PART B

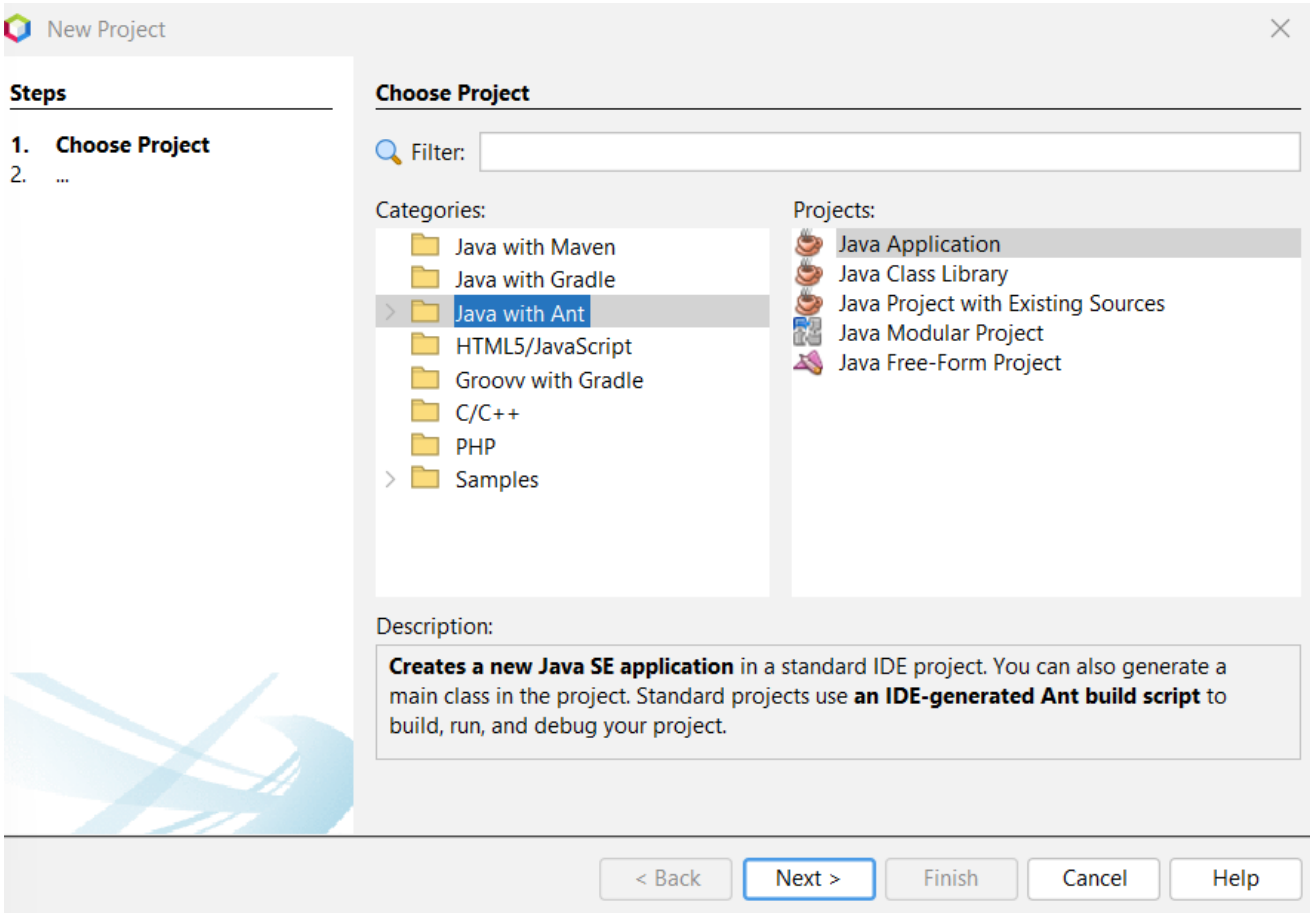
1)Write a menu driven JDBC program to perform basic operations with Student Table.

