

Assignment -2

IBM CLOUD DB2

| | |
|---------------------|-----------------|
| Assignment Date | 26 OCTOBER 2022 |
| Student Name | PRAVEEN V |
| Student Roll Number | 713319CS109 |
| Maximum Marks | 2 Marks |

Questions:

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

The image shows two screenshots of the IBM Db2 on Cloud web interface. The top screenshot displays the 'Tables' tab with a list of tables: PROTBL and USERTBL, both in the MVG77822 schema. The 'USERTBL' table is selected, and its definition is shown on the right. The table has three columns: USERNAME (CHAR, 244, nullable), EMAIL (CHAR, 244, nullable), and PASSWORD (CHAR, 244, nullable). The bottom screenshot shows the 'View data' page for the MVG77822.USERTBL table. It displays three rows of data:

| USERNAME | EMAIL | PASSWORD |
|----------|-----------------|----------|
| Admin | admin@gmail.com | Admin |
| kiki | kiki@gmail.com | kiki@1 |
| mad | mad@gmail.com | maddy |

2. Perform UPDATE, DELETE Queries with user table.

UPDATE TABLE:

IBM Db2 on Cloud

Data objects

Filter objects

MVG77822

*Untitled - 1

1 UPDATE USERTBL

2 SET EMAIL = 'madhave.susheel@gmail.com'

3 WHERE USERNAME = 'mad';

History

Find history

| Script | Date | Status | Runtime |
|---|------------------------|--------|---------|
| Untitled - 1 | Nov 4, 2022 3:28:06 PM | 1 | 0.005 s |
| UPDATE USERTBL SET EMAIL = 'madhave.susheel@gmail.com' WHERE USERNAME = 'mad' | | | 0.005 s |

IBM Db2 on Cloud

Load Data

Load History

Tables

Views

Indexes

Aliases

MQTs

Sequences

Application objects

MVG77822.USERTBL

Back

Export to CSV

| USERNAME | EMAIL | PASSWORD |
|----------|---------------------------|----------|
| kiki | kiki@gmail.com | kiki@1 |
| mad | madhave.susheel@gmail.com | maddy |

DELETE TABLE:

The screenshot shows the IBM Db2 on Cloud web interface. The top navigation bar includes 'Load Data', 'Load History', 'Tables', 'Views', 'Indexes', 'Aliases', 'MQTs', 'Sequences', and 'Application objects'. The 'Tables' tab is selected, and the table 'MVG77822.USER_TBL' is displayed. The table has three columns: 'USERNAME', 'EMAIL', and 'PASSWORD'. There are two rows of data: one for 'kiki' with email 'kiki@gmail.com' and password 'kiki@1', and another for 'madi' with email 'madi@gmail.com' and password 'maddy'. A 'Back' button is in the top right, and an 'Export to CSV' button is in the top right of the table area.

| USERNAME | EMAIL | PASSWORD |
|----------|----------------|----------|
| kiki | kiki@gmail.com | kiki@1 |
| madi | madi@gmail.com | maddy |

3. Connect python code to db2.

NOTE:- Question 4 contains Question 3 answer

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

Answers

Connect.py

```
dsn_hostname = "8e359033-a1c9-4643-82ef-
8ac06f5107eb.bs2io90l08kqb1od8lcg.databases.appdomain.cloud"
dsn_uid = "mvg77822"
dsn_pwd = "X0k7hJ5iGB6Dzgtx"
dsn_driver = "{IBM DB2 ODBC DRIVER}"
dsn_database = "BLUDB"
dsn_port = "30120"
dsn_security = "SSL"
dsn = ("DRIVER={0};"
"DATABASE={1};"
"HOSTNAME={2};"
"PORT={3};"
"UID={4};"
"PWD={5};"
"SECURITY={6};").format(dsn_driver,dsn_database,dsn_hostname,dsn_port,dsn_uid,dsn_pwd,dsn_security)
print(dsn)
try:
    conn = ibm_db.pconnect(dsn,"","")
    print("success")
except:
    print(ibm_db.conn_errormsg())
```

App.py

```

@app.route("/", methods=['GET', 'POST'])
def register():
    if request.method == 'POST':
        sql_stmt = "insert into USERTBL values(?,?,?)"
        stmt = ibm_db.prepare(conn, sql_stmt)
        username = request.form['username']
        email = request.form['email']
        password = request.form['password']
        ibm_db.bind_param(stmt, 1, username)
        ibm_db.bind_param(stmt, 2, email)
        ibm_db.bind_param(stmt, 3, password)
        try:
            ibm_db.execute(stmt)
            return redirect('/')
        except:
            print(ibm_db.stmt_errormsg())

    return render_template('index.html')

@app.route("/login", methods=('GET', 'POST'))
def loginpage():
    if request.method == 'POST':
        username = request.form['username']
        password = request.form['password']
        query = "select COUNT(*) from usertbl where username='"+username+"' and
password='"+password+"'"
        stmt5 = ibm_db.exec_immediate(conn, query)
        row = ibm_db.fetch_tuple(stmt5)
        if(row[0] ==1 ):
            return redirect("/shop/")
    return render_template("index.html")

