SPRINT-1

REGISTRATION PAGE

REGISTER.HTML

```
<html>
<head>
<meta charset="UTF-8">
<title> Register </title>
<link rel="stylesheet" href="static/stlll.css">
</head>
<body></br></br></br></br>
<div align="center">
<div align="center" class="border">
<div class="header">
<h1 class="word">Register</h1>
</div>
<h2 class="word">
<form action="{{ url_for('register') }}" method="post">
<div class="msg">{{ msg }}</div>
<label>Username</label>
<input id="username" name="username" type="text" placeholder="Enter
YourUsername" class="textbox"/>
<label>Password</label>
<input id="password" name="password" type="password"
placeholder="Enter YourPassword" class="textbox"/>
<label>Email ID</label>
```

```
<input id="email" name="email" type="text" placeholder="Enter Your
Email ID"class="textbox"/>
<input type="submit" class="btn" value="Sign Up">
</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>
```

STLLL.CSS

```
@import url('http://fonts.googleapis.com/css2?family=Poppins&display=swap');
*{
   margin: 0;
   padding: 0;
   box-sizing: border-box;
   font-family: 'Poppins',sans-
   serif;cursor: pointer;
}
body{
   height:
   100vh;width:
    100%;
    display: flex;
    justify-content:
    center;align-items:
    center;
    background: linear-gradient(to right,#3f5efb,#fc466b);
}
form{
    width:
    25rem;
    height:
    28rem;
    display: flex;
    flex-direction: column;
    background: rgba(255, 255, 255, .06);
    box-shadow: 0 8px 32px 0
    rgba(31,38,135,.37);border-radius: 30px;
    border-left: 1px solid rgba(255, 255, 255, .3);
    border-top: 1px solid rgba(255, 255, 255, .3);

}
h1
```

```
{
   font-size: 50px;
   text-align:
   center;color:
   white;
   text-shadow: 2px 2px 4px rgba(0, 0, 0,
   .2);letter-spacing: 3px;
   margin-bottom: 2%;
   opacity: .7;
}
label{
   font-size:
   20px; color:
   white; margin-
   left: 3%;
   opacity: .8;
   text-shadow: 2px 2px 4px rgba(0, 0, 0, .2);
}
input{
   width: 80%;
   margin: 5% auto;
   margin-bottom:
   8%;border: none;
   outline: none;
   background:
   transparent;color:
```

```
white;

border-bottom: 1px solid rgba(255, 255, 255, .6);
```

```
}

.btn

{

    width: 50%;

    margin: 3%

    auto;color:

    white; font-

    size: 15px;

    opacity: .7;

    background: rgba(255, 255, 255,

    .06);padding: 10px 30px;

    border: none;

    outline: none;

    border-radius:

    20px;

    text-shadow: 2px 2px 4px rgba(255, 255,

    255, .6);box-shadow: 3px 3px 5px

    rgba(31,38,135,.37); border-left: 1px solid

    rgba(255, 255, 255, .3);

    border-top: 1px solid rgba(255, 255, 255, .3);

}
```

app.py:

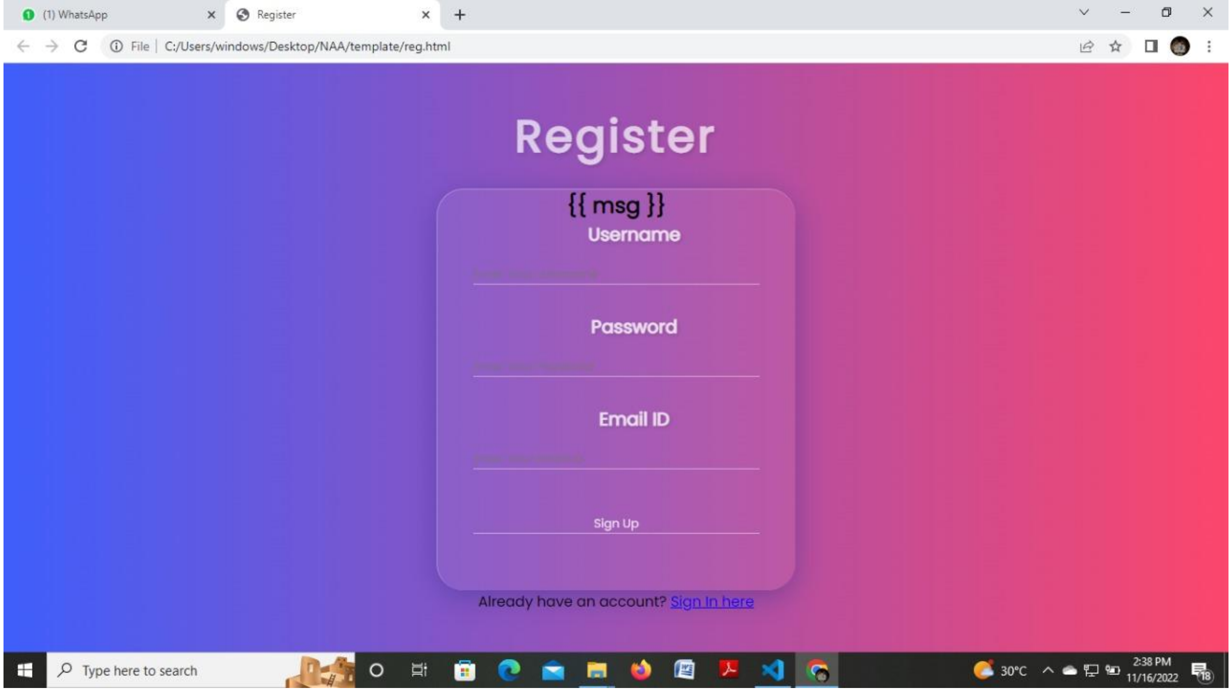
```
from flask import Flask, render_template, request, redirect, url_for, session
import ibm_db
import re
app = Flask( __name__)

app.secret_key = 'a'
```

```
conn = ibm_db.connect("DATABASE=bludb;HOSTNAME=b70af05b-76e4-4bca-a1f5-
23dbb4c6a74e.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=32716;Security=
SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=pbv90190;PWD=pEihUI0HD05i
gFYt;",'','')

@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']
    stmt = ibm_db.prepare(conn,'SELECT * FROM user WHERE username = ? AND
password = ?')
    ibm_db.bind_param(stmt,1,username)
    ibm_db.bind_param(stmt,2,password)
    ibm_db.execute(stmt)
    account = ibm_db.fetch_assoc(stmt)
    if account:
      session['loggedin'] = True
      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']
    stmt = ibm_db.prepare(conn,'SELECT * FROM user WHERE username = ?')
    ibm_db.bind_param(stmt,1,username)
    ibm_db.execute(stmt)
    account = ibm_db.fetch_assoc(stmt)
    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):
```