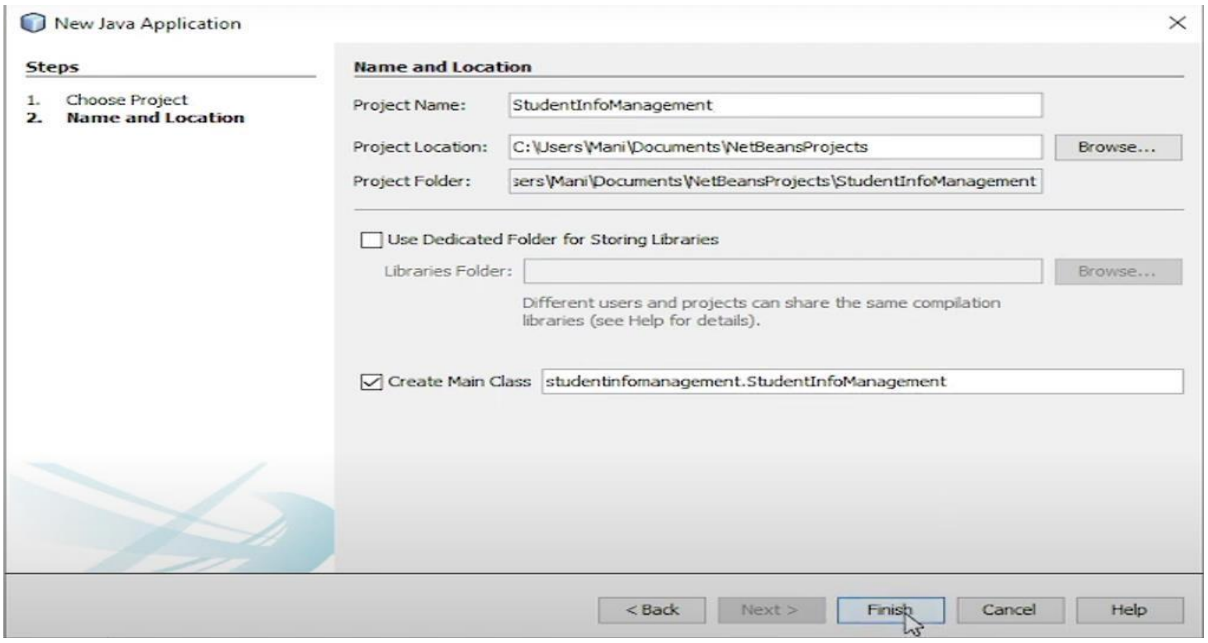
MENU

1. **Add new Student**
2. **Delete a specified students Record**
3. **Update Students Address specified students Record**
4. **Search for a particular Student**
5. **Exit**