```

Index.html

```

<div class="wrapper">
    <div class="title-text">
        <div class="title login">
            Login Form
        </div>
        <div class="title signup">
            Signup Form
        </div>
    </div>
    <div class="form-container">
        <div class="slide-controls">
            <input type="radio" name="slide" id="login" checked>
            <input type="radio" name="slide" id="signup">
            <label for="login" class="slide login">Login</label>
            <label for="signup" class="slide signup">Signup</label>
            <div class="slider-tab"></div>
        </div>
    </div>

```

```

        <div class="form-inner">
            <form action="/login" method="post" class="login">
                <div class="field">
                    <input type="text" name="username" placeholder="Username"
required>
                </div>
                <div class="field">
                    <input type="password" name="password" placeholder="Password"
required>
                </div>
                <div class="pass-link">
                    <a href="#">Forgot password?</a>
                </div>
                <div class="field btn">
                    <div class="btn-layer"></div>
                    <input type="submit" value="Login">
                </div>
                <div class="signup-link">
                    Not a member? <a href="">Signup now</a>
                </div>
            </form>

            <form action="/" class="signup" method="post">
                <div class="field">
                    <input type="text" name="username" placeholder="Username"
required>
                </div>
                <div class="field">
                    <input type="text" name="email" placeholder="Email" required>
                </div>
                <div class="field">
                    <input type="password" name="password" placeholder="Password"
required>
                </div>
                <div class="field btn">
                    <div class="btn-layer"></div>
                    <input type="submit" value="Signup">
                </div>
            </form>

        </div>
    </div>
</div>
<script>
    const loginText = document.querySelector(".title-text .login");
    const loginForm = document.querySelector("form.login");

    const loginBtn = document.querySelector("label.login");
    const signupBtn = document.querySelector("label.signup");

```

```

const signupLink = document.querySelector("form .signup-link a");
signupBtn.onclick = (()=>{
  loginForm.style.marginLeft = "-50%";
  loginText.style.marginLeft = "-50%";
});
loginBtn.onclick = (()=>{
  loginForm.style.marginLeft = "0%";
  loginText.style.marginLeft = "0%";
});
signupLink.onclick = (()=>{
  signupBtn.click();
  return false;
});
</script>

```

Style.css

```

@import
url('https://fonts.googleapis.com/css?family=Poppins:400,500,600,700&display=swap');
*{
  margin: 0;
  padding: 0;
  box-sizing: border-box;
  font-family: 'Poppins', sans-serif;
}
html,body{
  display: grid;
  height: 100%;
  width: 100%;
  place-items: center;
  background: -webkit-linear-gradient(left, #a445b2, #fa4299);
}
::selection{
  background: #fa4299;
  color: #fff;
}
.wrapper{
  overflow: hidden;
  max-width: 390px;
  background: #fff;
  padding: 30px;
  border-radius: 5px;
  box-shadow: 0px 15px 20px rgba(0,0,0,0.1);
}
.wrapper .title-text{
  display: flex;
  width: 200%;
}
.wrapper .title{
  width: 50%;
  font-size: 35px;

