Assignment -2

Assignment Date	27 September 2022
Student Name	C.Selva Yogiraam
Student Roll Number	9517201904143
Maximum Marks	2 Marks

Question-1:

1.Create User table with user with email,username,roll number, password.

2. Perform UPDATE,DELETE Queries with user table

3. Connect python code to db2.

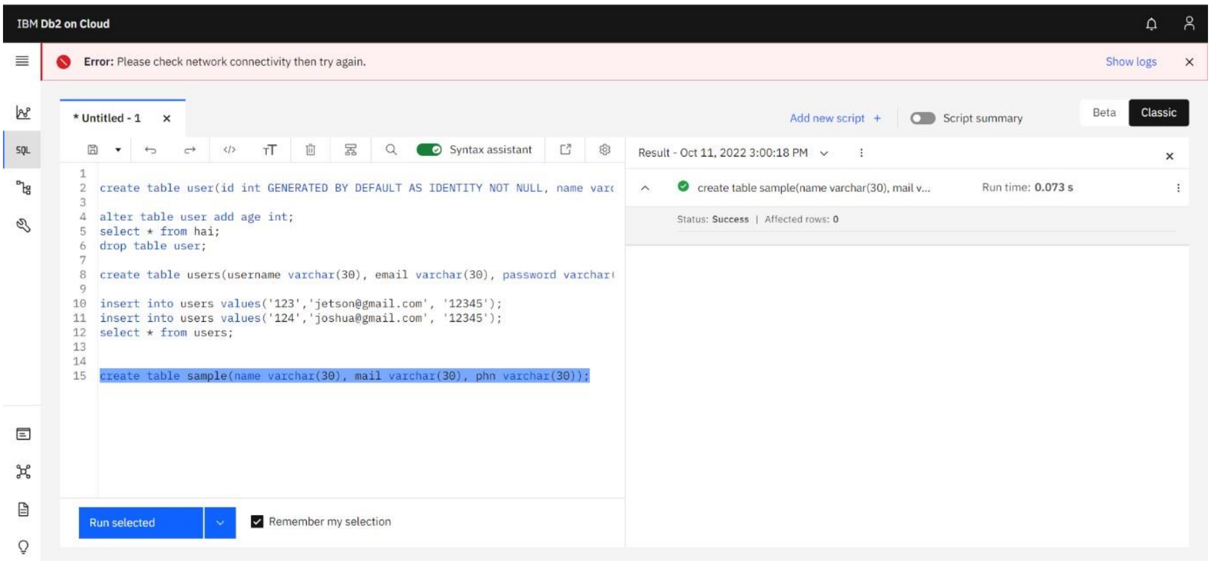
4. Create a flask app with registration page, login page and welcome page.

By default, load the registration page once the user enters all the fields store the data in database and navigate to login page authenticate user username and password. If the user is valid show the welcome page.

Solution:

Creating user table:

CREATE TABLE SAMPLE(name varchar(30), email varchar(30), password varchar(30))



Error: Please check network connectivity then try again.

```

create table user(id int GENERATED BY DEFAULT AS IDENTITY NOT NULL, name varc

alter table user add age int;
select * from hai;
drop table user;

create table users(username varchar(30), email varchar(30), password varchar(

insert into users values('123','jetson@gmail.com', '12345');
insert into users values('124','joshua@gmail.com', '12345');
select * from users;


create table sample(name varchar(30), mail varchar(30), phn varchar(30));
```

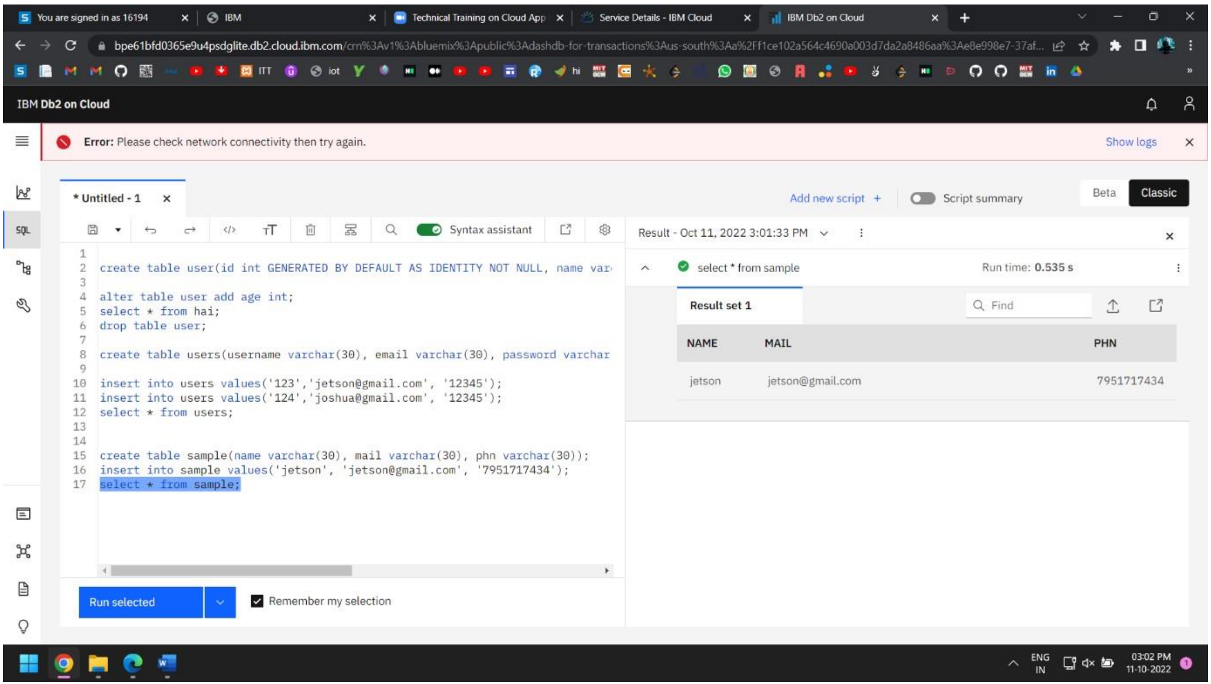
Result - Oct 11, 2022 3:00:18 PM

create table sample(name varchar(30), mail v... Run time: 0.073 s

Status: Success | Affected rows: 0

Performing update, delete:

update sample set mail= 'Selva@gmail.com' where name = 'Selva';



IBM Db2 on Cloud

Error: Please check network connectivity then try again.

