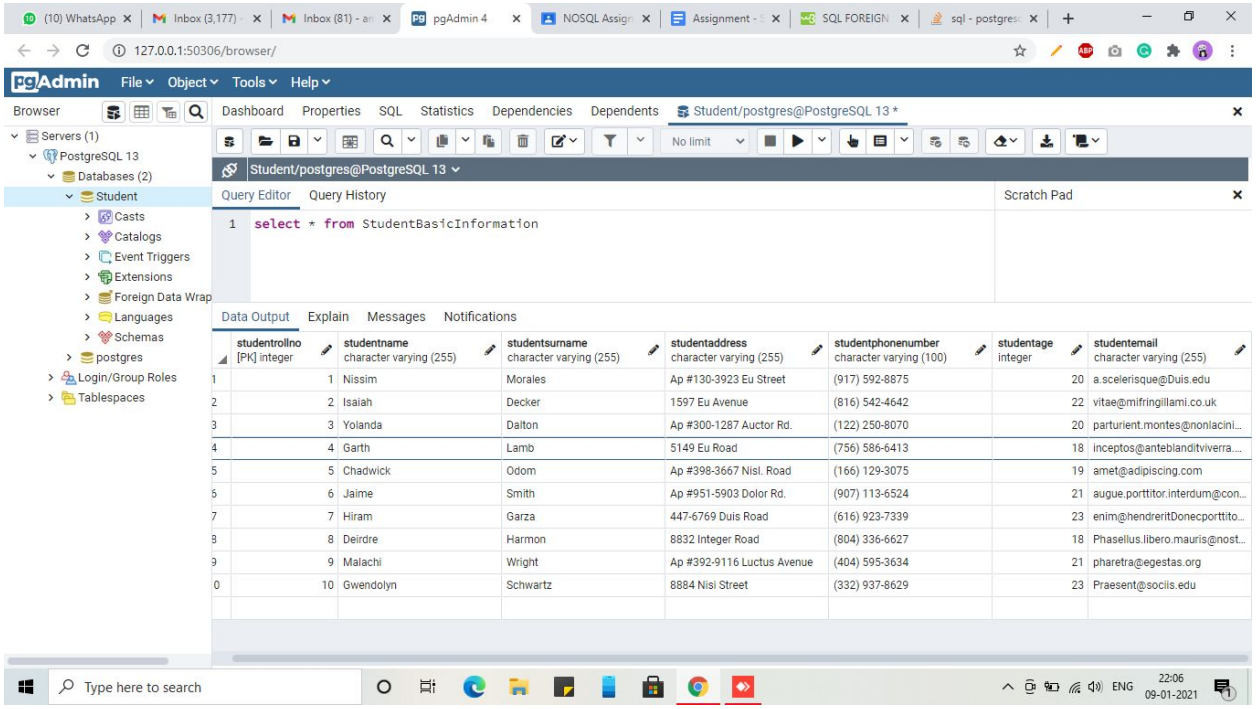
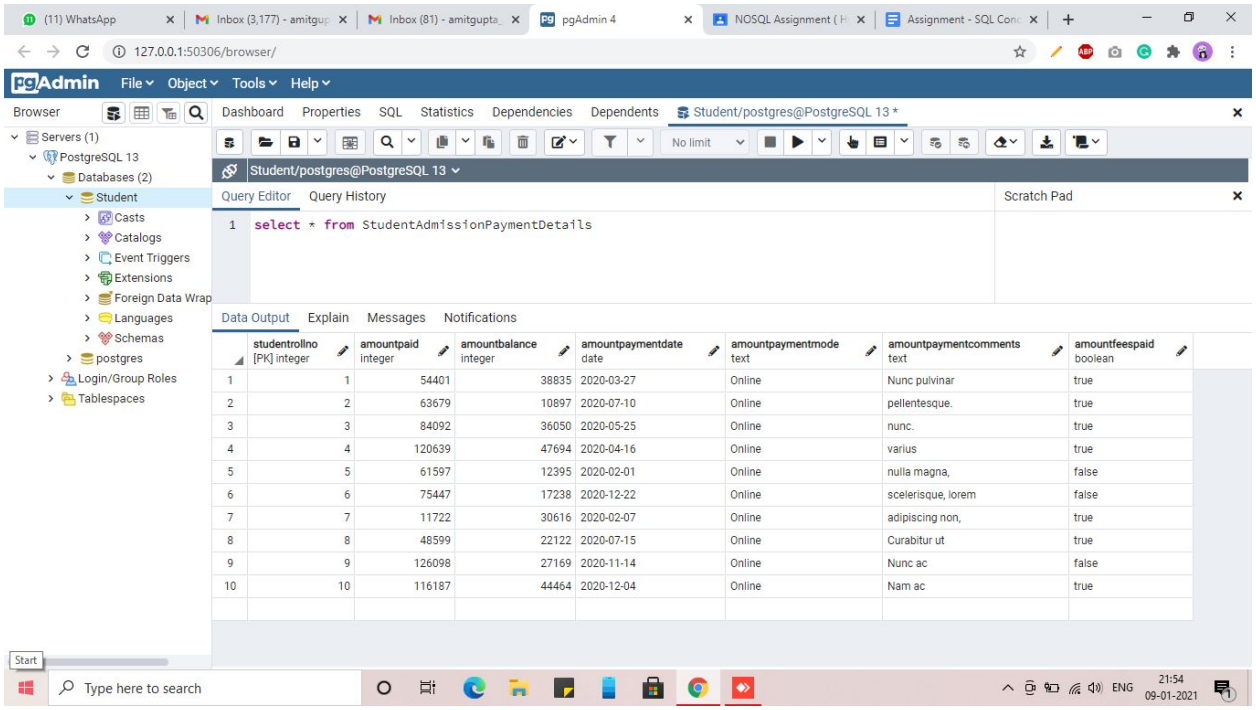
1. Create a Student Database
2. Create the following table under the Student Database:
    a. StudentBasicInformation
        i. Columns
            1. StudentName
            2. StudentSurname
            3. StudentRollNo
            4. StudentAddress
            5. Add more three basic columns of the name of your own

