**Name : Swetha S**

**Reg.no : 2031050**

**Backend**

**You need to create a REST service that can fetch bank details, using the data given in the API’s query parameters. You can use the data available in this** [**Link**](https://imageuploadtestingbob.s3.us-east-2.amazonaws.com/dev/backend-intern/bank_branches.csv) **in your backend DB**

**Query tab in postgresql:**

create table bank1(

ifsc varchar(750),bank\_id integer,

branch varchar(700),

address varchar(750),

city varchar(700),

district varchar(700),

states varchar(750),

bank\_name varchar(700),

primary key(ifsc)

);

COPY bank1(

ifsc,

bank\_id,

branch,

address,

city,

district,

states,

bank\_name

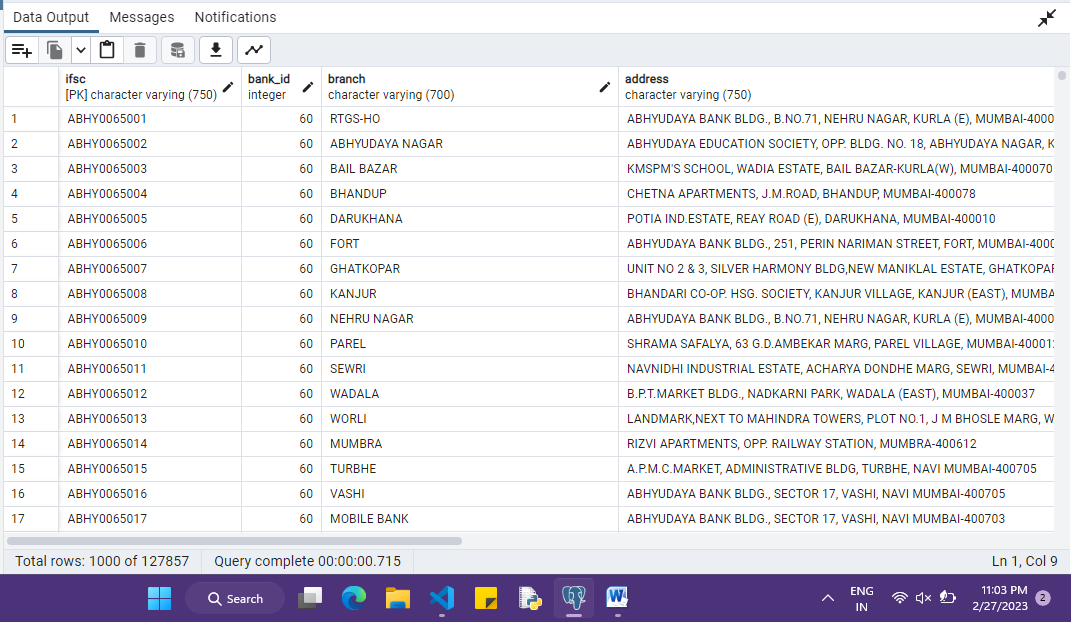
) from 'C:\Users\acer\Downloads\bank\_branches.csv'

Delimiter ','

CSV HEADER;

select \* from bank1;

**Output:**



**server.js**

const client = require('./connectivity.js')

const express = require('express');

const app = express();

var port\_number = process.env.PORT || 3003;

app.listen(port\_number,()=>{

    console.log(port\_number)

    console.log("Server running...")

});

client.connect();

app.get('/api/branch', (req, res)=>{

    client.query(`SELECT \* FROM bank1 WHERE branch LIKE '${req.query.q}' ORDER BY ifsc LIMIT ${req.query.limit} OFFSET ${req.query.offset};`, (err, result)=>{

        if(!err){

            res.send({"branches":result.rows});

        }

    });

    client.end;

})

app.get('/api/search', (req, res) => {

    client.query(`SELECT \* FROM bank1 WHERE branch like '%${req.query.q}%' or address like '%${req.query.q}%' or city like '%${req.query.q}%' or district like '%${req.query.q}%' or states like '%${req.query.q}%' or bank\_name like '%${req.query.q}%' order by ifsc  limit ${req.query.limit} offset ${req.query.offset}`, (err, result)=>{

        if(!err){

            res.send(result.rows);

