1st set up data base
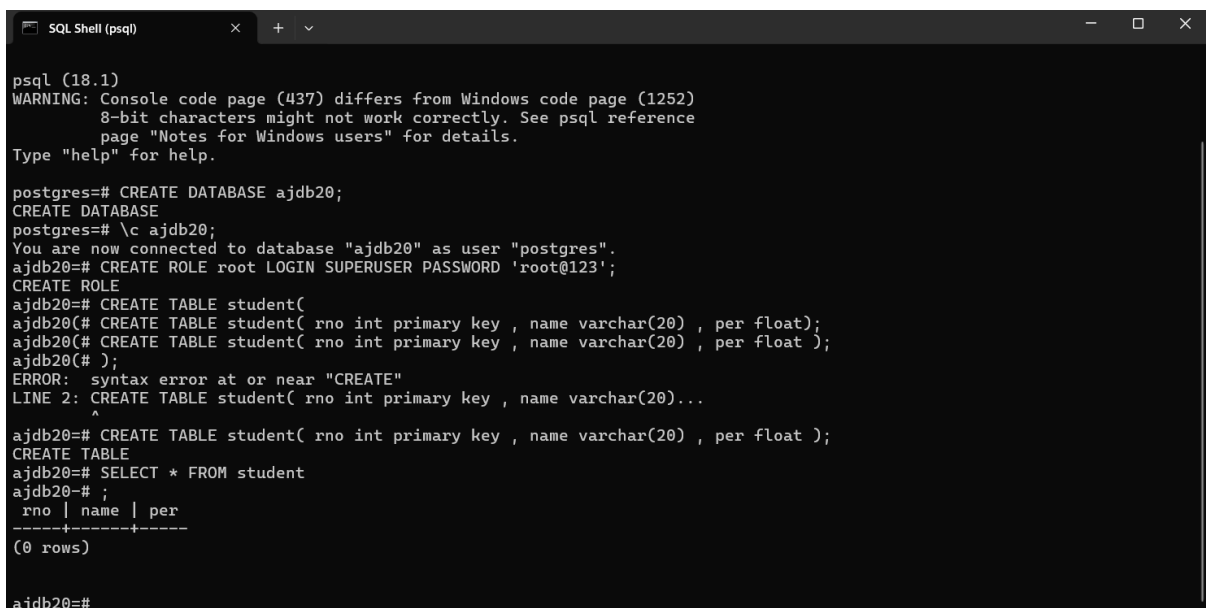
postgresql

    a) CREATE DATABASE ajdb20;
    b) Use database → \c ajdb20;
    c) CREATE ROLE root LOGIN SUPERUSER PASSWORD 'root@123';
            root → username (can be anything)
            root@123 → password (can be anything)
    d) CREATE TABLE student(rno int primary key , name varchar(20) , per float);

```
SQL Shell (psql)                                                    —  □  ×

psql (18.1)
WARNING: Console code page (437) differs from Windows code page (1252)
         8-bit characters might not work correctly. See psql reference
         page "Notes for Windows users" for details.
Type "help" for help.

postgres=# CREATE DATABASE ajdb20;
CREATE DATABASE
postgres=# \c ajdb20;
You are now connected to database "ajdb20" as user "postgres".
ajdb20=# CREATE ROLE root LOGIN SUPERUSER PASSWORD 'root@123';
CREATE ROLE
ajdb20=# CREATE TABLE student(
ajdb20(# CREATE TABLE student( rno int primary key , name varchar(20) , per float);
ajdb20(# CREATE TABLE student( rno int primary key , name varchar(20) , per float );
ajdb20(# );
ERROR:  syntax error at or near "CREATE"
LINE 2: CREATE TABLE student( rno int primary key , name varchar(20)...
        ^
ajdb20=# CREATE TABLE student( rno int primary key , name varchar(20) , per float );
CREATE TABLE
ajdb20=# SELECT * FROM student
ajdb20-# ;
 rno | name | per
-----+------+-----
(0 rows)


ajdb20=#
```

Steps to write jdbc application

1)Load the driver
2)form connection
3)prepare sql
4)fire/execute query
5)collect the result from sql → optional
6)close the connection

Code:

```java
package com.tca;
import java.io.*;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
public class AddStudent {
        public static void main(String[] args) {

                Connection con = null;  // To form connection
                PreparedStatement ps = null; // To fire query

                /*
                 * By creating connection we make a bridge between java application and database
server
                 * and by preparedstatement we can fire queries
                 */

                BufferedReader br = new BufferedReader(new InputStreamReader(System.in));

                try {

                        //Step 1 --> Load Driver

                        Class.forName("org.postgresql.Driver");  //---> ("Name of driver class")

                        //Step 2 --> Forming Connection

                        con = DriverManager.getConnection( "jdbc:postgresql://localhost/ajdb20" ,
"root" , "root@123");

                        /*
                         * 1)when we go in company instead of local host , 198.09.09.12:8729 ,
198.09.09.12-->IP address , 8729-->port no.
                         * 2)root --> username set at time of create role
                         * 3)root@123 ---> password set at time of create role
                         *
                         * ajdb20-->database on which we are working
                         * postgresql --> database server we are using
                         *
                         * DriverManager is a  class and getConnection is a static method of that
class
                         *
                         */

                        System.out.print("Enter the Roll Number :");
                        int rno = Integer.parseInt(br.readLine());

                        System.out.print("Enter Name :");
                        String name = br.readLine(); //no need of converstion
```

```java
                    System.out.print("Enter Percentage :");
                    double per = Double.parseDouble(br.readLine());

                    //Step3 --> Prepare Sql

                    ps = con.prepareStatement("Insert into student values(?,?,?)");  // 3 ?
because our table has 3 columns

                    ps.setInt(1 , rno);  // 1st ? is for rno
                    ps.setString(2 , name);
                    ps.setDouble(3 , per);

                    //Step4 --> Fire Sql

                    int sval = ps.executeUpdate(); //The sql which we have made in 3rd step will
get fired i.e. a record will be created

                    if(sval == 0) {
                            System.out.println("Failed to add record !!!");
                    }
                    else {
                            System.out.println("Record is inserted succesfully for RollNumber :"
+ rno);
                    }

                    //Step5 --> close connection
                    con.close();

            }

            catch(Exception e) {

                    System.out.println("*** Problem in JDBC Application ***");
                    e.printStackTrace();

            }
        }
}
```

Output :

```
CREATE DATABASE
postgres=# \c ajdb20;
You are now connected to database "ajdb20" as user "postgres".
ajdb20=# CREATE ROLE root LOGIN SUPERUSER PASSWORD 'root@123';
CREATE ROLE
ajdb20=# CREATE TABLE student(
ajdb20(# CREATE TABLE student( rno int primary key , name varchar(20) , per float);
ajdb20(# CREATE TABLE student( rno int primary key , name varchar(20) , per float );
ajdb20(# );
ERROR:  syntax error at or near "CREATE"
LINE 2: CREATE TABLE student( rno int primary key , name varchar(20)...
        ^
ajdb20=# CREATE TABLE student( rno int primary key , name varchar(20) , per float );
CREATE TABLE
ajdb20=# SELECT * FROM student
ajdb20-# ;
 rno | name | per
-----+------+-----
(0 rows)


ajdb20=# select * from student
ajdb20-# ;
 rno | name |  per
-----+------+--------
 101 | aaa  | 72.35
(1 row)
```