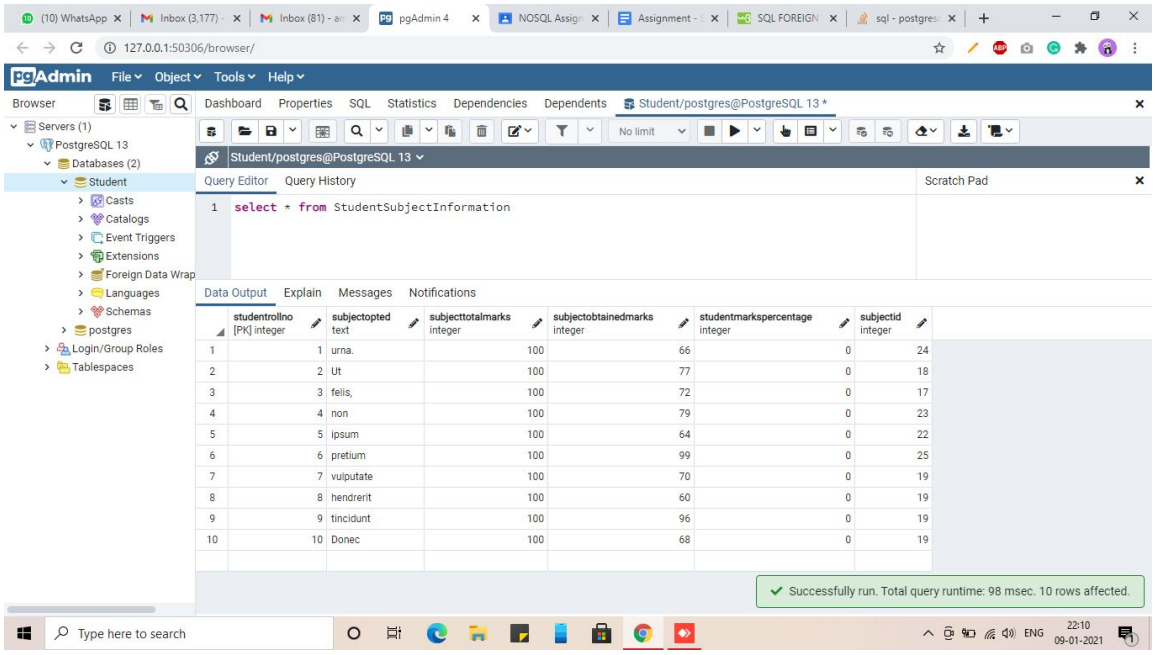


    b. StudentAdmissionPaymentDetails
        i. Columns
            1. StudentRollNo
            2. AmountPaid
            3. AmountBalance
            4. Add more four basic columns of the name of your own



	studentrollno [PK] integer	amountpaid integer	amountbalance integer	amountpaymentdate date	amountpaymentmode text	amountpaymentcomments text	amountfeespaid boolean
1	1	54401	38835	2020-03-27	Online	Nunc pulvinar	true
2	2	63679	10897	2020-07-10	Online	pellentesque.	true
3	3	84092	36050	2020-05-25	Online	nunc.	true
4	4	120639	47694	2020-04-16	Online	varius	true
5	5	61597	12395	2020-02-01	Online	nulla magna,	false
6	6	75447	17238	2020-12-22	Online	scelerisque, lorem	false
7	7	11722	30616	2020-02-07	Online	adipiscing non,	true
8	8	48599	22122	2020-07-15	Online	Curabitur ut	true
9	9	126098	27169	2020-11-14	Online	Nunc ac	false
10	10	116187	44464	2020-12-04	Online	Nam ac	true

c. StudentSubjectInformation
   i. Columns
      1. SubjectOpted
      2. StudentRollNo
      3. SubjectTotalMarks
      4. SubjectObtainedMarks
      5. StudentMarksPercentage
      6. Add more one columns of the name of your own