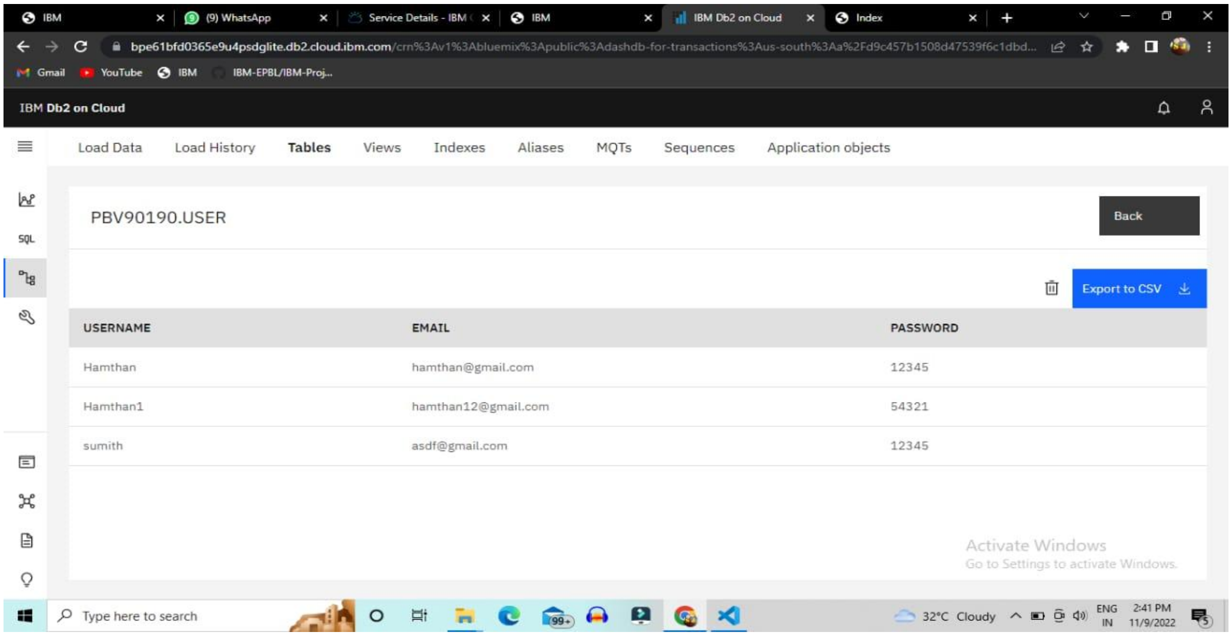
```
                msg = 'Username must contain only characters and numbers !'
        elif not username or not password or not email:
            msg = 'Please fill out the form !'
        else:
            prep_stmt = ibm_db.prepare(conn,"INSERT INTO user VALUES(?, ?, ?)")
            ibm_db.bind_param(prep_stmt, 1, username)
            ibm_db.bind_param(prep_stmt, 2, email)
            ibm_db.bind_param(prep_stmt, 3, password)
            ibm_db.execute(prep_stmt)
            msg = 'You have successfully registered !'
    elif request.method == 'POST':
        msg = 'Please fill out the form !'
    return render_template('register.html', msg = msg)

if __name__ == '__main__':
    app.debug = True
    app.run(host='0.0.0.0',port=8080)
```

OUTPUT:

REGISTER:



DATABASE:



PBV90190.USER

USERNAME	EMAIL	PASSWORD
Hamthan	hamthan@gmail.com	12345
Hamthan1	hamthan12@gmail.com	54321
sumith	asdf@gmail.com	12345