Student

StRegNo	StName	Stdob	StAddress	StClass	StCourse





StudentInfoManagement.java

```
package studentinfomanagement;


import java.sql.Connection;
import java.util.Scanner;
import java.sql.Date;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.logging.Level;
import java.util.logging.Logger;

public class StudentInfoManagement {
public static void main(String[] args) {
      try {
         Scanner in=new Scanner(System.in);
         int choice=5;
         int regno;
```

```
        String sname,sadd,sclass,scourse,dob,sql;
        Class.forName("org.apache.derby.jdbc.ClientDriver");
        Connection
con=DriverManager.getConnection("jdbc:derby://localhost:1527/Student","ma
ni","abcd1234");
        Statement stmt=con.createStatement();
        ResultSet rs;
do{
           System.out.println("MENU");
           System.out.println(" ----------------");
           System.out.println("1.Add student");
           System.out.println("2.Delete student");
           System.out.println("3.Update student");
           System.out.println("4.Search student");
           System.out.println("5.Exit");
           System.out.println(" ----------------");
           System.out.print("Enter your choice-> ");
           choice=in.nextInt();
           switch(choice){
              case 1:
                 System.out.println("------Enter student details -------");
                 System.out.print("Reg.No: ");
                 regno=in.nextInt();
                 in.nextLine();
                 System.out.print("Name: ");
                 sname=in.nextLine();
                 System.out.print("DOB [yyyy-mm-dd]: ");
                 dob=in.nextLine();

                 System.out.print("Address: ");
                 sadd=in.nextLine();

                 System.out.print("Class: ");
                 sclass=in.nextLine();
                 System.out.print("Course: ");
                 scourse=in.nextLine();
```

```
                    sql="INSERT INTO STDTABLE(STREGNO, STNAME, STDOB,
STADDRESS, STCLASS, STCOURSE)VALUES ("+regno+", '"+sname+"', '"+dob+"',
'"+sadd+"', '"+sclass+"', '"+scourse+"')";
                  int result= stmt.executeUpdate(sql);
                  if(result==1){
                     System.out.println("Student details are saved");
                  }else{
                     System.out.println("Error!while saving student details");
                  }
                   break;

case 2:
                   System.out.print("Enter student Reg.No: ");
                   regno=in.nextInt();
                   sql="SELECT COUNT(*) FROM STDTABLE WHERE
STREGNO="+regno;
                   rs=stmt.executeQuery(sql);
                   rs.next();
                   if(rs.getInt(1)==1){
                     sql="DELETE FROM STDTABLE WHERE STREGNO="+regno;
                     int res=stmt.executeUpdate(sql);
                     if(res==1){
                        System.out.println("Record  deleted");
                     }else{
                        System.out.println("Eroor! while deleting student record");
                     }


                   }else{
                      System.out.println("Student record not found");
                   }
                   break;

case 3:
                   System.out.print("Enter RegNo: ");
                   regno=in.nextInt();
                   in.nextLine();
```

```
                sql="SELECT COUNT(*) FROM STDTABLE WHERE
STREGNO="+regno;
                rs=stmt.executeQuery(sql);
                rs.next();
                if(rs.getInt(1)==1){
                   sql="SELECT STADDRESS FROM STDTABLE WHERE
STREGNO="+regno;
                   rs=stmt.executeQuery(sql);
                   rs.next();
                   System.out.println("Old address is:"+rs.getString(1));
                   System.out.print("Enter new address: ");
                   String add=in.nextLine();
                   sql="UPDATE STDTABLE SET STADDRESS='"+add+"' WHERE
STREGNO="+regno;
                   if(stmt.executeUpdate(sql)==1){
                      System.out.println("Address updated");
                   }else{
                      System.out.println("Error! While updating address");
                   }

                }else{
                  System.out.println("Student record not found");
                }
                break;

case 4:
                System.out.print("Enter Reg No: ");
                regno=in.nextInt();
                sql="SELECT * FROM STDTABLE WHERE STREGNO="+regno;
                rs=stmt.executeQuery(sql);
                if(rs!=null){
                   rs.next();
                   System.out.println("Student details are");
                   System.out.println(" --------------------");
                   System.out.println("Reg No: "+rs.getInt(1));
                   System.out.println("Name: "+rs.getString(2));
                   System.out.println("DOB: "+rs.getString(3));
```

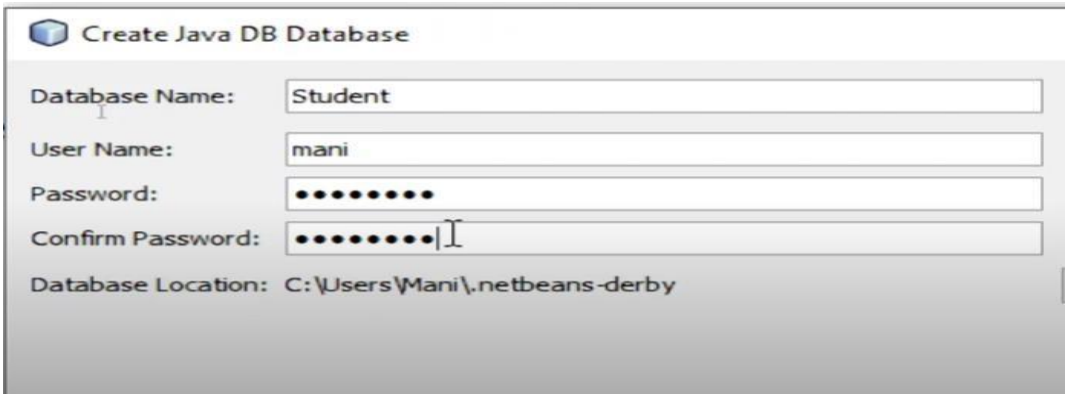
```
                    System.out.println("Address: "+rs.getString(4));
                    System.out.println("Class: "+rs.getString(5));
                    System.out.println("Course: "+rs.getString(6));
                    System.out.println(" --------------------");

                }
                break;

case 5:
                stmt.close();
                con.close();
                System.out.println("Thank you");
                return;
              default:
                System.out.println("Wrong Choice\n Try Again!");
            }

        }while(true);
      } catch (ClassNotFoundException ex) {

Logger.getLogger(StudentInfoManagement.class.getName()).log(Level.SEVERE,
null, ex);
      } catch (SQLException ex) {

Logger.getLogger(StudentInfoManagement.class.getName()).log(Level.SEVERE,
null, ex);
      }

   }

}
```

