**ASSIGNMENT-2**

|  |  |
| --- | --- |
| Student Name | Ms.V.MERCY |
| Student Roll Number | AC19UIT026 |
| Assignment Date | September 19th |

***Question-1****: Create User table with user with email, username, roll number, password*

<!DOCTYPE html>

<html>

<head>

<title>Form</title><style>

.data{

font-size: x-large;

font-weight: bolder;

}</style> </head>

<body style=”background-color: #d69cdc;”>

<center>

<form action=””>

<p class=”data”>Data collection form</p><br/><br/><br/><br/><br/><br/><br/>

<label for=”uname”>Enter your name:</label>

<input style="margin-left: 150px;" type="text" name="uname" placeholder="Enter name"/><br/>

<label for="Regno">Enter your Register Number:</label>

<input style="margin-left: 75px;" type="text" name="Regno" placeholder="Enter Register Number"/><br/>

<label type="email">Enter your email:</label>

<input style="margin-left: 148px;" type="email" name="email" placeholder="Enter email"/><br/>

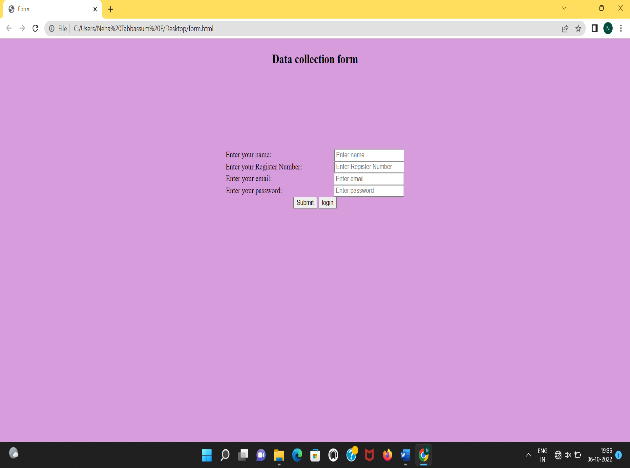
<label for="pwd">Enter your password:</label>

<input style="margin-left: 123px;" type="password" name="pwd" placeholder="Enter password" /><br>

<input type="submit" name="submit">

<button type="submit" name="submit">login</button>

</form> </center> </body> </html>

****

***Question-2:*** *Perform UPDATE, DELETE Queries with user table*

CREATE TABLE EMPLOYEE (

empId INTEGER PRIMARY KEY,

name TEXT NOT NULL,

dept TEXT NOT NULL

);

INSERT INTO EMPLOYEE VALUES (23, 'Kavya', 'Student');

INSERT INTO EMPLOYEE VALUES (24, 'Menaga', 'Student');

INSERT INTO EMPLOYEE VALUES (26, 'Mercy', 'Student');

