

M.Ramya

820119104015

Cloud computing

News Tracking Application

Ramyamahalingam46@gmail.com

ASSIGNMENT-3:

1. Create a Bucket in IBM object storage.
2. Upload an 5 images to IBM object storage and make it public. Write html code to displaying all the 5 images.
3. Upload a CSS page to the object storage and use the same page in your HTML code.
4. Design a chatbot using IBM Watson assistant for hospital. Ex: User comes with query to know the Branches for that hospital in your city. Submit the web URL of that chat bot as aassignment.
5. Create Watson assistant service with 10 steps and use 3 conditions in it. Load that scriptin HTML Page.

SOURCE CODE:

Assign3.py

```
From flask import Flask, render_template, request, redirect, url_for,
sessionImport ibm_db

Import
bcryptConn

=

Ibm_db.connect("DATABASE=bludb;HOSTNAME=;PORT=;SECURITY=SSL;S
SLServerCertifi cate=DigiCertGlobalRootCA.crt;UID=;PWD=",',',')

# url_for('static', filename='style.css')

App = Flask( name )

App.secret_key =

b'_5#y2L"F4Q8z\n\xec]/'

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

Def home():

If 'email' not in session:

Return redirect(url_for('login'))

Return

render_template('home.html',name='Home')

@app.route("/register",methods=['GET','PO
ST']) Def register(): if

Request.method == 'POST':

Email = request.form['email']

Username =

Request.form['username']rollNo
```

```

= request.form['rollNo']

password =
request.form['password']

if not email or not username or not rollNo or not password:
return render_template('register.html',error='Please fill
all fields')hash=bcrypt.hashpw(password.encode('utf-
8'),bcrypt.gensalt()) query = "SELECT * FROM USER
WHERE email=? OR rollNo=?" stmt =

ibm_db.prepare(conn, query)

ibm_db.bind_param(stmt,1,email)

ibm_db.bind_param(stmt,2,rollNo)

ibm_db.execute(stmt)


isUser =

ibm_db.fetch_assoc(stmt)if
not isUser:

insert_sql = "INSERT INTO User(username,email,PASSWORD,rollNo) 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, hash)

Ibm_db.bind_param(prepare_stmt, 4, rollNo)

```

```

Ibm_db.execute(prepare_stmt) return
Render_template('register.html',success="You can
login")Else:
Return render_template('register.html',error='Invalid
Credentials')Return
render_template('register.html',name='Home')
@app.route("/login",methods=['GET','POST'])
Def login():
If request.method == 'POST':
Email = request.form['email']
Password =
request.form['password']If not
email or not password:
Return render_template('login.html',error='Please fill all
fields')Query = "SELECT * FROM USER WHERE
email=?"
Stmt = ibm_db.prepare(conn,
query)
Ibm_db.bind_param(stmt,1,ema
il) Ibm_db.execute(stmt) isUser
= Ibm_db.fetch_assoc(stmt)
Print(isUser,password)
If not isUser:
Return render_template('login.html',error='Invalid Credentials')isPasswordMatch =
bcrypt.checkpw(password.encode('utf-

```

```
8'),isUser['PASSWORD'].encode('utf-8'))
```

```
If not isPasswordMatch:
```

```
Return render_template('login.html',error='Invalid Credentials')
```

```
Session['email'] =
```

```
isUser['EMAIL']Return
```

```
redirect(url_for('home'))
```

```
Return
```

```
render_template('login.html',name='Home')
```

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

```
Logout():
```

```
Session.pop('email', None) return
```

```
Redirect(url_for('login'))
```

Homepage.html

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8" />
<meta http-equiv="X-UA-Compatible" content="IE=edge" />
<meta name="viewport" content="width=device-width, initial-scale=1.0" />
<title>Document</title>
</head>
<body>
WELCO
ME
<button><a href="/logout">LOGOUT</a></button>
</body>
</html>
```

Loginpage.html

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8" />
<meta http-equiv="X-UA-Compatible" content="IE=edge" />
<meta name="viewport" content="width=device-width, initial-scale=1.0" />
<title>Login</title>
</head>
```

```
<body>
<h3>Login</h3>
<form method="POST">
<input type="email" name="email" placeholder="Email" required />
<input type="password" name="password" placeholder="Password" required />
<button type="submit">Login</button>
</form>
<p>{{ success }}</p>
<p style="color: red">{{ error }}</p>
<a href="/register">Don't have an account? Register</a> </body>
</html>
```

Registerpage.html

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8" />
<meta http-equiv="X-UA-Compatible" content="IE=edge" />
<meta name="viewport" content="width=device-width, initial-scale=1.0" />
<title>Login</title>
</head>
<body>
<h3>Login</h3>
<form method="POST">
<input type="email" name="email" placeholder="Email" required />
<input type="password" name="password" placeholder="Password" required />
<button type="submit">Login</button>
```

</form>

<p>{{ success }}</p>

<p style="color: red">{{ error }}</p>

Don't have an account? Register </body>

</html>

