

PHP - PostgreSQL Lab

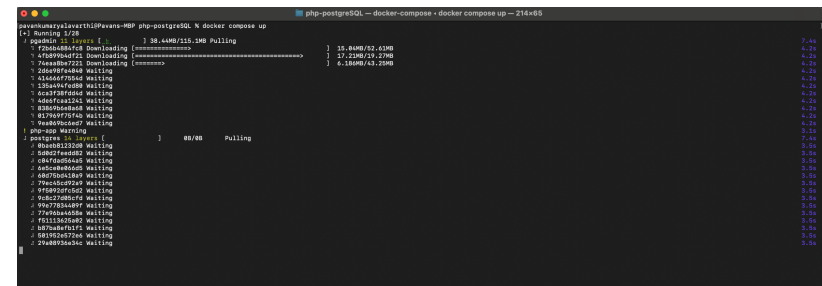
Problem Statement:

In this exercise, you've to create a web application to manage the marks of CS699 students (of course, you'll be dealing with dummy marks!).

- a. In PostgreSQL, a database is created named db.
- b. In db, we have a table named "users", to store information about each student: roll number, name, email address, and a login password (for this website, not your LDAP login or any real password). Roll number is the primary key for this table. Each field is a varchar of max length 256 characters.
- c. Another table called 'marks' in the database, to store information about each students' marks in: lab, midsem, project, and endsem. roll number as foreign key into the 'users' table. Each marks field is a 'real' data type.
- e. Now create a web form for a user to enter his/her roll number and password. On submitting this form, a php script must be run to show the students' marks as an unordered list. Don't forget to do error handling such as: Login Failed, No marks found for Roll Number: XYZ
- f. When username is admin and password is admin redirect him to processStudents.php which shows form to add students, which when submitted need to add given details to users table. Show error message if any during insertion on the top of the page.
- g. And admin when clicking on add marks hyperlink, navigates to processMarks.php which shows form to add marks, which when submitted need to add given details to marks table. Show error message if any during insertion on the top of the page.
- H. Forms are already present. You need to write the PHP code to handle the post requests.

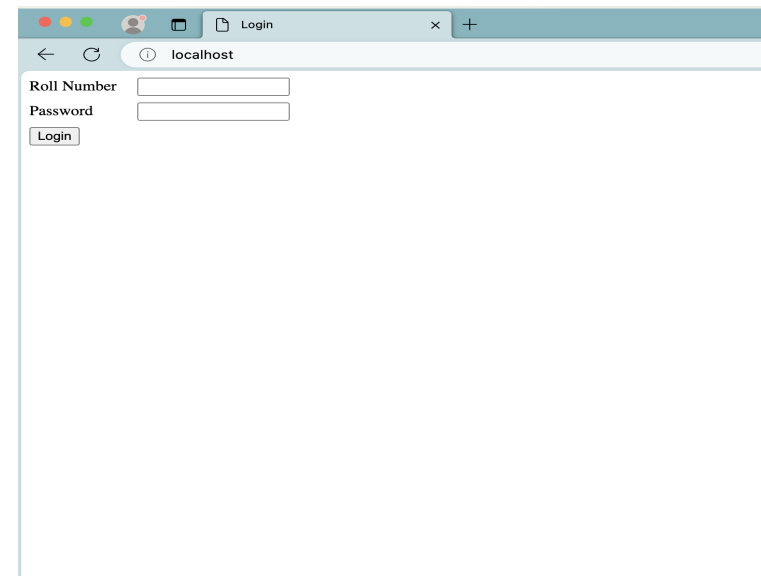
Installation Instructions:

1. Download the zip file [PHP - PostgreSQL](#) and unzip it
2. Open the terminal inside the folder, and run the command
docker compose up