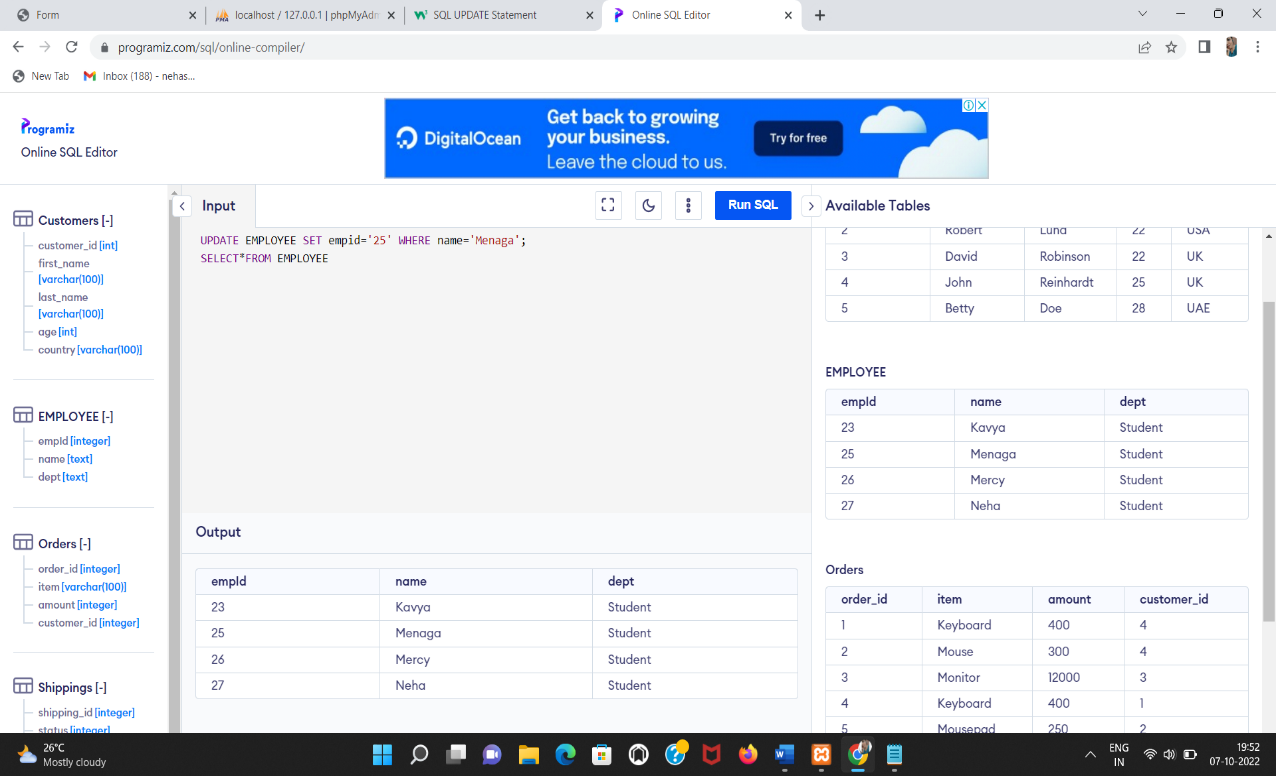
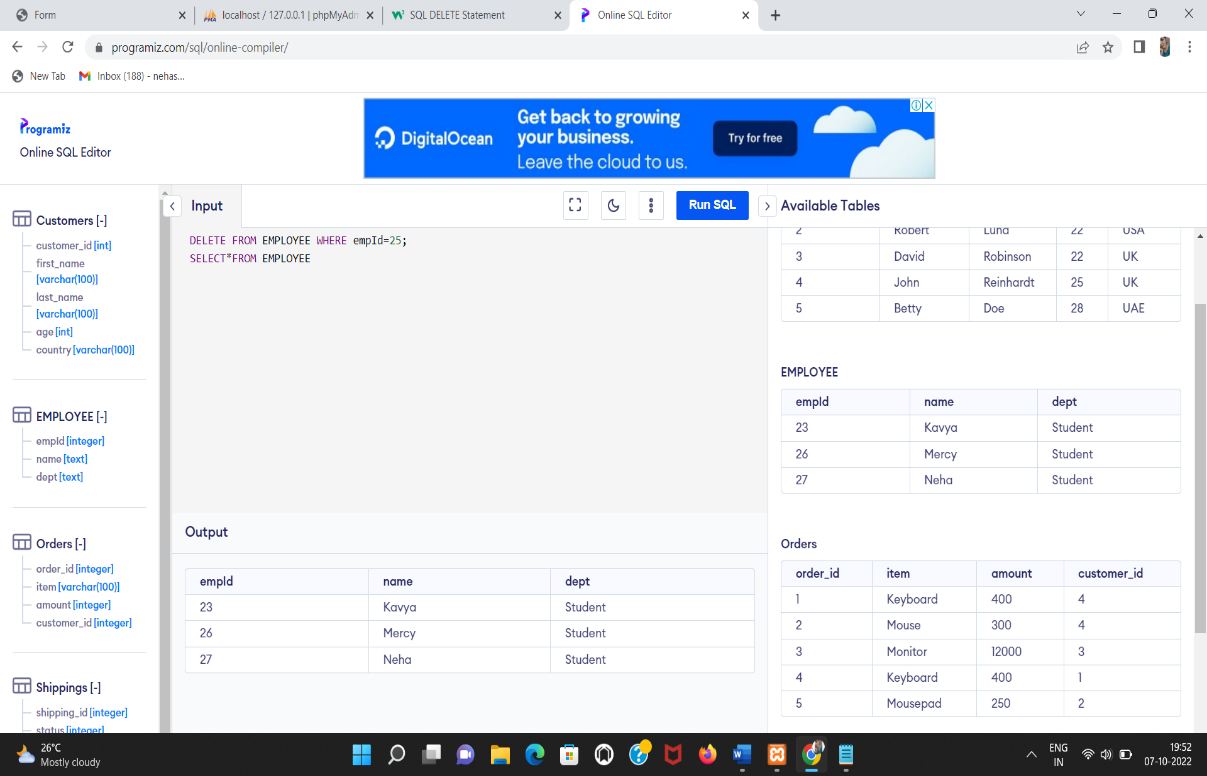
INSERT INTO EMPLOYEE VALUES (27, 'Neha', 'Student');

