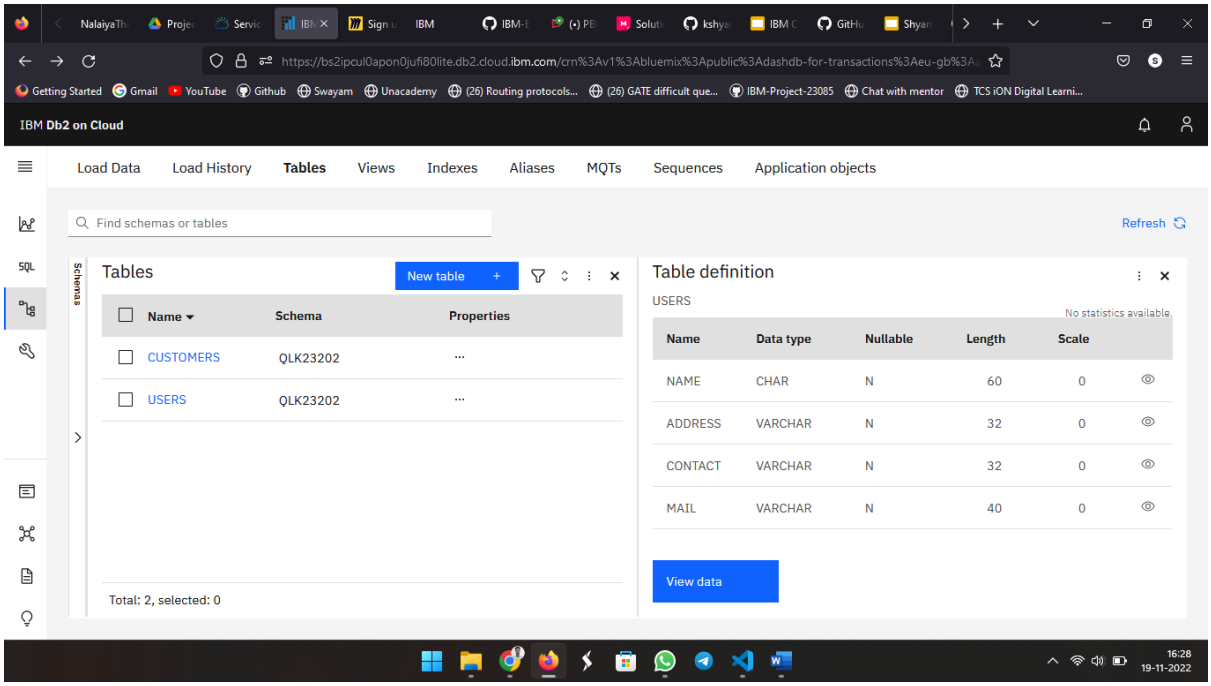
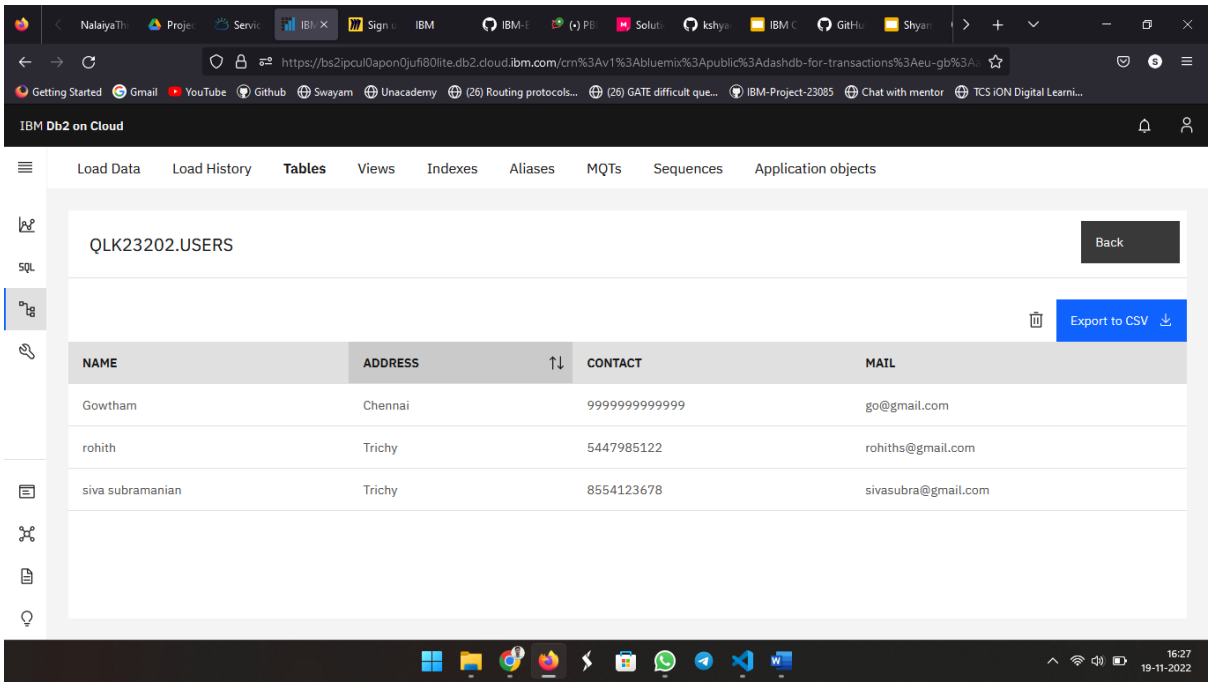
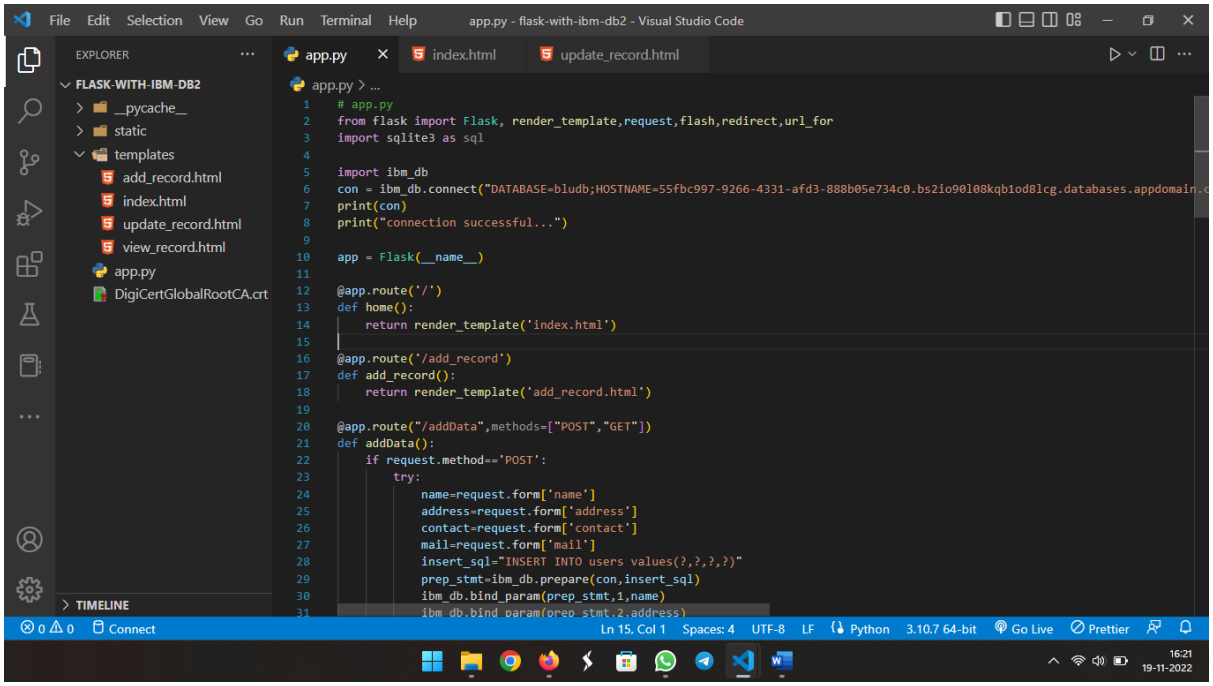
