

Assignment -1
Python Programming

Assignment Date	27 September 2022
Student Name	Athi
Student Roll Number	95071912015
Maximum Marks	2 Marks

Question-1:

Create registration page in html with username, Email and phone number and by using POST method display it in next html page

Solution:

```
<html>
```

```
<head>
```

```
<script>
```

```
function Validation() {
```

```
    var name = document.forms.RegForm.Name.value;
```

```
    var email = document.forms.RegForm.EMail.value;
```

```
    var phone = document.forms.RegForm.Telephone.value;
```

```
var regEmail=/^\w+([\.-]?\w+)*@\w+([\.-]?\w+)*(\.\w{2,3})+$/g;
```

```
var regPhone=/^\d{10}$/;
```

```
var regName = /\d+$/g;
```

```
if (name == "" || regName.test(name)) {
```

```
    window.alert("Please enter your name properly.");
```

```
    name.focus();
```

```
    return false;
```

```
}
```

```
if (email == "" || !regEmail.test(email)) {
```

```
    window.alert("Please enter a valid e-mail address.");
```

```
    email.focus();
```

```
    return false;
```

```
}
```

```
if (phone == "" || !regPhone.test(phone)) {
```

```
    alert("Please enter valid phone number.");
```

```
    phone.focus();
```

```
    return false;
```

```
}
```

```
if (what.selectedIndex == -1) {
```

```
    alert("Please enter your course.");
```

```
    what.focus();
```

```
    return false;
```

```
}
```

```
    return true;
```

```
}
```

```
</script>
```

```
<style>
```

```
div {  
  
    box-sizing: border-box;  
  
    width: 100%;  
  
    border: 100px solid black;  
  
    float: left;  
  
    align-content: center;  
  
    align-items: center;  
  
}
```

```
form {  
  
    margin: 0 auto;  
  
    width: 600px;  
  
}
```

```
</style>
```

```
</head>
```