UPDATE EMPLOYEE

SET empId= '25',

WHERE name = ‘Menaga’;

DELETE FROM EMPLOYEE WHERE empId=25;

***Question-3:*** *Connect Python code to db2*

*#You need to install ibm\_db using pip*

pip install ibm\_db

*#Creation of connection object*

import ibm\_db

conn=ibm\_db.connect(“database”,”username”,”password”)

from ibm\_db import connect

connection = connect(‘DATABASE=,database name>;’

‘HOSTNAME=<database ip>;’

‘PORT=<database port>;’

‘PROTOCOL=TCPIP;’

‘UID=<database username>;’

‘PWD=<username password>;’,’’,’’)

*#Calling fetch methods*

def results(command):

from ibm\_db import fetch\_assoc

ret=[]

result=fetch\_assoc(command)

while result:

ret.append(result)

result=fetch\_assoc(command)

return ret

***Question-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 authenticates user username and password. If the user is valid show the welcome page*

**Create a python file with name app.py**

from flask import Flask, render\_template, request, redirect, url\_for, session

from flask\_mysqldb import MySQL

import MySQLdb.cursors

import re

app = Flask(\_\_name\_\_)

app.secret\_key = 'your secret key'

app.config['MYSQL\_HOST'] = 'localhost'

app.config['MYSQL\_USER'] = 'root'

app.config['MYSQL\_PASSWORD'] = 'your password'

app.config['MYSQL\_DB'] = 'geeklogin'

mysql = MySQL(app)

@app.route('/')

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

def login():

msg = ''

if request.method == 'POST' and 'username' in request.form and 'password' in request.form:

username = request.form['username']

password = request.form['password']

cursor = mysql.connection.cursor(MySQLdb.cursors.DictCursor)

cursor.execute('SELECT \* FROM accounts WHERE username = % s AND password = % s', (username, password, ))

account = cursor.fetchone()

if account:

session['loggedin'] = True

session['id'] = account['id']

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

msg = 'Logged in successfully !'

return render\_template('index.html', msg = msg)

else:

msg = 'Incorrect username / password !'

return render\_template('login.html', msg = msg)

@app.route('/logout')

def logout():

session.pop('loggedin', None)

session.pop('id', None)

session.pop('username', None)

return redirect(url\_for('login'))

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

def register():

msg = ''

if request.method == 'POST' and 'username' in request.form and 'password' in request.form and 'email' in request.form :

username = request.form['username']

password = request.form['password']

email = request.form['email']

cursor = mysql.connection.cursor(MySQLdb.cursors.DictCursor)

cursor.execute('SELECT \* FROM accounts WHERE username = % s', (username, ))

account = cursor.fetchone()

if account:

msg = 'Account already exists !'

elif not re.match(r'[^@]+@[^@]+\.[^@]+', email):

msg = 'Invalid email address !'

elif not re.match(r'[A-Za-z0-9]+', username):

msg = 'Username must contain only characters and numbers !'

elif not username or not password or not email:

msg = 'Please fill out the form !'