```


```
font-weight: 600;
text-align: center;
transition: all 0.6s cubic-bezier(0.68,-0.55,0.265,1.55);
}
.wrapper .slide-controls{
  position: relative;
  display: flex;
  height: 50px;
  width: 100%;
  overflow: hidden;
  margin: 30px 0 10px 0;
  justify-content: space-between;
  border: 1px solid lightgrey;
  border-radius: 5px;
}
.slide-controls .slide{
  height: 100%;
  width: 100%;
  color: #fff;
  font-size: 18px;
  font-weight: 500;
  text-align: center;
  line-height: 48px;
  cursor: pointer;
  z-index: 1;
  transition: all 0.6s ease;
}
.slide-controls label.signup{
  color: #000;
}
.slide-controls .slider-tab{
  position: absolute;
  height: 100%;
  width: 50%;
  left: 0;
  z-index: 0;
  border-radius: 5px;
  background: -webkit-linear-gradient(left, #a445b2, #fa4299);
  transition: all 0.6s cubic-bezier(0.68,-0.55,0.265,1.55);
}
input[type="radio"]{
  display: none;
}
#signup:checked ~ .slider-tab{
  left: 50%;
}
#signup:checked ~ label.signup{
  color: #fff;
  cursor: default;
  user-select: none;
}
#signup:checked ~ label.login{
```



```
    color: #000;
}
#login:checked ~ label.signup{
    color: #000;
}
#login:checked ~ label.login{
    cursor: default;
    user-select: none;
}
.wrapper .form-container{
    width: 100%;
    overflow: hidden;
}
.form-container .form-inner{
    display: flex;
    width: 200%;
}
.form-container .form-inner form{
    width: 50%;
    transition: all 0.6s cubic-bezier(0.68,-0.55,0.265,1.55);
}
.form-inner form .field{
    height: 50px;
    width: 100%;
    margin-top: 20px;
}
.form-inner form .field input{
    height: 100%;
    width: 100%;
    outline: none;
    padding-left: 15px;
    border-radius: 5px;
    border: 1px solid lightgrey;
    border-bottom-width: 2px;
    font-size: 17px;
    transition: all 0.3s ease;
}
.form-inner form .field input:focus{
    border-color: #fc83bb;
    /* box-shadow: inset 0 0 3px #fb6aae; */
}
.form-inner form .field input::placeholder{
    color: #999;
    transition: all 0.3s ease;
}
form .field input:focus::placeholder{
    color: #b3b3b3;
}
.form-inner form .pass-link{
    margin-top: 5px;
}
.form-inner form .signup-link{
```

```
    text-align: center;
    margin-top: 30px;
}
.form-inner form .pass-link a,
.form-inner form .signup-link a{
    color: #fa4299;
    text-decoration: none;
}
.form-inner form .pass-link a:hover,
.form-inner form .signup-link a:hover{
    text-decoration: underline;
}
form .btn{
    height: 50px;
    width: 100%;
    border-radius: 5px;
    position: relative;
    overflow: hidden;
}
form .btn .btn-layer{
    height: 100%;
    width: 300%;
    position: absolute;
    left: -100%;
    background: -webkit-linear-gradient(right, #a445b2, #fa4299, #a445b2, #fa4299);
    border-radius: 5px;
    transition: all 0.4s ease;;
}
form .btn:hover .btn-layer{
    left: 0;
}
form .btn input[type="submit"]{
    height: 100%;
    width: 100%;
    z-index: 1;
    position: relative;
    background: none;
    border: none;
    color: #fff;
    padding-left: 0;
    border-radius: 5px;
    font-size: 20px;
    font-weight: 500;
    cursor: pointer;
}
```

Output page:



A centered white card titled "Signup Form" on a purple-to-pink gradient background. The card contains a "Login" button and a highlighted "Signup" button. Below these are input fields for "Name" (containing "Madhave Susheel"), "Email" (containing "madhave.susheel@gmail.com"), and "Password" (masked with dots). A final "Signup" button is at the bottom.

Signup Form

Login Signup

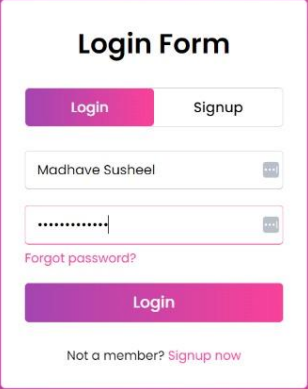
Madhave Susheel

madhave.susheel@gmail.com

.....

Signup

Login Page:



A centered white card titled "Login Form" on a purple-to-pink gradient background. The card contains a highlighted "Login" button and a "Signup" button. Below these are input fields for "Name" (containing "Madhave Susheel") and "Password" (masked with dots). A "Forgot password?" link is below the password field. A "Login" button is at the bottom, followed by a link "Not a member? Signup now".

Login Form

Login Signup

Madhave Susheel

.....

[Forgot password?](#)


Login

Not a member? [Signup now](#)

Welcome Page:


Sarah's Botique

Search for products




Red Dress
3333

[View Detail](#) [Add To Cart](#)



Blue Dress
4444

[View Detail](#) [Add To Cart](#)



Black Dress
5555

[View Detail](#) [Add To Cart](#)

Data Base Table

IBM Db2 on Cloud

Load DataLoad HistoryTablesViewsIndexesAliasesMQTsSequencesApplication objects

MVG77822.USERTBL

Back

Export to CSV

| USERNAME | EMAIL | PASSWORD |
|----------|-----------------|----------|
| Admin | admin@gmail.com | Admin |
| kiki | kiki@gmail.com | kiki@1 |
| mad | mad@gmail.com | maddy |