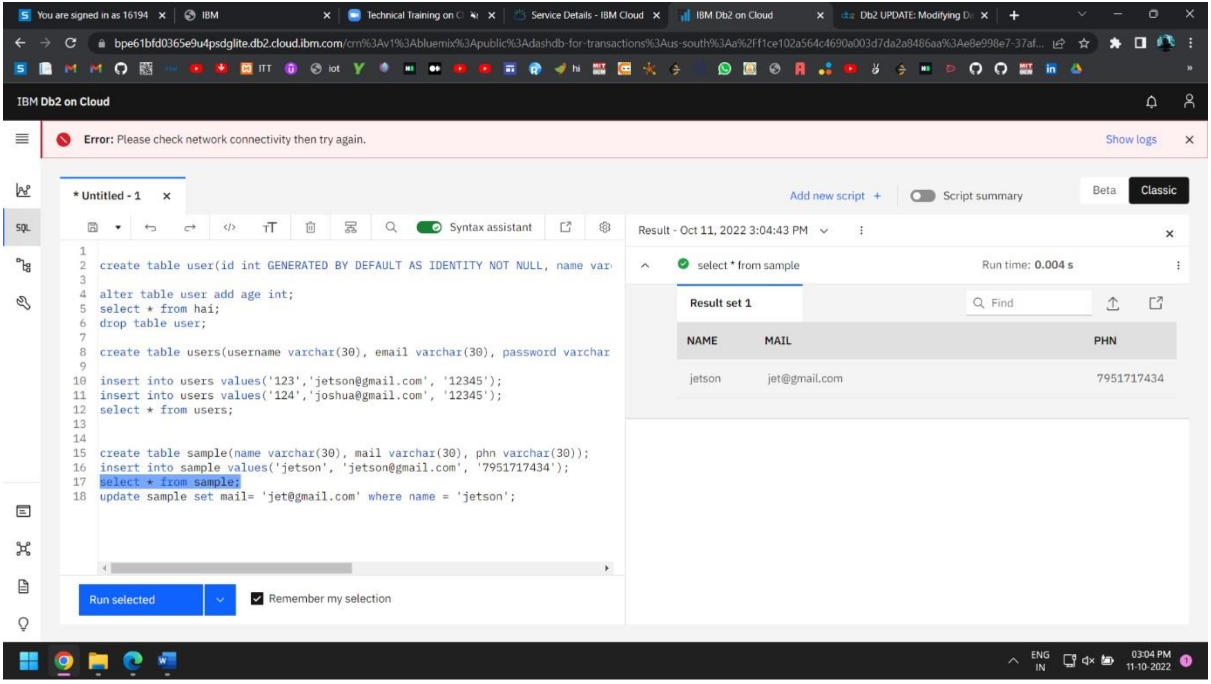
```
create table user(id int GENERATED BY DEFAULT AS IDENTITY NOT NULL, name var

alter table user add age int;
select * from hai;
drop table user;

create table users(username varchar(30), email varchar(30), password varchar

insert into users values('123','jetson@gmail.com', '12345');
insert into users values('124','joshua@gmail.com', '12345');
select * from users;


create table sample(name varchar(30), mail varchar(30), phn varchar(30));
insert into sample values('jetson', 'jetson@gmail.com', '7951717434');
select * from sample;
```

Result - Oct 11, 2022 3:01:33 PM

select * from sample   Run time: 0.535 s

Result set 1

NAME	MAIL	PHN
jetson	jetson@gmail.com	7951717434



IBM Db2 on Cloud

Error: Please check network connectivity then try again.

```
create table user(id int GENERATED BY DEFAULT AS IDENTITY NOT NULL, name var

alter table user add age int;
select * from hai;
drop table user;

create table users(username varchar(30), email varchar(30), password varchar

insert into users values('123','jetson@gmail.com', '12345');
insert into users values('124','joshua@gmail.com', '12345');
select * from users;


create table sample(name varchar(30), mail varchar(30), phn varchar(30));
insert into sample values('jetson', 'jetson@gmail.com', '7951717434');
select * from sample;
update sample set mail= 'jet@gmail.com' where name = 'jetson';
```

