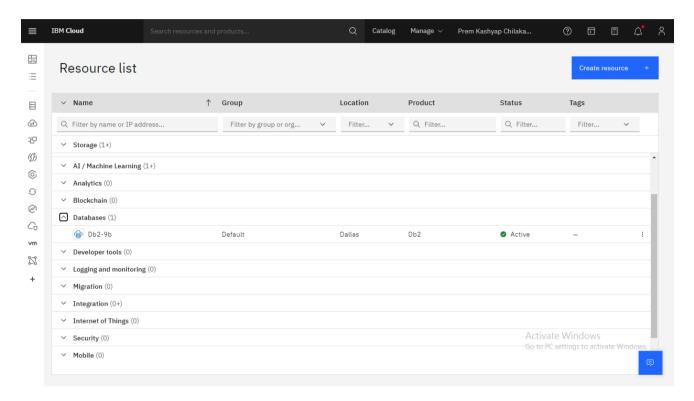
### IMPLEMENTING WEB APPLICATION

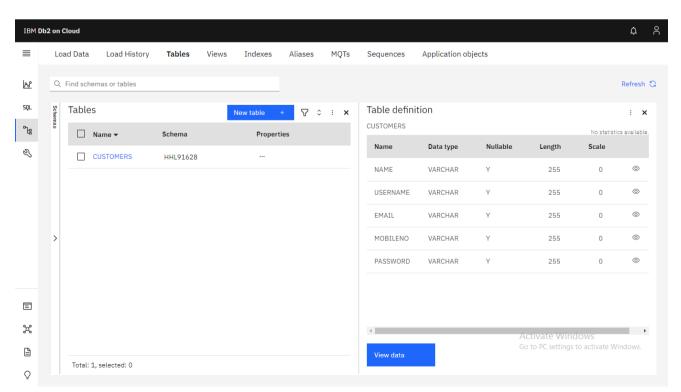
## Create UI to interact with application

Date	15 November 2022
Team ID	PNT2022TMID03865
Project Name	Personal Expense Tracker Application

### **DATABASE:**



## **TABLE AND ITS DETAILS:**



#### **TABLE DATA:**

IBM <b>D</b> I	o2 on Cloud									ф	2
■	Load Data	Load History	Tables	Views	Indexes	Aliases	MQTs	Sequences	Application objects	;	
<b>№</b> SQL	HHL91628.CUSTOMERS									Back	
В										Export to CSV	Ł
Ø	NAME USERNAME  Jhon Jhon  Johnff Johnff  Johnff Johnff  Johnff Johnff				EN	EMAIL			MOBILENO	PASSWORD	Â
					Jh	Jhon@gmail.com			9857648672	Jhon@32424	П
					he	llo			095026	cgtj	П
					he	hello@sgaasdf hello@sga			095026	sdafsf	П
					he				095026	sdfa	П
	Johnfffasd		Johnfffasd		he	hello@sgaasdfsdgf			095026	sdfg	П
≡	Prem Kashya	р	Prem			iamprem@gmail.com			8789806965	Prem@3456	
×	Prem Kashya	р	Prem		nn	nani3008@gn	nail.com		0985764867	1234	
<u></u>	Wick		Wick		wi	wick2@gmail.com			0985764867	Activate Windows Go to WC settings to activate Windows.	
Ç	ghfsd		ghfsd		df	gs			dsgfd	dsgf	•
-											

# app.py:

```
CODE TO CONNECT DATABASE:
     import ibm_db
     from flask import Flask, flash, redirect, render template, request, url for
     app = Flask(__name__)
     app.debug=True
     conn = ibm db.connect("DATABASE=bludb;HOSTNAME=98538591-7217-4024-b027-
     8baa776ffad1.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud;PORT=30875;SECURITY=SSL;SSLSe
     rverCertificate=DigiCertGlobalRootCA.crt;UID=hhl91628;PWD=vM4lAZjxo4LsBPoJ","","")
      @app.route("/")
     def home():
       return render_template("home.html")
      @app.route("/signup", methods=('GET', 'POST'))
     def signup():
       if request.method == 'POST':
         name = request.form['fname']
         username =name.split(" ")[0]
         email = request.form['femail']
         mobile = request.form['mobile']
         password = request.form['password']
         sql = "SELECT * FROM customers WHERE username =? and email=?"
         stmt = ibm db.prepare(conn, sql)
         ibm_db.bind_param(stmt,1,username)
         ibm_db.bind_param(stmt,2,email)
         ibm_db.execute(stmt)
          account = ibm_db.fetch_assoc(stmt)
```

```
if account:
      return render template('registered.html')
    else:
      insert sql = "INSERT INTO customers VALUES (?,?,?,?,?)"
      prep stmt = ibm db.prepare(conn, insert sql)
      ibm_db.bind_param(prep_stmt, 1, name)
      ibm db.bind param(prep stmt, 2, username)
      ibm_db.bind_param(prep_stmt, 3, email)
      ibm db.bind param(prep stmt, 4, mobile)
      ibm_db.bind_param(prep_stmt, 5, password)
      ibm_db.execute(prep_stmt)
      return render template('success.html')
 return render_template('signup.html')
@app.route("/about")
def about():
  return render template('about.html')
@app.route("/signin", methods=('GET', 'POST'))
def signin():
 if request.method == 'POST':
    femail = request.form['femail']
    fpassword = request.form['password']
    sql = "SELECT * FROM customers WHERE email =? and password=?"
    stmt = ibm db.prepare(conn, sql)
    ibm_db.bind_param(stmt,1,femail)
    ibm db.bind param(stmt,2,fpassword)
    ibm db.execute(stmt)
    account = ibm_db.fetch_assoc(stmt)
    if account:
      return render template('loginsuccess.html')
    else:
      return render_template('loginfailure.html')
  return render template('signin.html')
if __name__ == '__main__':
 app.run()
```