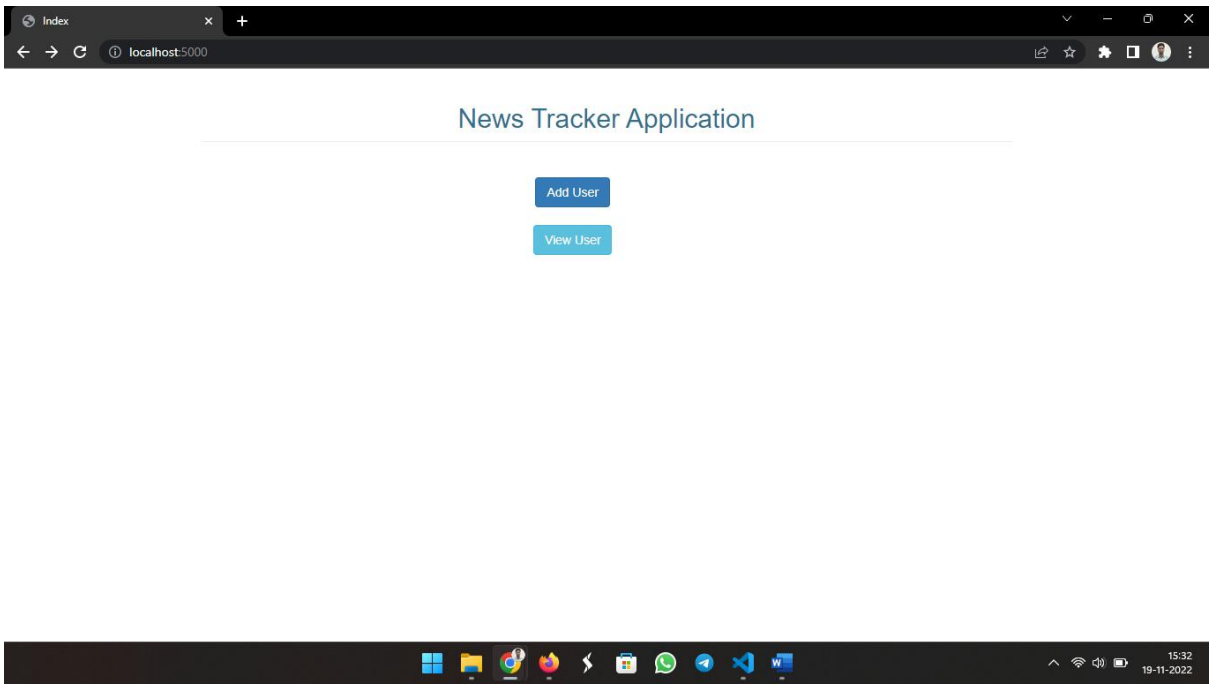
Creation	Create IBM DB2 and connect with Application
Team ID:	PNT2022TMID06749