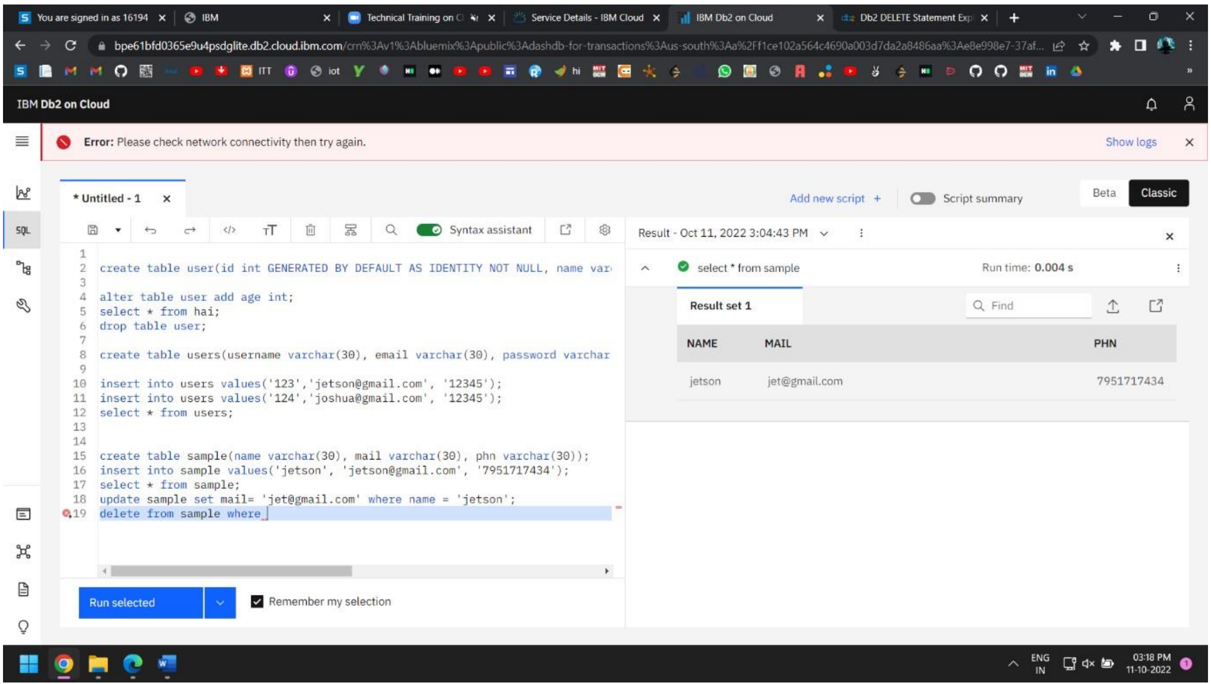
Result - Oct 11, 2022 3:04:43 PM

select * from sample   Run time: 0.004 s

Result set 1

NAME	MAIL	PHN
jetson	jet@gmail.com	7951717434

delete from sample where name = 'selva';



Error: Please check network connectivity then try again.

```
create table user(id int GENERATED BY DEFAULT AS IDENTITY NOT NULL, name var

alter table user add age int;
select * from hai;
drop table user;

create table users(username varchar(30), email varchar(30), password varchar

insert into users values('123','jetson@gmail.com', '12345');
insert into users values('124','joshua@gmail.com', '12345');
select * from users;


create table sample(name varchar(30), mail varchar(30), phn varchar(30));
insert into sample values('jetson', 'jetson@gmail.com', '7951717434');
select * from sample;
update sample set mail= 'jet@gmail.com' where name = 'jetson';
delete from sample where
```

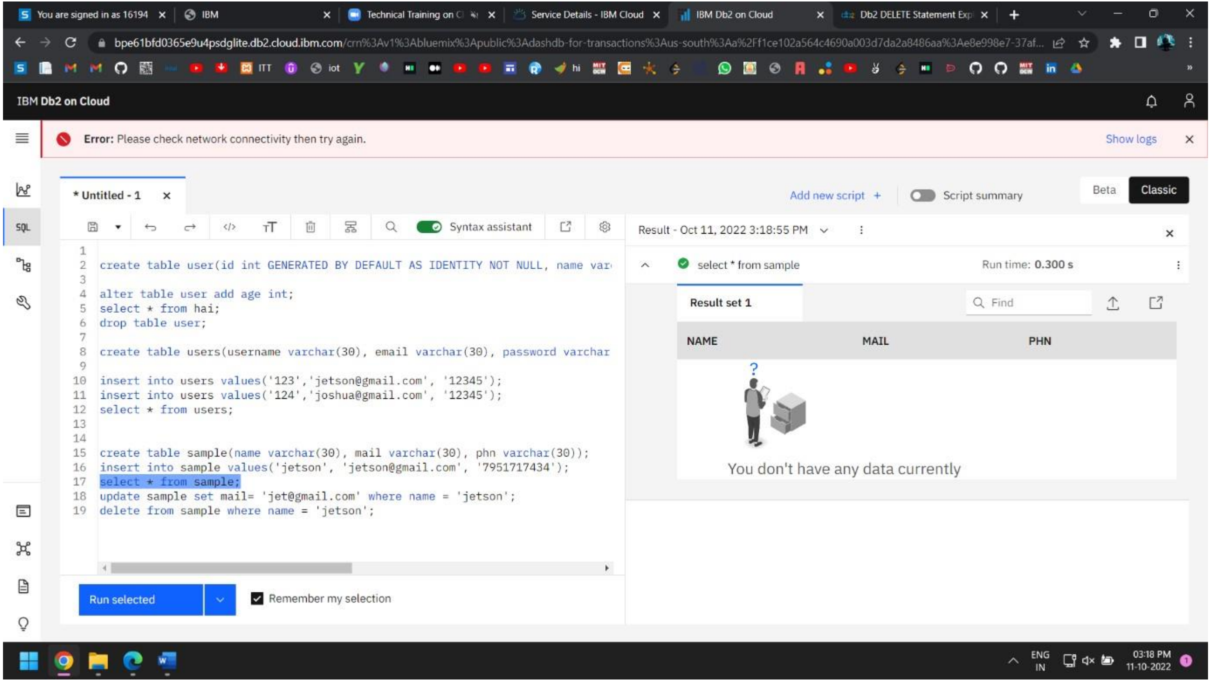
☑ Remember my selection

Result - Oct 11, 2022 3:04:43 PM

select * from sample — Run time: 0.004 s

Result set 1

NAME	MAIL	PHN
jetson	jet@gmail.com	7951717434



Error: Please check network connectivity then try again.

```
create table user(id int GENERATED BY DEFAULT AS IDENTITY NOT NULL, name var

alter table user add age int;
select * from hai;
drop table user;

create table users(username varchar(30), email varchar(30), password varchar

insert into users values('123','jetson@gmail.com', '12345');
insert into users values('124','joshua@gmail.com', '12345');
select * from users;


create table sample(name varchar(30), mail varchar(30), phn varchar(30));
insert into sample values('jetson', 'jetson@gmail.com', '7951717434');
select * from sample;
update sample set mail= 'jet@gmail.com' where name = 'jetson';
delete from sample where name = 'jetson';
```