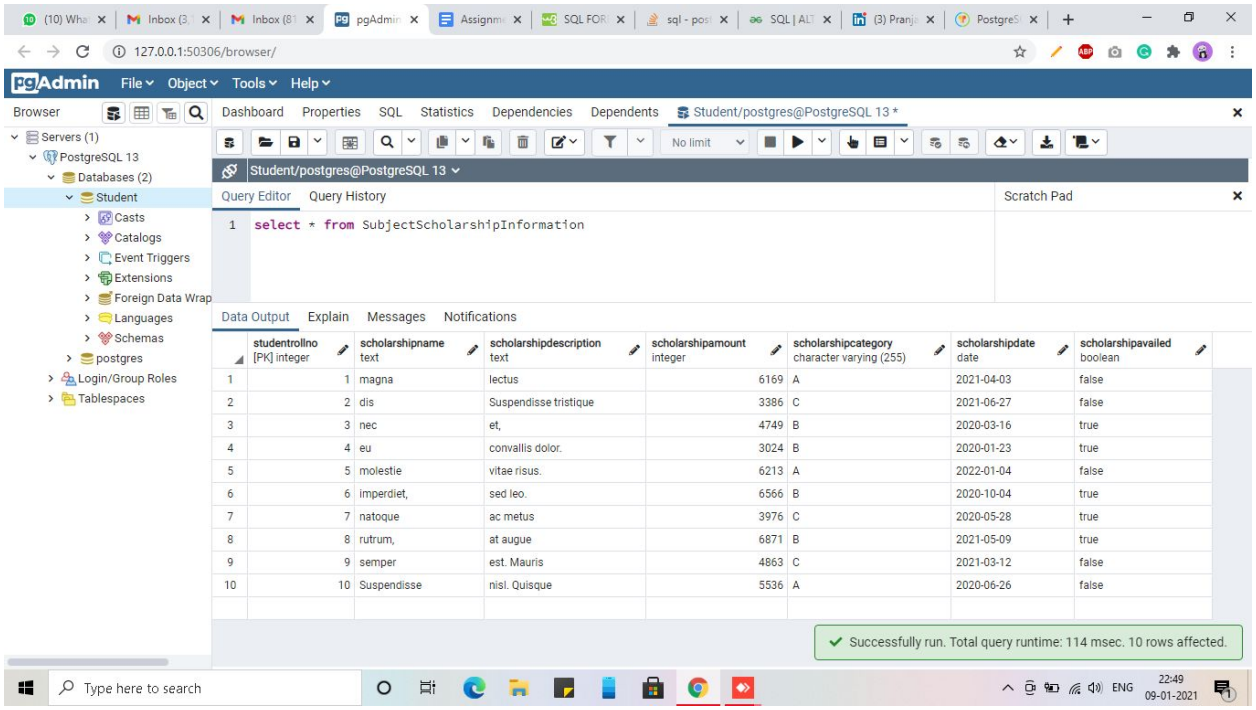


	studentrollno [PK] integer	subjectopted text	subjecttotalmarks integer	subjectobtainedmarks integer	studentmarkspercentage integer	subjectid integer
1	1	urna.	100	66	0	24
2	2	Ut	100	77	0	18
3	3	felis,	100	72	0	17
4	4	non	100	79	0	23
5	5	ipsum	100	64	0	22
6	6	pretium	100	99	0	25
7	7	vulputate	100	70	0	19
8	8	hendrerit	100	60	0	19
9	9	tincidunt	100	96	0	19
10	10	Donec	100	68	0	19