else:

cursor.execute('INSERT INTO accounts VALUES (NULL, % s, % s, % s)', (username, password, email, ))

mysql.connection.commit()

msg = 'You have successfully registered !'

elif request.method == 'POST':

msg = 'Please fill out the form !'

return render\_template('register.html', msg = msg)

**Create a html file index.html**

*<!-- Store this code in 'index.html' file inside the 'templates' folder-->*

*<html>*

*<head>*

*<meta charset="UTF-8">*

*<title> Index </title>*

*<link rel="stylesheet" href="{{ url\_for('static', filename='style.css') }}">*

*</head>*

*<body></br></br></br></br></br>*

*<div align="center">*

*<div align="center" class="border">*

*<div class="header">*

*<h1 class="word">Index</h1>*

*</div></br></br></br>*

*<h1 class="bottom">*

*Hi {{session.username}}!!</br></br> Welcome to the index page...*

*</h1></br></br></br>*

*<a href="{{ url\_for('logout') }}" class="btn">Logout</a>*

*</div>*

*</div>*

*</body>*

*</html>*

**Login.html**

<!-- Store this code in 'login.html' file inside the 'templates' folder -->

<html>

<head>

<meta charset="UTF-8">

<title> Login </title>

<link rel="stylesheet" href="{{ url\_for('static', filename='style.css') }}">

</head>

<body></br></br></br></br></br>

<div align="center">

<div align="center" class="border">

<div class="header">

<h1 class="word">Login</h1>

</div></br></br></br>

<h2 class="word">

<form action="{{ url\_for('login') }}" method="post">

<div class="msg">{{ msg }}</div>

<input id="username" name="username" type="text" placeholder="Enter Your Username" class="textbox"/></br></br>

<input id="password" name="password" type="password" placeholder="Enter Your Password" class="textbox"/></br></br></br>

<input type="submit" class="btn" value="Sign In"></br></br>

</form>

</h2>

<p class="bottom">Don't have an account? <a class="bottom" href="{{url\_for('register')}}"> Sign Up here</a></p>

</div>

</div>

</body>

</html>

**register.html**

<!-- Store this code in 'register.html' file inside the 'templates' folder -->

<html>

<head>

<meta charset="UTF-8">

<title> Register </title>

<link rel="stylesheet" href="{{ url\_for('static', filename='style.css') }}">

</head>

<body></br></br></br></br></br>

<div align="center">

<div align="center" class="border">

<div class="header">

<h1 class="word">Register</h1>

</div></br></br></br>

<h2 class="word">

<form action="{{ url\_for('register') }}" method="post">

<div class="msg">{{ msg }}</div>

<input id="username" name="username" type="text" placeholder="Enter Your Username" class="textbox"/></br></br>

<input id="password" name="password" type="password" placeholder="Enter Your Password" class="textbox"/></br></br>

<input id="email" name="email" type="text" placeholder="Enter Your Email ID" class="textbox"/></br></br>

<input type="submit" class="btn" value="Sign Up"></br>

</form>

</h2>

<p class="bottom">Already have an account? <a class="bottom" href="{{url\_for('login')}}"> Sign In here</a></p>

</div>

</div>

</body>

</html>

**style.css**

.header{

padding: 5px 120px;

width: 150px;

height: 70px;

background-color: #236B8E;

}

.border{

padding: 80px 50px;

width: 400px;

height: 450px;

border: 1px solid #236B8E;

border-radius: 0px;

background-color: #9AC0CD;

}

.btn {

padding: 10px 40px;

background-color: #236B8E;

color: #FFFFFF;

font-style: oblique;

font-weight: bold;

border-radius: 10px;

}

.textbox{

padding: 10px 40px;

background-color: #236B8E;

color: #FFFFFF;

border-radius: 10px;

}

::placeholder {

color: #FFFFFF;

opacity: 1;

font-style: oblique;

font-weight: bold;

}

.word{

color: #FFFFFF;

font-style: oblique;

font-weight: bold;

}

.bottom{

color: #236B8E;

font-style: oblique;

font-weight: bold;

}