☑ Remember my selection

Result - Oct 11, 2022 3:18:55 PM

select * from sample — Run time: 0.300 s

Result set 1

NAME	MAIL	PHN

You don't have any data currently

Base.html:

```
<html>
<head>
  <link rel="stylesheet" href="static/css/main.css">
  {% block head %}
  {% endblock %}
</head>
<body>
  {% block body %}
  {% endblock %}
</body>
</html>
```

Main.css:

```
.login {    width: 60%;    border-radius: 1rem;    background: gray;
padding: 10px;    margin: auto;    text-align: center;    font-family: 'Gill
Sans', 'Gill Sans MT', Calibri, 'Trebuchet MS', sans-serif;
}
.back{    background-color:
black;
}
.sub{    background-color:
black;    color: white;
border-radius: 1rem;
}
.box{
  border-radius: 1rem;
background-color: black;    color:
white;
}
```

Dashboard.html:

```
{% extends 'base.html' %}
{% block head %}
<title>Login</title>
{% endblock %}
{% block body %}
<br><br><br><br><br>    <br><br><br><br><br>
<div class="login">
    <h1>
        Welcome to dashboard.. {{ msg }}
    </h1>
    <hr>
</div>
{% endblock %}
```

Login.html:

```
{% extends 'base.html' %}
{% block head %}
<title>Login</title>
{% endblock %}
{% block body %}
<br><br><br><br><br><br><br>
<div class="login">
    <h1>
        Login
    </h1>
    <hr>
    <form action="/login" method="POST">
        <br>
        username: <input type="text" class="box" name="username" id="username"><br><br>
        Password: <input type="password" class="box" name="passwd" id="passwd"><br><br>
        <input type="submit" class="sub" value="Sign in">
    </form>
    Don't have account? <a href="/signup">Sign up</a>
```

```
</div>
{% endblock %}
```

Signup.html:

```
{% extends 'base.html' %}
{% block head %}
<title>Login</title>
{% endblock %}
{% block body %}
<br><br><br><br><br> <br><br>
<div class="login">
  <h1>
    Sign Up
  </h1>
  <hr>
  <form action="/signup" method="POST">
    <br>
    username: <input type="text" class="box" name="username" id="username"><br><br>
email: <input type="text" class="box" name="email" id="eamil"><br><br>
    Password: <input type="password" class="box" name="passwd" id="passwd"><br><br>
    <input type="submit" class="sub" value="Sign up"><br>
  </form>
  Already have an account? <a href="/login">Login</a>
</div>
{% endblock %}
```

App.py:

```
from flask import Flask, render_template, request, session, redirect, url_for
import ibm_db import
re import ibm_db_dbi
app = Flask(__name__)
app.secret_key = 'a'
```