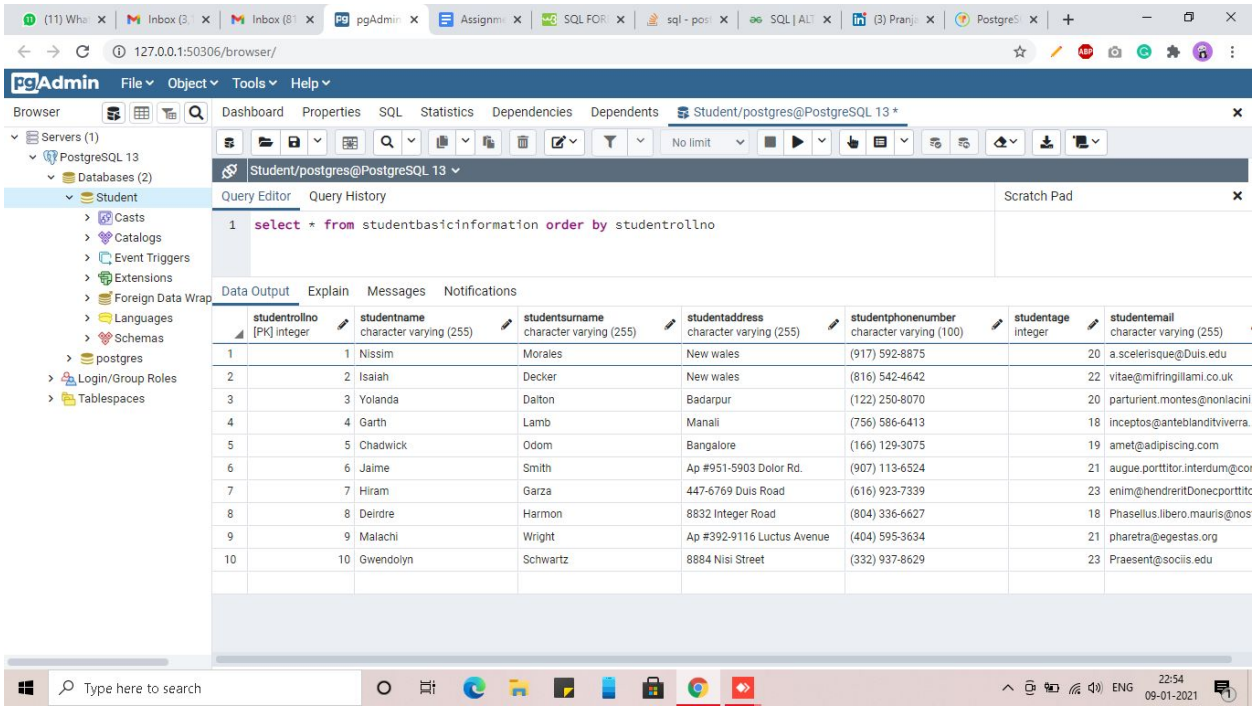
D. SubjectScholarshipInformation

i. Columns

1. StudentRollNo
2. ScholarshipName
3. ScholarshipDescription
4. ScholarshipAmount
5. ScholarshipCategory
6. Add more two columns of the name of your own



5) Update any 5 records of your choice in any table like update the StudentAddress with some other address content and likewise so on with any records of any table of your choice



7) Select the student details records who has received the scholarship more than 5000Rs/-

Data Output | Explain | Messages | Notifications

	studentrollno [PK] integer	scholarshipname text	scholarshipdescription text	scholarshipamount integer	scholarshipcategory character varying (255)	scholarshipdate date	scholarshipavailed boolean
1	1	magna	lectus	6169	A	2021-04-03	false
2	5	molestie	vitae risus.	6213	A	2022-01-04	false
3	6	imperdiet,	sed leo.	6566	B	2020-10-04	true
4	8	rutrum,	at augue	6871	B	2021-05-09	true
5	10	Suspendisse	nisl. Quisque	5536	A	2020-06-26	false

8) Select the students who opted for scholarship but has not got the scholarship

	studentrollno [PK] integer	scholarshipname text	scholarshipdescription text	scholarshipamount integer	scholarshipcategory character varying (255)	scholarshipdate date	scholarshipavailed boolean
1	1	magna	lectus	6169	A	2021-04-03	false
2	2	dis	Suspendisse tristique	3386	C	2021-06-27	false
3	5	molestie	vitae risus.	6213	A	2022-01-04	false
4	9	semper	est. Mauris	4863	C	2021-03-12	false
5	10	Suspendisse	nisl. Quisque	5536	A	2020-06-26	false