<body>

<h1 style="text-align: center;">REGISTRATION FORM</h1>

<form name="RegForm" onsubmit="return Validation()" method="post">

<p>Name: <input type="text" size="65" name="Name" /></p>

<p>E-mail Address: <input type="text" size="65" name="EMail" /></p>

<p>Telephone: <input type="text" size="65" name="Telephone" /></p>
 <p>

SELECT YOUR COURSE

<select type="text" value="" name="Subject">

<option>BTECH</option>

<option>BBA</option>

<option>BCA</option>

<option>B.COM</option> </select>

</p>

<p>Comments: <textarea cols="55" name="Comment"> </textarea></p>

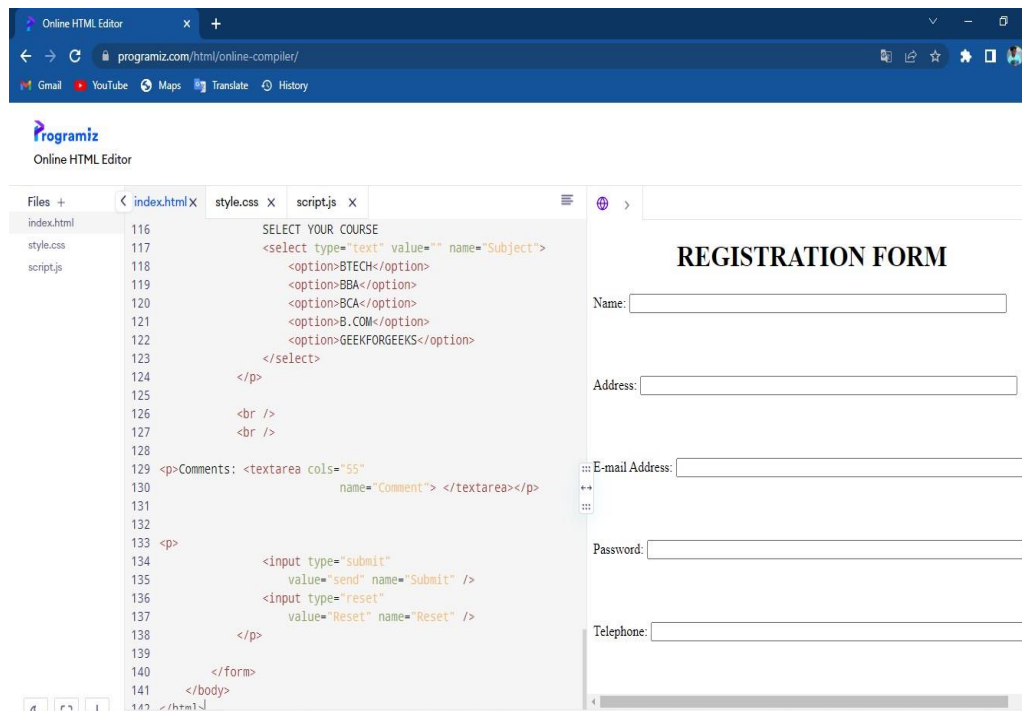
<p>

<input type="submit" value="send" name="Submit" />

<input type="reset" value="Reset" name="Reset" /> </p>

</form>

</body> </html>



Question-2:

Develop a flask program which should contain atleast 5 packages used from pypi.org.

Solution:

```
from flask import Flask, render_template, request, redirect, url_for, session
```

```
import ibm_db
```

```
import re
```

```
app = Flask(__name__)
```

```
app.scret_key = 'a'
```

```
conn =
```

```
ibm_db.conect("DATABASE=;HOSTNAME=;PORT=;SECURITY=SSL;SSL  
ServerCertificate=;UID=;PhD=", ' ', ' ')
```

```
@app.route('/')
```

```
def home():
```

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

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

```
def login():
```

```
    global userid
```

```
    msg = ''
```

```
    if request.method == 'POST':
```

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

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

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

```
        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']

else:

    msg = 'Incorrect username/password'

    return render_template('login.html', msg=msg)
```

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

```
def register():
```

```
    if request.method == 'POST':
```

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

```
        email = request.form['email']
```

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

```

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 = "Account already exists!"

elif not re.match(r'^[a-zA-Z0-9_]+@[a-zA-Z0-9_]+\.[a-zA-Z0-9_]+$', email):

    msg = "format does not match"

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

    msg = "name must contain characters and numbers"

else:

    insert_sql = "INSERT INTO users VALUES(?, ?, ?)"

    prep_stmt = ibm_db.prepare(conn, insert_sql)

    ibm_db.bind_param(prepare_stmt, 1, username)

    ibm_db.bind_param(prepare_stmt, 2, email)

    ibm_db.bind_param(prepare_stmt, 3, password)

    ibm_db.execute(prepare_stmt)

    msg = "You have successfully registered"

elif (request.method == "POST"):

    msg == "Please fill out the form"

```

```
return render_template('register.html', msg=msg)
```

```
@app.route('/dashboard')
```

```
def dash():
```

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

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

```
def app():
```

```
    msg = ''
```

```
    if request.method == "POST":
```

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

```
        email = request.form['email']
```

```
        qualification = request.form['qualification']
```

```
        skills = request.form['skills']
```

```
        jobs = request.form['s']
```

```
        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 = "there is only 1 job position"
```

```
    return render_template('apply.html', msg=msg)
```

```
insert_sql = "INSERT INTO job VALUES(?, ?, ?, ?, ?)"
```

```
prep_stmt = ibm_db.prepare(conn, insert_sql)
```

```
ibm_db.bind_param(prepare_stmt, 1, username)
```

```
ibm_db.bind_param(prepare_stmt, 2, email)
```

```
ibm_db.bind_param(prepare_stmt, 3, qualification)
```

```
ibm_db.bind_param(prepare_stmt, 4, skills)
```

```
ibm_db.bind_param(prepare_stmt, 5, jobs)
```

```
ibm_db.execute(prepare_stmt)
```

```
msg = "You have successfully applied for job"
```

```
session['LoggedIn'] = True
```

```
TEXT = "Hello user,a new application for job position" + job + isrequested
```

```
"
```

```
elif request.method == "POST"
```

```
    msg = "Please fill out the form"
```

```
    return render_template('register.html', msg=msg)
```

```
@app.route('/display')
```

```
def display():
```

```
    print
```

```
    session["username"], session['id']
```

```
    cursor = mysql.connection.cursor()
```

```
    cursor.execute('SELECT*FROM job WHERE userid=%s', (sessio['id'],))
```

```
    account = cursor.fetchone()
```

```
    print("accountdisplay", account)
```