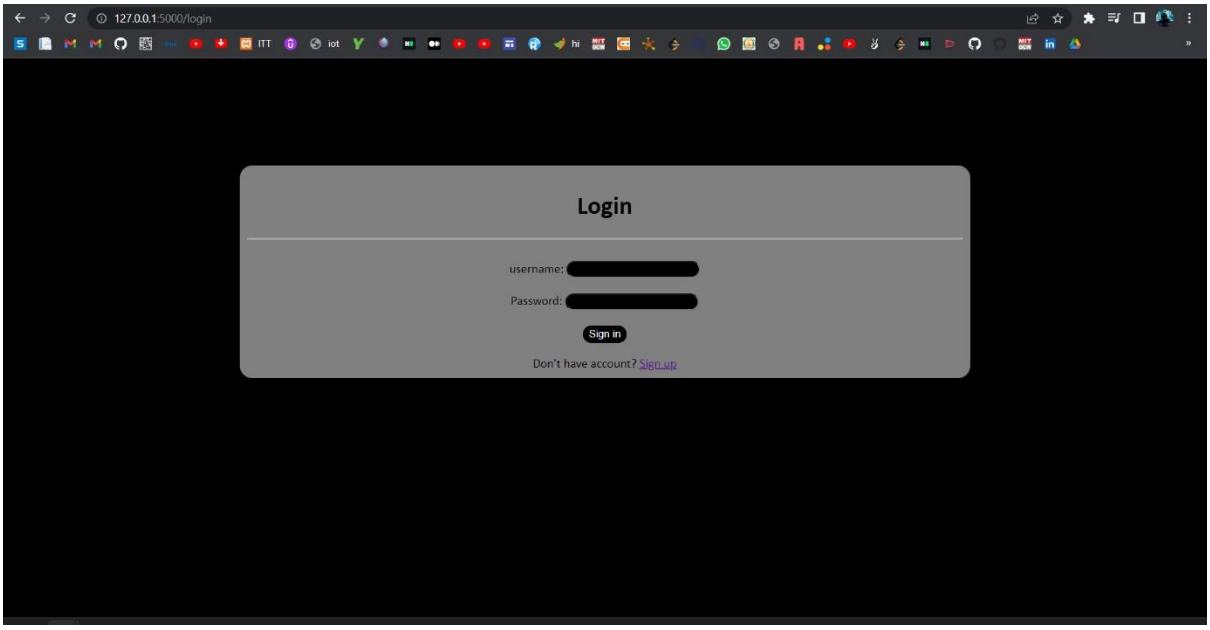
```
conn = ibm_db.connect("DATABASE=bludb;HOSTNAME=b70af05b-76e4-4bca-
a1f523dbb4c6a74e.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=32716;SECURIT
Y=S
SL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=xyn12614;PWD=oXjOuam0AOYeLVKq
", '', '')


@app.route('/login', methods=['POST', "GET"])

def login():

global userid

msg = ''

  if request.method == 'POST':

    username = request.form['username']        password =

request.form['passwd']        sql = "select * from users where

username = ? and password = ?"        stmt = ibm_db.prepare(conn,

sql)        ibm_db.bind_param(stmt, 1, username)

ibm_db.bind_param(stmt, 2, password)        ibm_db.execute(stmt)

account = ibm_db.fetch_assoc(stmt)

    print(account)

if account:

      session['Loggedin'] = True

session['id'] = account['USERNAME']           userid

= account['USERNAME']           session['username']

= account['USERNAME']           msg = 'logged in

successfully'


      return render_template('dashboard.html', msg=msg)

    else:

      msg = 'Incorrect user credentials'           return

render_template('dashboard.html', msg=msg)

  else:

    return render_template('login.html')
```

```
@app.route('/signup', methods=["POST", "GET"])
def signup():     if request.method == "POST":
username = request.form['username']        email =
request.form['email']        password =
request.form['passwd']        sql = "select * from
users where username = ?"        stmt =
ibm_db.prepare(conn, sql)
ibm_db.bind_param(stmt, 1, username)
ibm_db.execute(stmt)        account =
ibm_db.fetch_assoc(stmt)
    print(account)
if account:
      msg = 'Acccount already exists'        elif not
re.match(r'[^@]+@[^@]+\.[^@]+', email):
      msg = 'invalid email'        elif not
re.match(r'[A-Za-z0-9]+', username):
      msg = 'name must contain charectors and numbers'
    else:
      insert_sql = "insert into users values(?,?,?)"
prep_stmt = ibm_db.prepare(conn, insert_sql)
ibm_db.bind_param(prep_stmt, 1, username)
ibm_db.bind_param(prep_stmt, 2, email)
ibm_db.bind_param(prep_stmt, 3, password)
ibm_db.execute(prep_stmt)
      print("successs")           msg
= "succesfully signed up"
    return render_template('dashboard.html', msg=msg)
  else:
    return render_template('signup.html')
```

if __name__ == '__main__':

app.run(debug=False) **Output:**

Login:



127.0.0.1:5000/login

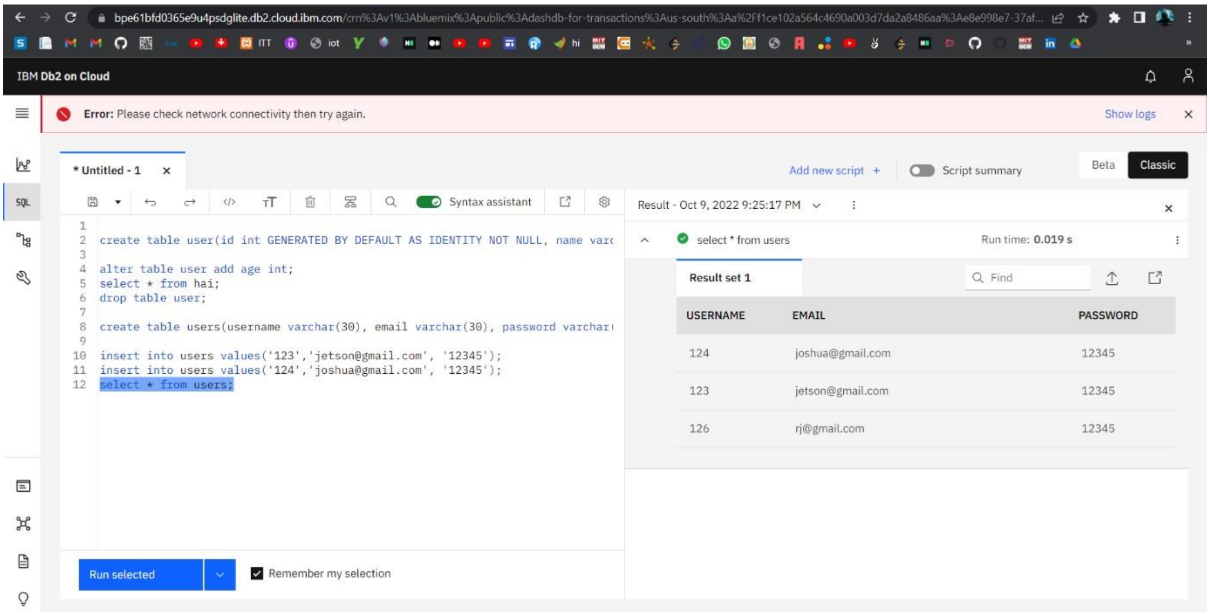
Login

username:

Password:

Sign in

Don't have account? Sign up



bpe61bfd0365e9u4psdglite.db2.cloud.ibm.com/crn%3Av1%3Abluemix%3Apublic%3Adashdb-for-transactions%3Aus-south%3Aa%2Ff1ce102a564c4690a003d7da2a8486aa%3Ae8e998e7-37af...

IBM Db2 on Cloud

Error: Please check network connectivity then try again.