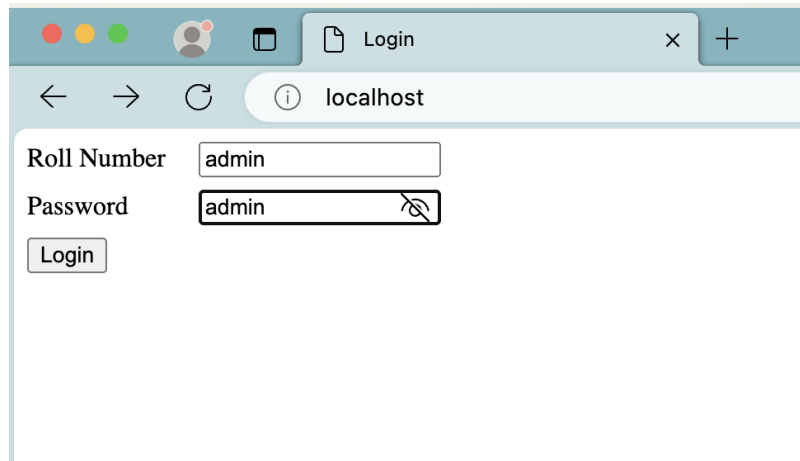
This downloads the required image for the lab and opens port 80 for PHP and 5050 for pgadmin

3. For database table schema, refer to `./script/createTable.sql`
4. Type `localhost` in your browser you will see the login screen as shown below if the build is successful.



5. Handling case of admin:

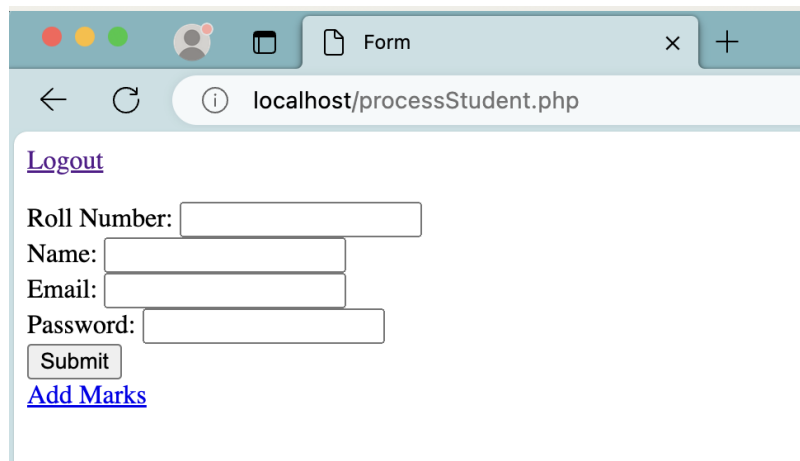
- a. When roll number and password are admin, page need to navigate to processStudent.php [Login logic needed to be handled in process.php]



Roll Number:

Password:

Below is the screenshot of processStudent.php



[Logout](#)

Roll Number:

Name:

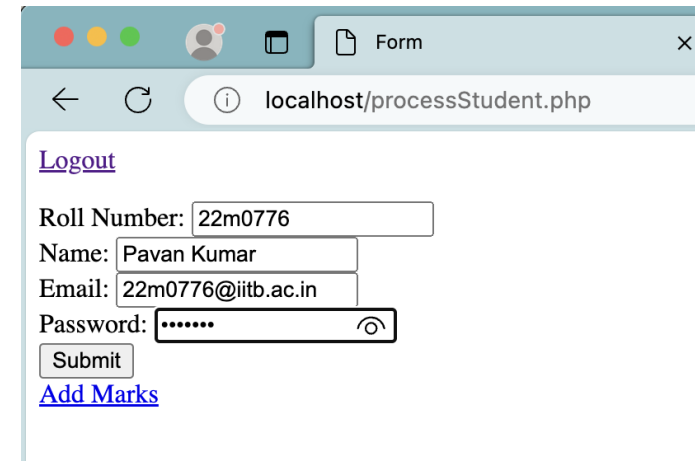
Email:

Password:

[Add Marks](#)

- b. processStudent.php needs to handle logic to insert the details entered in the form into db users.

- i. On Successful insertion, it need to show "Insertion successful"



[Logout](#)

Roll Number:

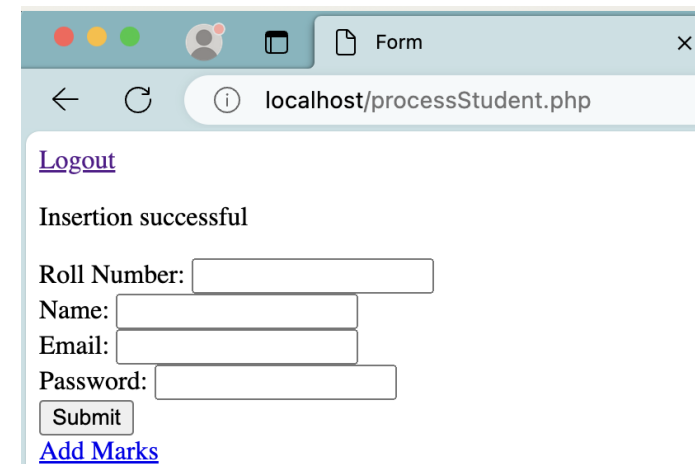
Name:

Email:

Password:

[Add Marks](#)

Success message...