9) Fill in data for the percentage column i.e. StudentMarksPercentage in the table StudentSubjectInformation by creating and using the stored procedure created

Data Output | Explain | Messages | Notifications

	studentrollno [PK] integer	subjectopted text	subjecttotalmarks integer	subjectobtainedmarks integer	studentmarkspercentage integer	subjectid integer
1	1	urna.	100	66	66	24
2	2	Ut	100	77	77	18
3	3	felis,	100	72	72	17
4	4	non	100	79	79	23
5	5	ipsum	100	64	64	22
6	6	pretium	100	99	99	25
7	7	vulputate	100	70	70	19
8	8	hendrerit	100	60	60	19
9	9	tincidunt	100	96	96	19
10	10	Donec	100	68	68	19

10) Decide the category of the scholarship depending upon the marks/percentage obtained by the student and likewise update the ScholarshipCategory column, create a stored procedure in order to handle this operation

Data Output | Explain | Messages | Notifications

	studentrollno [PK] integer	scholarshipname text	scholarshipdescription text	scholarshipamount integer	scholarshipcategory character varying (255)	scholarshipdate date	scholarshippreviouslyavailed boolean
1	8	rutrum,	at augue	6871	Diamond	2021-05-09	true
2	10	Suspendisse	nisl. Quisque	5536	Diamond	2020-06-26	false
3	1	magna	lectus	6169	Diamond	2021-04-03	false
4	5	molestie	vitae risus.	6213	Diamond	2022-01-04	false
5	7	natoque	ac metus	3976	CROWN	2020-05-28	true
6	3	nec	et,	4749	CROWN	2020-03-16	true
7	4	eu	convallis dolor.	3024	CROWN	2020-01-23	true
8	2	dis	Suspendisse tristique	3386	CROWN	2021-06-27	false
9	9	semper	est. Mauris	4863	ACE	2021-03-12	false
10	6	imperdiet,	sed leo.	6566	ACE	2020-10-04	true