****

To create database and table

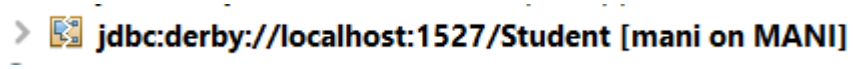
Gotowindows-> services->Databases->JavaDB->right click on JavaDB->select Start Server.

Next right click on JavaDB->select Create Database

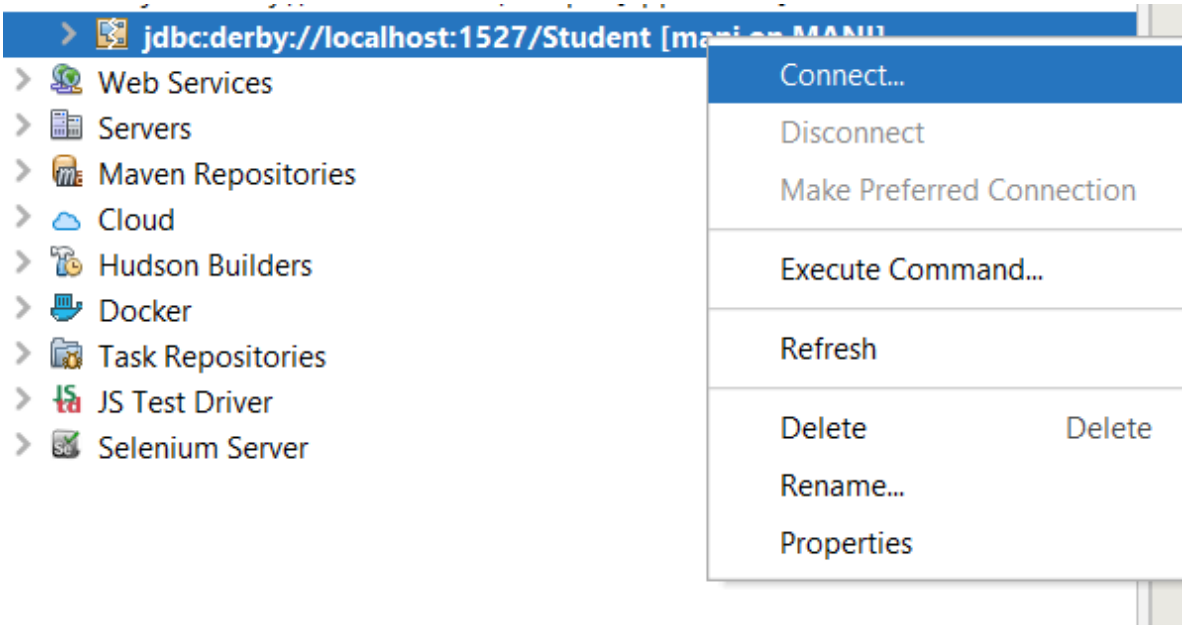


Password: abcd1234

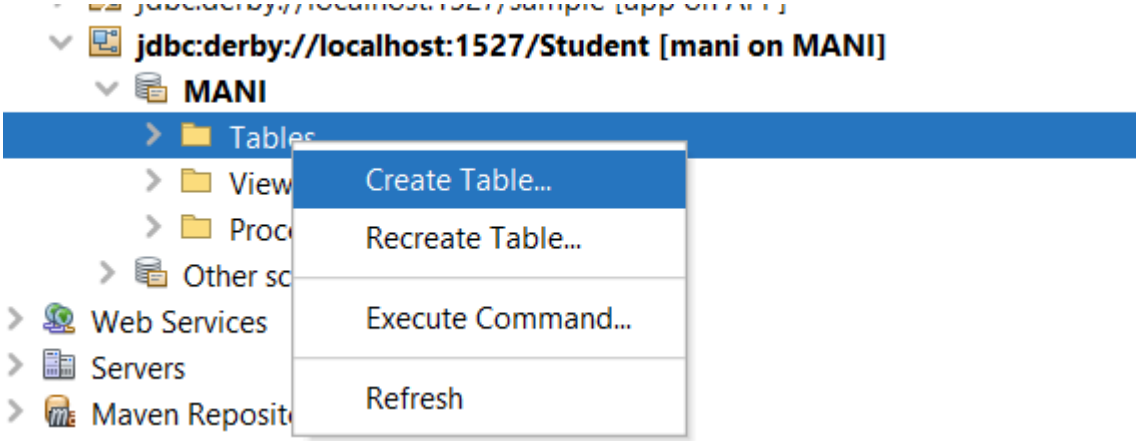
Database is created.