[Logout](#)

Insertion successful

Roll Number:

Name:

Email:

Password:

[Add Marks](#)

- ii. On failure, the status needs to show failure with the reason.

Logout

Insertion successful

Roll Number: 22m0776

Name: Pavan Kumar

Email: 22m0776@iitb.ac.in

Password:

Submit

[Add Marks](#)

Failure message

Logout

Error: Insertion failed. Reason: ERROR: duplicate key value violates unique constraint "users_pkey" DETAIL: Key (roll_number)=(22m0776) already exists.

Roll Number:

Name:

Email:

Password:

Submit

[Add Marks](#)

- c. On clicking Add marks, navigate to processMarks.php, where we can enter the marks of respective students. processMarks.php needs to handle logic to insert the details entered in the form into db marks.
- Show "Insertion Successful" message when insertion went well else error message in similar pattern of error handling done in processStudent.php

Logout

Roll Number: 22m0776

Lab: 10

Midsem: 20

Project: 20

Endsem: 30

Submit

[Add Student](#)

Success message

Logout

Insertion successful

Roll Number:

Lab:

Midsem:

Project:

Endsem:

Submit

[Add Student](#)

Failure Message

Logout

Error: Insertion failed. Reason: ERROR: duplicate key value violates unique constraint "marks_pkey" DETAIL: Key (roll_number)=(22m0776) already exists.

Roll Number:

Lab:

Midsem:

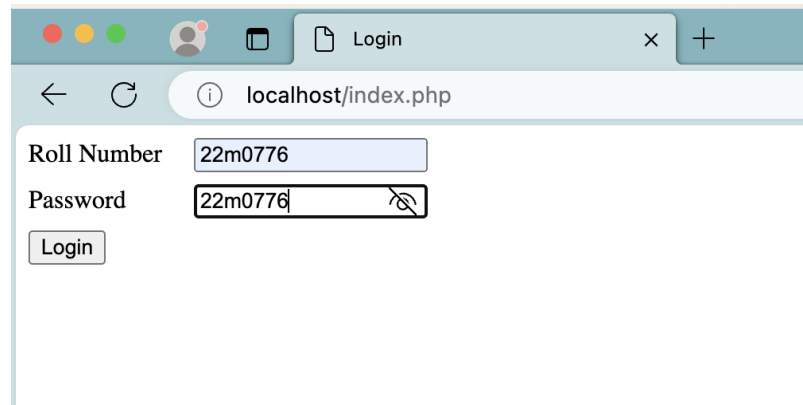
Project:

Endsem:

Submit

[Add Student](#)

- Handling cases of individual users.
 - If it is not admin, process.php needs to check for roll number and password in users table. If successful, website need to show their respective marks in an unordered list as shown in the below screenshot.