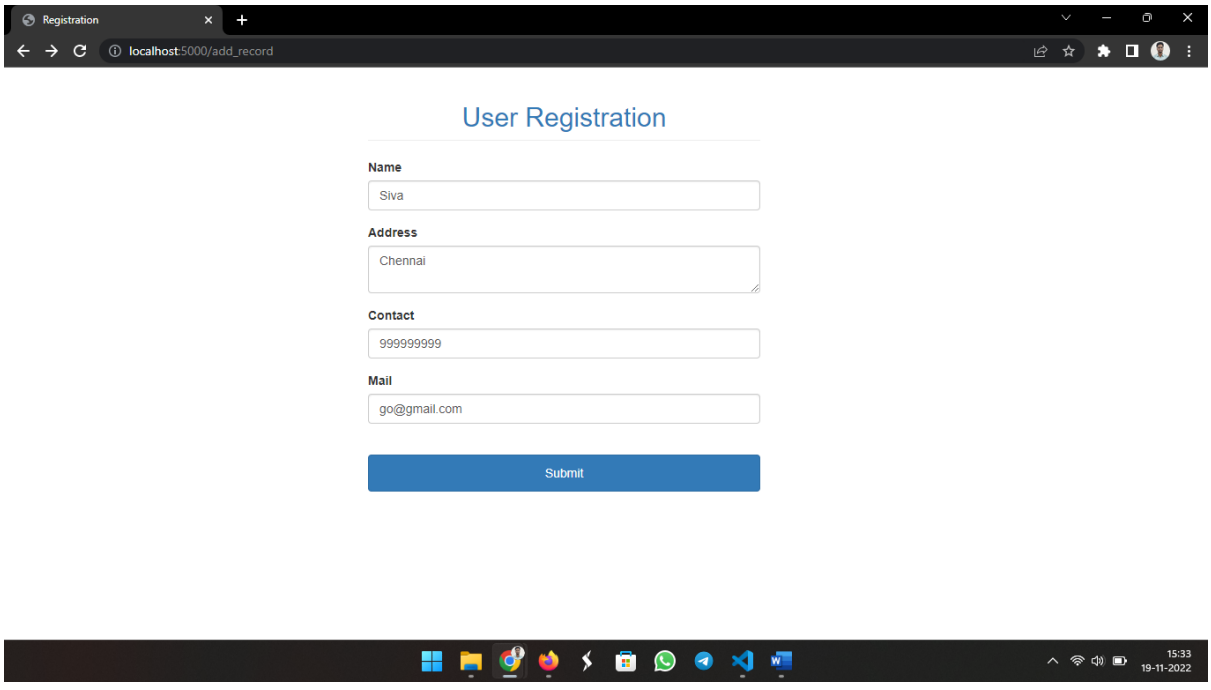
Create IBM DB2 and connect with Application