11) Create the View which shows the balance amount to be paid by the student along with the student detailed information (use join)

Data Output | Explain | Messages | Notifications

	amountbalance integer	studentrollno integer	studentname character varying (255)	studentsurname character varying (255)	studentaddress character varying (255)	studentphonenumber character varying (100)	studentage integer
1	38835	1	Nissim	Morales	New wales	(917) 592-8875	20
2	10897	2	Isaiah	Decker	New wales	(816) 542-4642	22
3	36050	3	Yolanda	Dalton	Badarpur	(122) 250-8070	20
4	47694	4	Garth	Lamb	Manali	(756) 586-6413	18
5	12395	5	Chadwick	Odom	Bangalore	(166) 129-3075	19
6	17238	6	Jaime	Smith	Ap #951-5903 Dolor Rd.	(907) 113-6524	21
7	30616	7	Hiram	Garza	447-6769 Duis Road	(616) 923-7339	23
8	22122	8	Deirdre	Harmon	8832 Integer Road	(804) 336-6627	18
9	27169	9	Malachi	Wright	Ap #392-9116 Luctus Avenue	(404) 595-3634	21
10	44464	10	Gwendolyn	Schwartz	8884 Nisi Street	(332) 937-8629	23

12) Get the details of the students who haven't got any scholarship (use joins/subqueries)

```
1  select si.StudentRollNo,si.StudentName,si.StudentSurname,si.StudentAddress,si.StudentPhoneNumb
2  from studentbasicinformation si inner join SubjectScholarshipInformation ss
3  on si.StudentRollNo=ss.StudentRollNo
4  where ss.ScholarshipAvailed= false;
```

Data Output | Explain | Messages | Notifications

	studentrollno [PK] integer	studentname character varying (255)	studentsurname character varying (255)	studentaddress character varying (255)	studentphonenumber character varying (100)
1	1	Nissim	Morales	New wales	(917) 592-8875
2	5	Chadwick	Odom	Bangalore	(166) 129-3075
3	10	Gwendolyn	Schwartz	8884 Nisi Street	(332) 937-8629
4	2	Isaiah	Decker	New wales	(816) 542-4642
5	9	Malachi	Wright	Ap #392-9116 Luctus Avenue	(404) 595-3634

13) Create a Stored Procedure which will return the amount balance to be paid by the student as per the student roll number passed through the stored procedure as the input

Query Editor | Query History

```
1  drop procedure balanceamount;
2  create procedure balanceamount (IN rollno i
3      language plpgsql
4      as $$
5      declare amount int;
6      begin
7          select amountbalance into amount
8          from StudentAdmissionPaymentDetails
9          raise notice 'balance amount to be
10     end; $$
11
12     call balanceamount(1);
```

Data Output | Explain | Messages | Notifications

```
NOTICE:  balance amount to be paid : 38835
CALL

Query returned successfully in 110 msec.
```

14) Retrieve the top five student details as per the StudentMarksPercentage values (use subqueries)

Query Editor | Query History | Scratch Pad

```
1  select t.StudentRollNo,si.StudentName,t.Stud
2  from studentbasicinformation si
3  inner join (
4      SELECT studentrollno,StudentMarksPercent
5             row_number() over (order by Stude
6      FROM StudentSubjectInformation
7  )t
8  on si.StudentRollNo=t.StudentRollNo
9  WHERE t.rn <= 5
```

Data Output | Explain | Messages | Notifications

	studentrollno integer	studentname character varying (255)	studentmarkspercentage integer
1	6	Jaime	99
2	9	Malachi	96
3	2	Isaiah	77
4	3	Yolanda	72
5	4	Garth	79

Try to use all the three types of join learned today in a relevant way, and explain the same way you thought of using that particular join for your selected scenarios (try to cover relevant and real-time scenarios for all the three studied joins)

16) Mention the differences between the delete, drop and truncate commands

1. DELETE :

Basically, it is a Data Manipulation Language Command (DML). It is use to delete the one or more tuples of a table. With the help of “DELETE” command we can either delete all the rows in one go or can delete row one by one. i.e., we can use it as per the requirement or the condition using Where clause. It is comparatively slower than TRUNCATE cmd.