* Untitled - 1

```

create table user(id int GENERATED BY DEFAULT AS IDENTITY NOT NULL, name varc

alter table user add age int;
select * from hai;
drop table user;

create table users(username varchar(30), email varchar(30), password varchar(

insert into users values('123','jetson@gmail.com', '12345');
insert into users values('124','joshua@gmail.com', '12345');
select * from users;
```

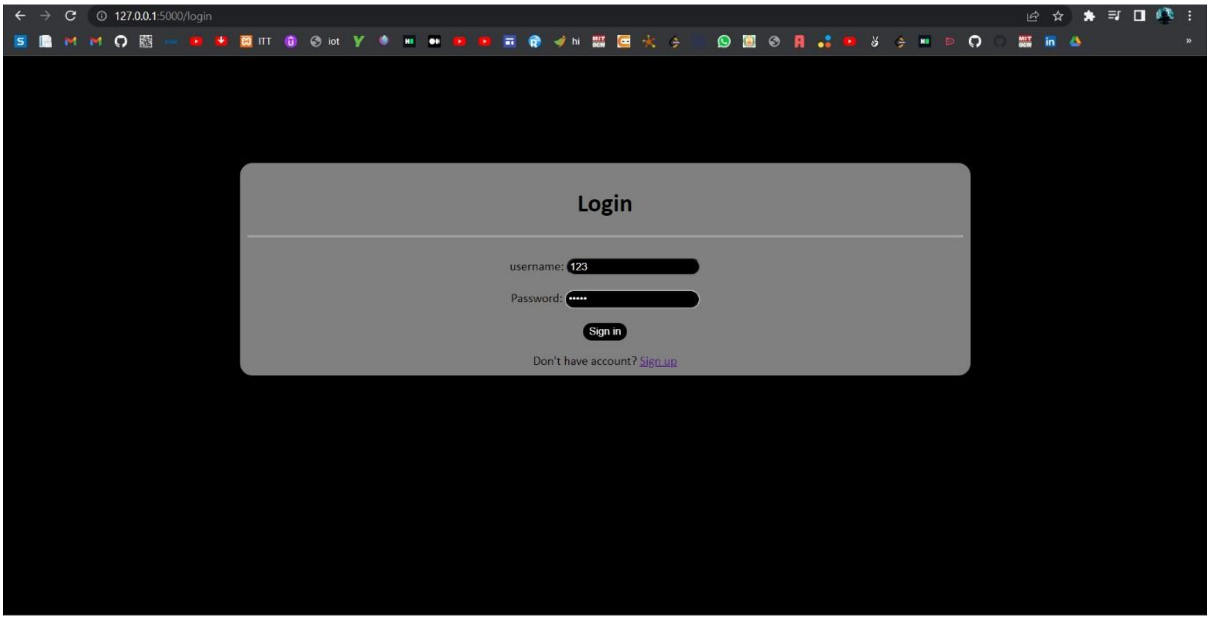
Result - Oct 9, 2022 9:25:17 PM

select * from users — Run time: 0.019 s

Result set 1

USERNAME	EMAIL	PASSWORD
124	joshua@gmail.com	12345
123	jetson@gmail.com	12345
126	rj@gmail.com	12345

With correct credentials:

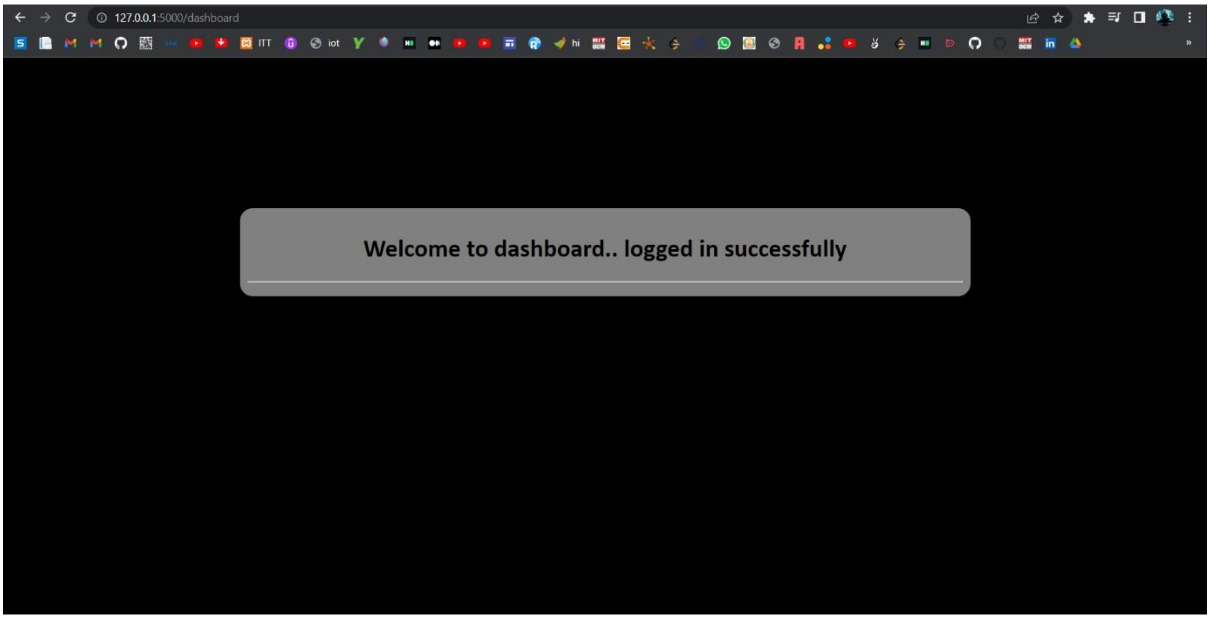


Login

username: 123

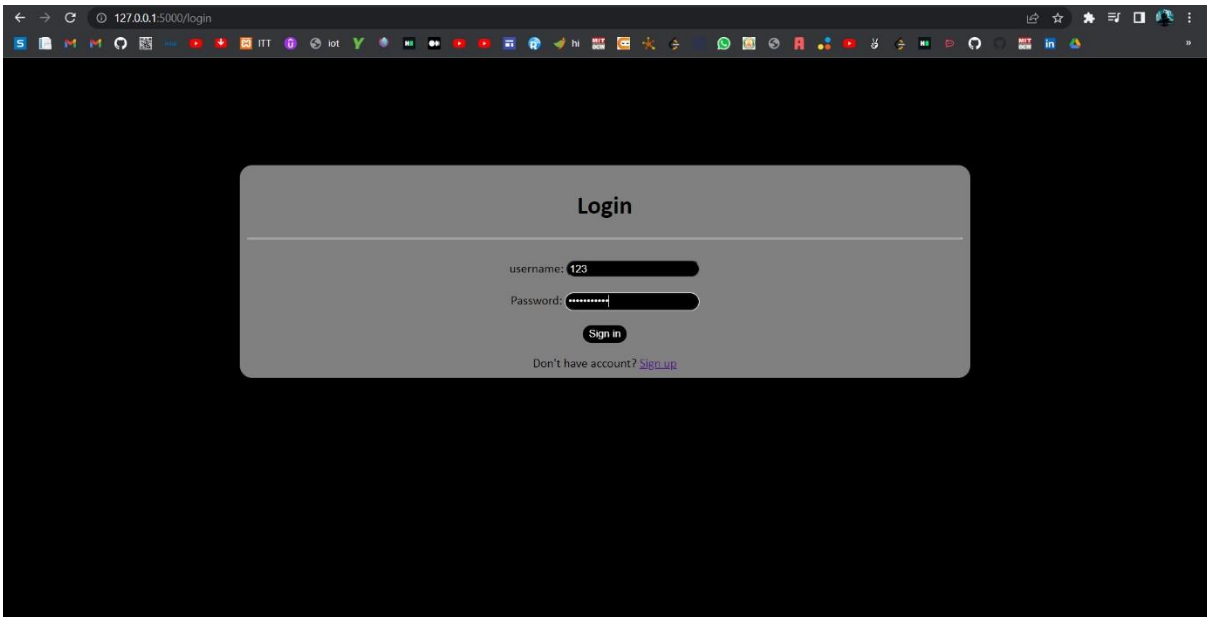
Password: •••••

Don't have account? Sign up



Welcome to dashboard.. logged in successfully

With wrong credentials:



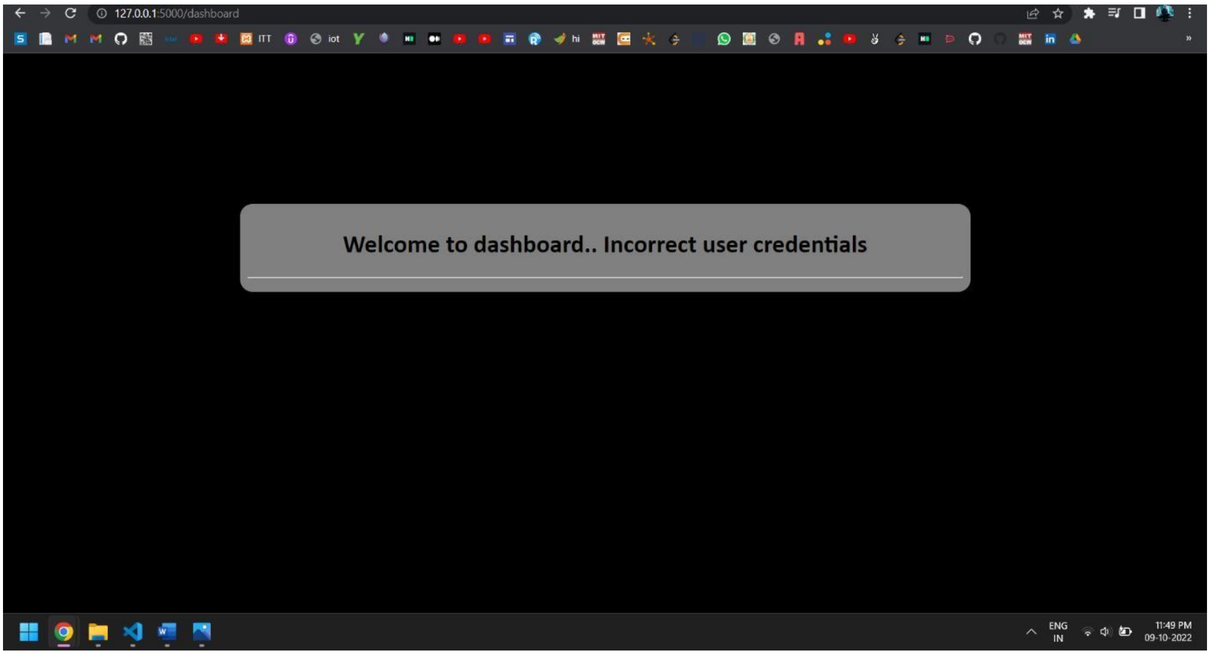
127.0.0.1:5000/login

Login

username: 123

Password: ••••••••••

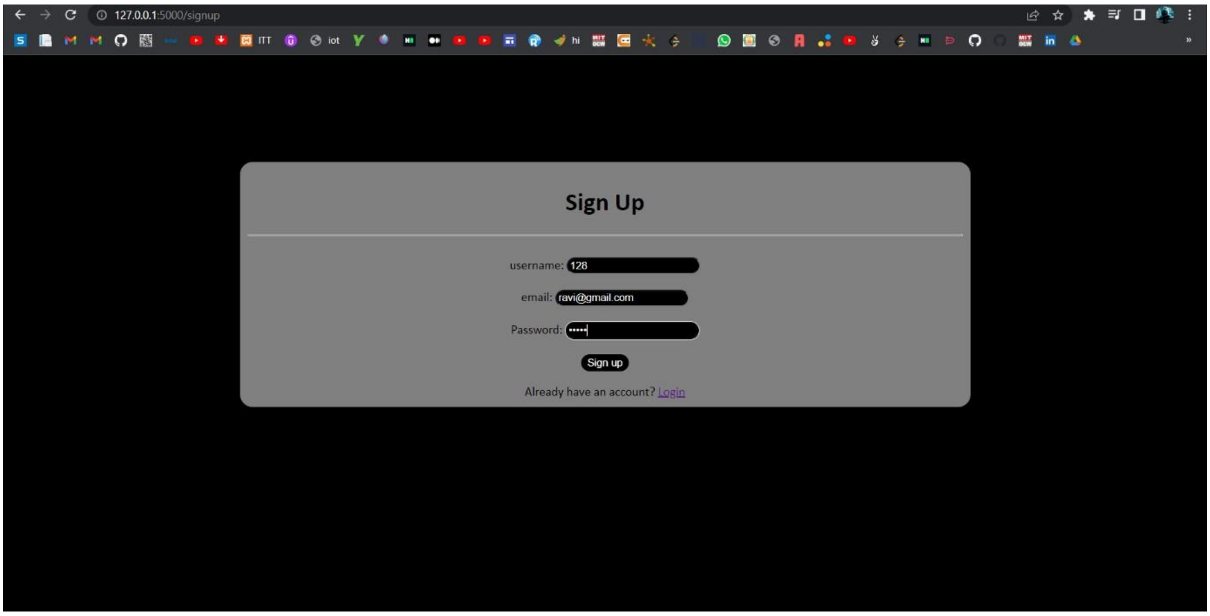
Don't have account? Sign up



127.0.0.1:5000/dashboard

Welcome to dashboard.. Incorrect user credentials

Sign up:



127.0.0.1:5000/signup

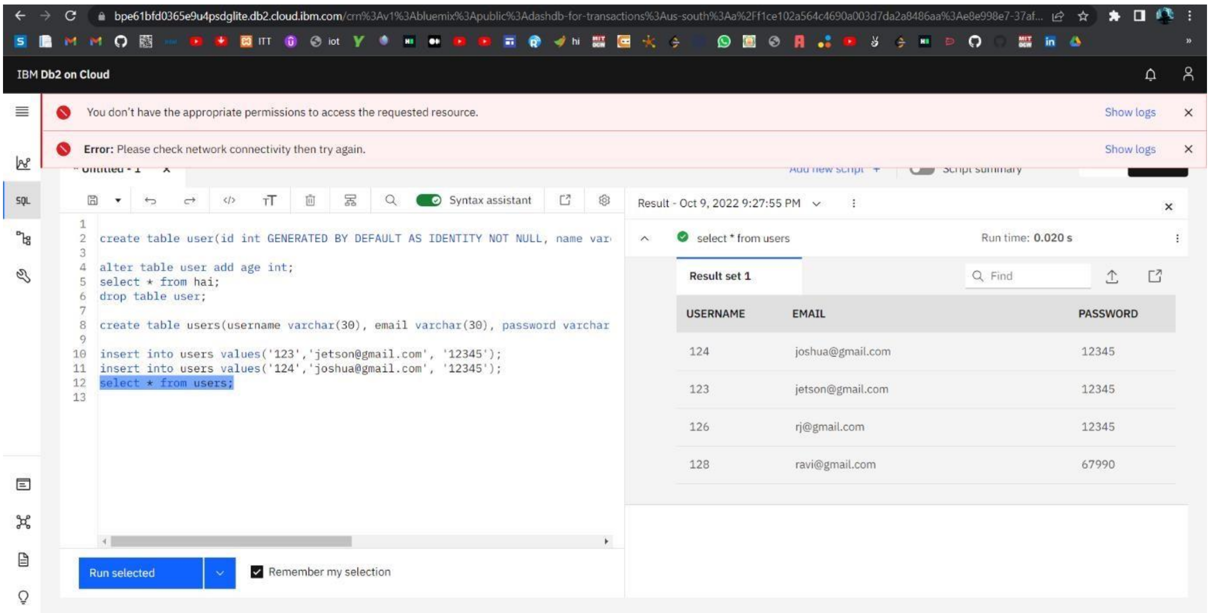
Sign Up

username: 128

email: ravi@gmail.com

Password: •••••

Already have an account? Login



bpe61bfd0365e9u4psdglite.db2.cloud.ibm.com/crn%3Av1%3Abluemix%3Apublic%3Adashdb-for-transactions%3Aus-south%3Aa%2Ff1ce102a564c4690a003d7da2a8486aa%3Ae8e998e7-37af...

You don't have the appropriate permissions to access the requested resource.

Error: Please check network connectivity then try again.

```
create table user(id int GENERATED BY DEFAULT AS IDENTITY NOT NULL, name var

alter table user add age int;
select * from hai;
drop table user;

create table users(username varchar(30), email varchar(30), password varchar

insert into users values('123','jetson@gmail.com', '12345');
insert into users values('124','joshua@gmail.com', '12345');
select * from users;
```

Result - Oct 9, 2022 9:27:55 PM

select * from users   Run time: 0.020 s

Result set 1

USERNAME	EMAIL	PASSWORD
124	joshua@gmail.com	12345
123	jetson@gmail.com	12345
126	rj@gmail.com	12345
128	ravi@gmail.com	67990