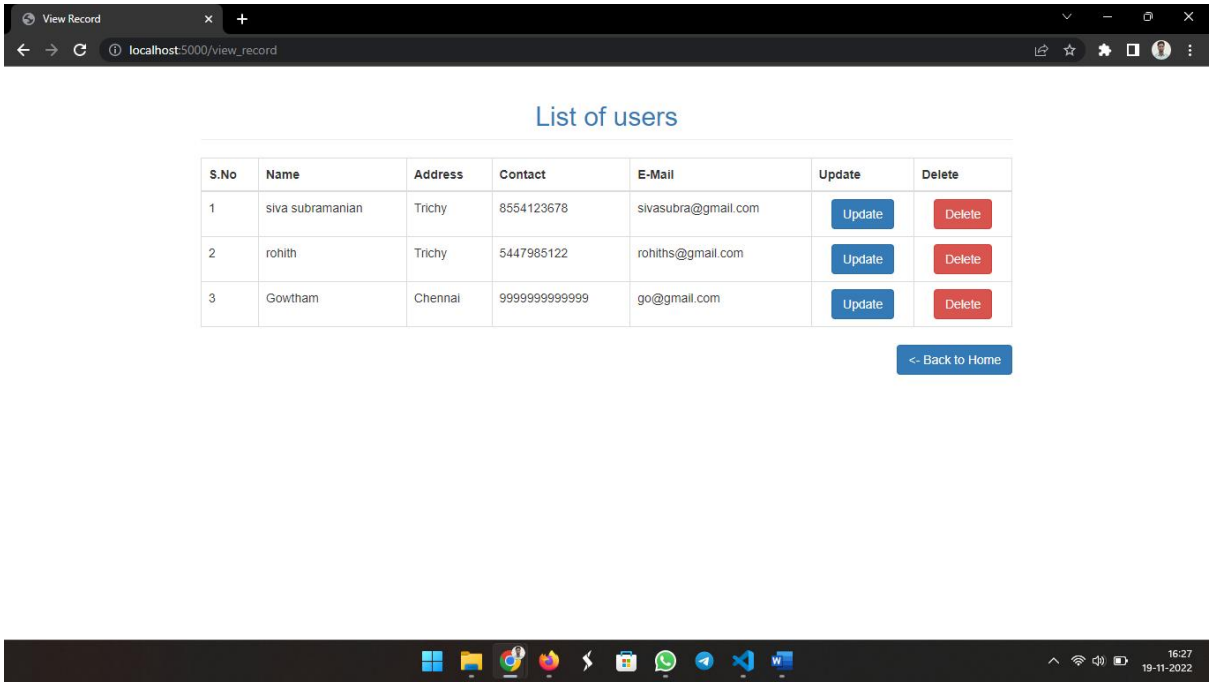





File Edit Selection View Go Run Terminal Help
app.py - flask-with-ibm-db2 - Visual Studio Code
EXPLORER
FLASK-WITH-IBM-DB2
__pycache__
static
templates
add_record.html
index.html
update_record.html
view_record.html
app.py
DigiCertGlobalRootCA.crt
TIMELINE
app.py
index.html
update_record.html
app.py > ...
# app.py
from flask import Flask, render_template,request,flash,redirect,url_for
import sqlite3 as sql
import ibm_db
con = ibm_db.connect("DATABASE=bludb;HOSTNAME=55fbc997-9266-4331-afd3-888b05e734c0.bs2io90l08kqb1od8lcg.databases.appdomain.c
print(con)
print("connection successful...")
app = Flask(__name__)
@app.route('/')
def home():
return render_template('index.html')
@app.route('/add_record')
def add_record():
return render_template('add_record.html')
@app.route("/addData",methods=["POST","GET"])
def addData():
if request.method=='POST':
try:
name=request.form['name']
address=request.form['address']
contact=request.form['contact']
mail=request.form['mail']
insert_sql="INSERT INTO users values(?,?,?,?)"
prep_stmt=ibm_db.prepare(con,insert_sql)
ibm_db.bind_param(prep_stmt,1,name)
ibm_db.bind_param(prep_stmt,2,address)
0 0 Connect
Ln 15, Col 1 Spaces: 4 UTF-8 LF Python 3.10.7 64-bit Go Live Prettier
16:21
19-11-2022