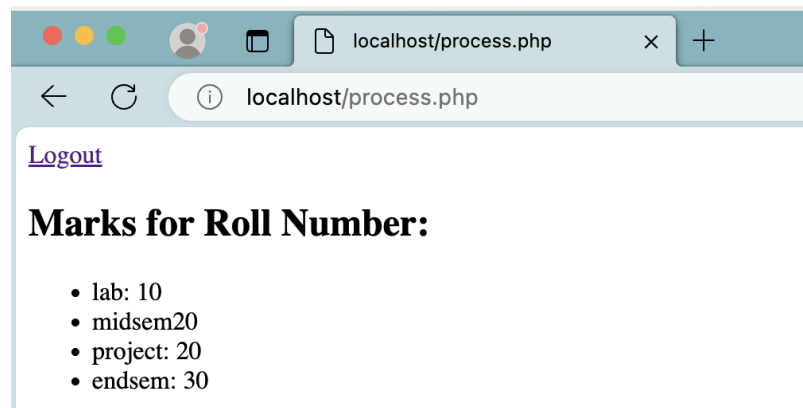


Roll Number

Password

Login

Successful login

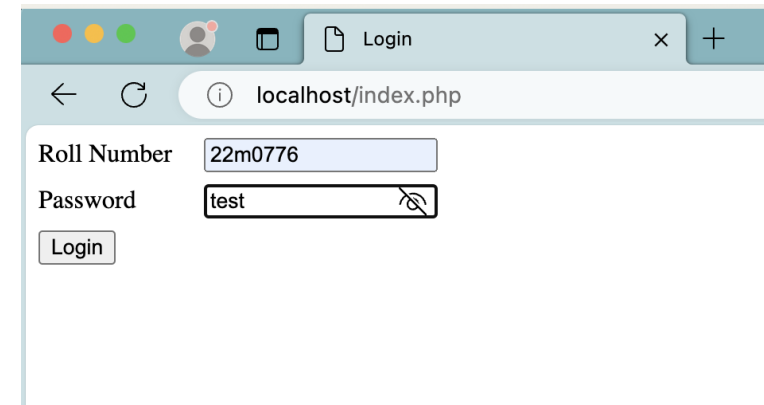


[Logout](#)

Marks for Roll Number:

- lab: 10
- midsem20
- project: 20
- endsem: 30

b. If no match found in users db, mention login failed.

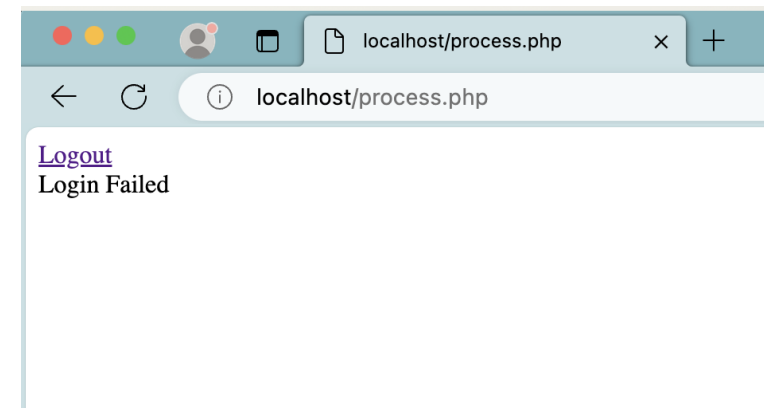


Roll Number

Password

Login

Failure message



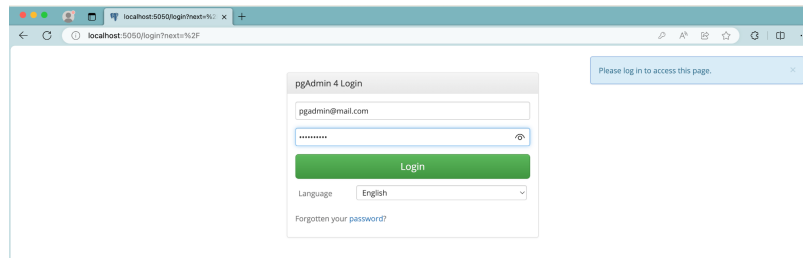
[Logout](#)

Login Failed

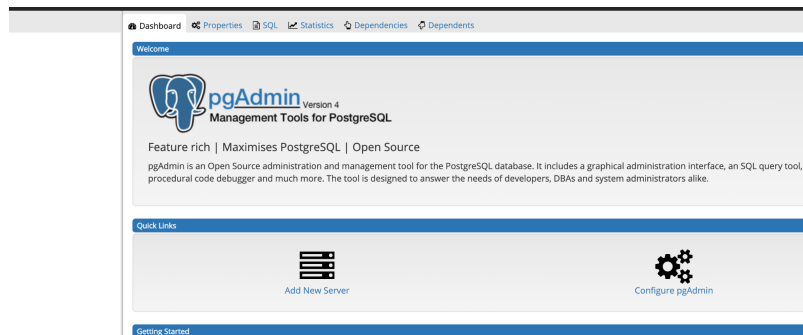
Optional:

If you need to access Database to see or query data, below are the steps.