- **SYNTAX –**

  If we want to delete all the rows of the table:

  ```
  DELETE from ;
  ```

2. DROP :

It is a Data Definition Language Command (DDL). It is use to drop the whole table. With the help of “DROP” command we can drop (delete) the whole structure in one go i.e. it removes the named elements of the schema. By using this command the existence of the whole table is finished or say lost.

- **SYNTAX –**

  If we want to drop the table:

  ```
  DROP table ;
  ```

3. TRUNCATE :

It is also a Data Definition Language Command (DDL). It is use to delete all the rows of a relation (table) in one go. With the help of “TRUNCATE” command we can’t delete the single row as here WHERE clause is not used. By using this command the existence of all the rows of the table is lost. It is comparatively faster than delete command as it deletes all the rows fastly.

- **SYNTAX –**

  If we want to use truncate :

  ```
  TRUNCATE  ;
  ```

17) Get the count of the Scholarship category which is highly been availed by the students, i.e. get the count of the total number of students corresponding to each scholarships category

Data Output | Explain | Messages | Notifications

	scholarshipcategory character varying (255)	count bigint
1	CROWN	4
2	Diamond	4
3	ACE	2

18) Along with assignment no. 17 try to retrieve the maximum used scholarship category

Data Output | Explain | Messages | Notifications

	maximumusedscholarshipcategory character varying (255)	count bigint
1	CROWN	4

19) Retrieve the percentage of the students along with students detailed information who has scored the highest percentage along with availing the maximum scholarship amount

Student/postgres@PostgreSQL 13

Query Editor | Query History

```
2  from studentbasicinformation si
3  inner join studentsubjectinformation ss on
4  si.studentrollno=ss.studentrollno
5  where studentmarkspercentage=(select studentmarkspercentage from subjectscholarshipinformatior
6      inner join studentsubjectinformation ss
7      on ssl.studentrollno=ss.studentrollno
8      order by studentmarkspercentage desc,scholarshipamount desc
9      limit 1)
```

Data Output | Explain | Messages | Notifications

	studentrollno integer	studentname character varying (255)	studentsurname character varying (255)	studentmarkspercentage integer
1	6	Jaime	Smith	99

20. VIEWS: The view is a virtual table. It does not physically exist. Rather, it is created by a query joining one or more tables.• A view contains rows and columns, just like a real table• The fields in a view are fields from one or more real tables in the databaseCreating an SQL VIEW.

Syntax: CREATE VIEW view_name ASSELECT column_name(s)FROM table_nameWHERE condition;

TRIGGER: What is a Trigger?A trigger is a block structure which is fired when a DML statements like Insert,Delete, Update is executed on a database table. A trigger is triggered automatically when an associated DML statement is executed.•

Syntax : CREATE [OR REPLACE] TRIGGER trigger_name{BEFORE | AFTER | INSTEAD OF }{INSERT [OR] | UPDATE [OR] | DELETE }[OF col_name] ON table_name [REFERENCING OLD AS o NEW AS n] [FOR EACH ROW]WHEN (condition) ;

Functions: A function is a group of statements that executes upon request• Python provides many built-in functions and allows programmers to define their own functions• A request to execute a function is known as a function call• When a function is called, it may be passed arguments that specify data upon which the function performs its computation• Functions defined within class statements are also called methodsThe def Statement• The def statement is the most common way to define a functionSyntaxdef function-name(parameters):statement(s).

Stored procedure: In a database management system (DBMS), a stored procedure is a set of Structured QueryLanguage (SQL) statements with an assigned name that's stored in the database in compiled form so that it can be shared by a number of programs.•