Index
localhost:5000
News Tracker Application
Add User
View User
15:32
19-11-2022


Registration
localhost:5000/add_record
User Registration
Name
Siva
Address
Chennai
Contact
999999999
Mail
go@gmail.com
Submit
15:33
19-11-2022



Code <app.py>

```
# app.py
from flask import Flask,
render_template,request,flash,redirect,url_for
import sqlite3 as sql

import ibm_db
con = ibm_db.connect("DATABASE=bludb;HOSTNAME=55fbc997-9266-4331-
afd3-
888b05e734c0.bs2io90l08kqb1od8lcg.databases.appdomain.cloud;PORT=319
29;SECURITY=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=ql
k23202;PWD=VRhFlQo0AHQQwX6f",'','')
print(con)
print("connection successful...")

app = Flask(__name__)

@app.route('/')
def home():
    return render_template('index.html')

@app.route('/add_record')
def add_record():
    return render_template('add_record.html')
```

```python
@app.route("/addData",methods=["POST","GET"])
def addData():
    if request.method=='POST':
        try:
            name=request.form['name']
            address=request.form['address']
            contact=request.form['contact']
            mail=request.form['mail']
            insert_sql="INSERT INTO users values(?,?,?,?)"
            prep_stmt=ibm_db.prepare(con,insert_sql)
            ibm_db.bind_param(prep_stmt,1,name)
            ibm_db.bind_param(prep_stmt,2,address)
            ibm_db.bind_param(prep_stmt,3,contact)
            ibm_db.bind_param(prep_stmt,4,mail)
            ibm_db.execute(prep_stmt)
            print(name+address+contact+mail)

            flash("Record Added Successfully","success")
        except:
            flash("Error in Insert Operation","danger")
        finally:
            return redirect(url_for("home"))

@app.route('/view_record')
def view_record():
    users=[]
    sql="select * from users"
    stmt=ibm_db.exec_immediate(con,sql)
    dictionary=ibm_db.fetch_both(stmt)
    while dictionary != False:
        users.append(dictionary)
        dictionary=ibm_db.fetch_both(stmt)
    return render_template("view_record.html",data=users)

@app.route('/update_record/<string:id>',methods=["POST","GET"])
def update_record(id):
    insert_sql="SELECT * FROM users where contact=?"
    prep_stmt=ibm_db.prepare(con,insert_sql)
    ibm_db.bind_param(prep_stmt,1,id)
    data=ibm_db.fetch_row(prep_stmt)

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

```
        try:
            name=request.form['name']
            address=request.form['address']
            contact=request.form['contact']
            mail=request.form['mail']

            insert_sql="UPDATE users SET name=?,address=?,mail=?
where contact=?"
            prep_stmt=ibm_db.prepare(con,insert_sql)
            ibm_db.bind_param(prep_stmt,1,name)
            ibm_db.bind_param(prep_stmt,2,address)
            ibm_db.bind_param(prep_stmt,3,mail)
            ibm_db.bind_param(prep_stmt,4,contact)
            ibm_db.execute(prep_stmt)

            flash("Update Successfully","success")
        except:
            flash("Error in Update Operation","danger")
        finally:
            return redirect(url_for("home"))
    print(data)
    return render_template('update_record.html',data=data)

@app.route('/delete_record/<string:id>')
def delete_record(id):
    try:
        insert_sql="DELETE FROM users where contact=?"
        prep_stmt=ibm_db.prepare(con,insert_sql)
        ibm_db.bind_param(prep_stmt,1,id)
        ibm_db.execute(prep_stmt)

        flash("Record Deleted Successfully","success")
    except:
        flash("Record Delete Failed","danger")
    finally:
        return redirect(url_for("home"))


if __name__ == '__main__':
    app.run(debug=True)
```