1. Go to localhost:5050 and login with credentials pgadmin@mail.com and mypassword. [These are mentioned in env file]



2. Click on Add new server



3. In General tab, name it as test and in Connection tab, give
- Host name as postgres
 - Username as dev
 - Password as secret
 - And click on save

Create - Server

General

Connection

SSL

Advanced

Name

Test

Server group

Servers

Background

☐

Foreground

☐

Connect now?

☒

Comments

Either Host name or Host address must be specified.

i

?

Save

Cancel

Reset

Create - Server

General Connection SSL Advanced

Host name/address: postgres

Port: 5432

Maintenance database: postgres

Username: dev

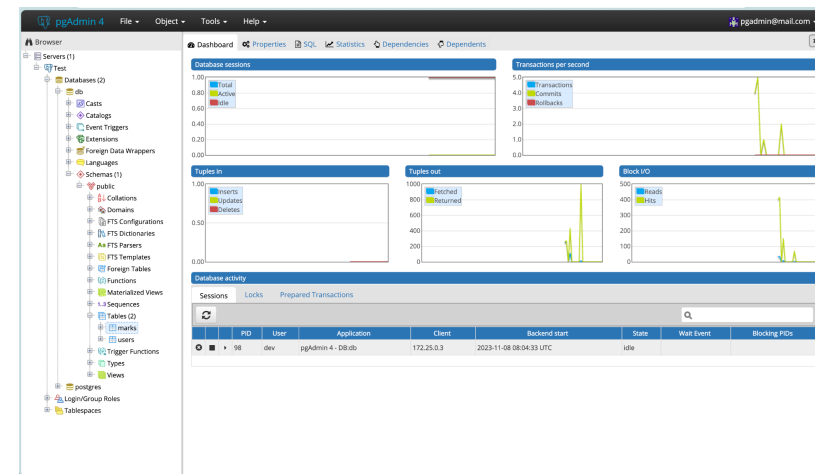
Password:

Save password? ☒

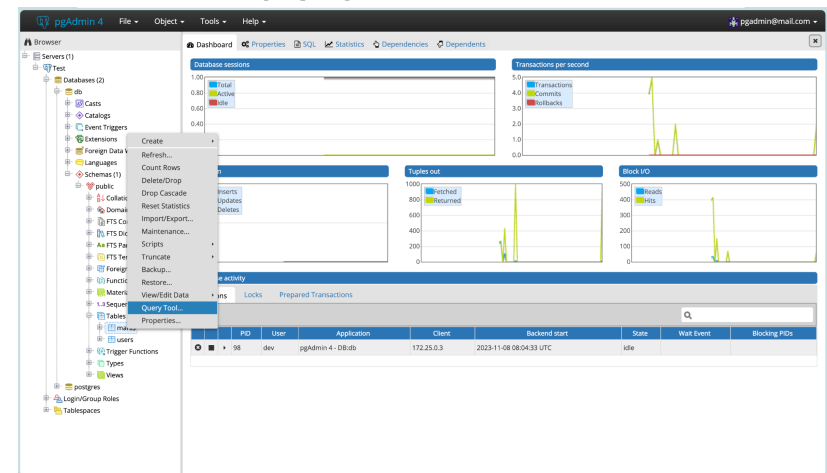
Role:

Save Cancel Reset

4. Expand **test > databases > db > schemas > public > Tables**. You can see the tables we created, it schema etc.



5. If need to query on table, right click on the table and select **Query Tool**, write and execute the query required.



pgAdmin 4

File • Object • Tools • Help •

pgadmin@mail.com

Servers (1)

Test

Databases (2)

db

Casts

Catalog

Event Triggers

Extensions

Foreign Data Wrappers

Languages

Schemas (1)

public

Collations

Domains

FTS Configurations

FTS Dictionaries

As FTS Parsers

FTS Templates

Foreign Tables

Functions

Materialized Views

Sequences

Tables (2)

marks

Users

Trigger Functions

Types

Views

progres

Login/Group Roles

Tablespaces

Dashboard Properties SQL Statistics Dependencies Dependencies Query: db on dev@Test

db on dev@Test

```
1 select * from users;
```

Data Output Explain Messages Query History

	roll_number	name	email_address	password
	character varying (256)	character varying (256)	character varying (256)	character varying (256)
1	22m0776	Pawan Kumar	22m0776@biba.ac.in	22m0776

✓ Successfully run. Total query runtime: 114 msec. 1 rows affected.