        }

    });

    client.end;

})

**connectivity.js**

const {Client} = require('pg');

const client = new Client({

    user:"postgres",

    password:"root123",

 connectionString: "postgres://postgres:root123@localhost:5432/postgres",

 //ssl: true

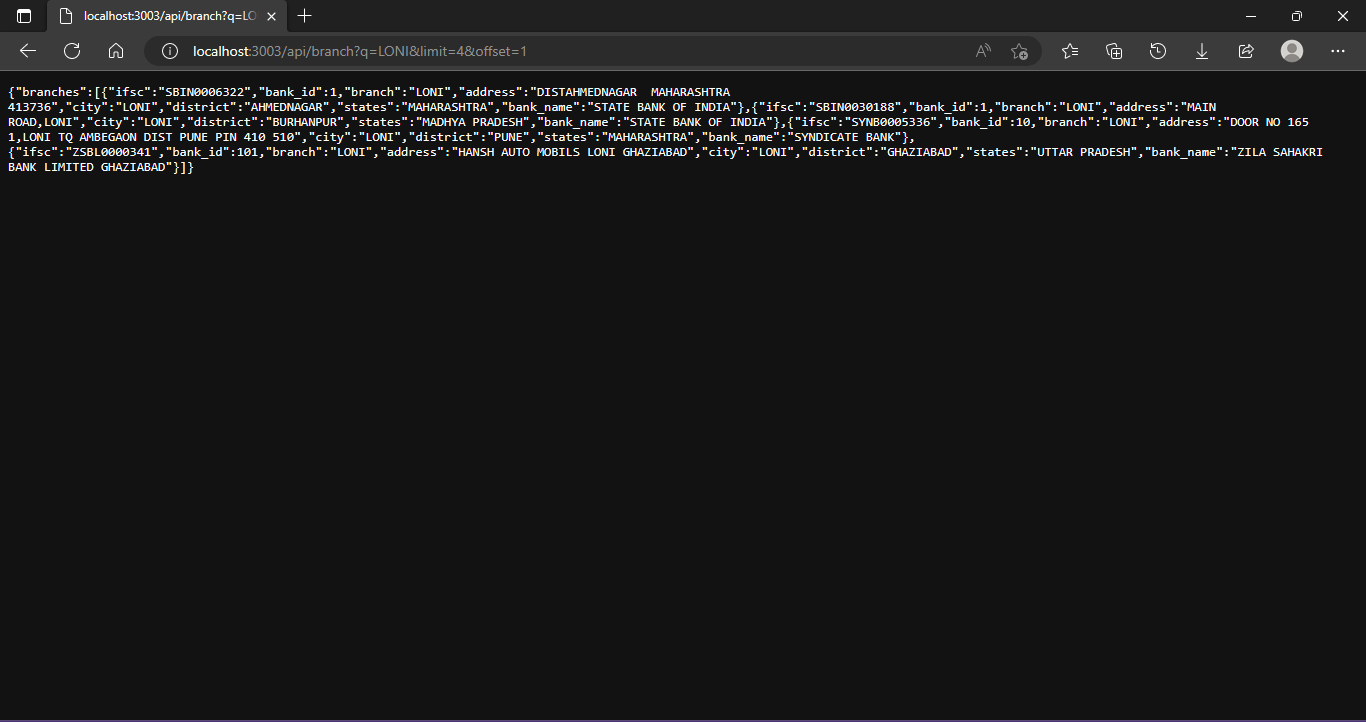
});

module.exports = client

**Case 1**

Search API to return possible matches across all columns and all rows, ordered by IFSC code (ascending order) with limit and offset.

Request URL - /api/search?q=Mumbai&limit=2&offset=1



**Case 2**

Branch API to return possible matches based on the branch name ordered by IFSC code (descending order) with limit and offset

Request URL - /api/branch?q=LONI&limit=1&offset=1