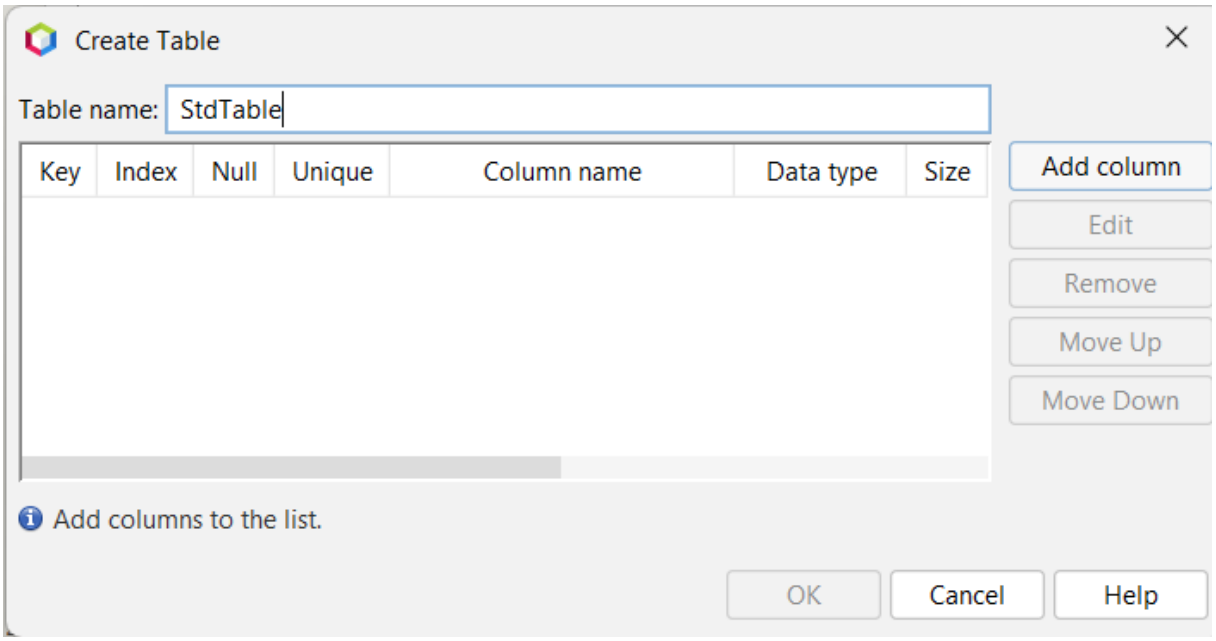


Next right click

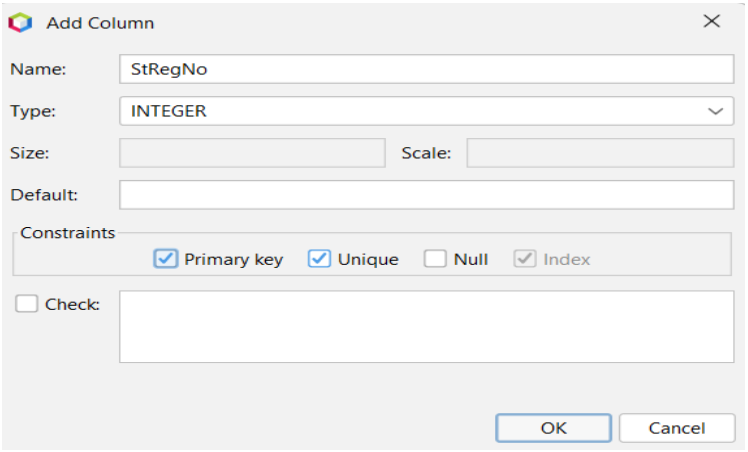


To create Table: Right click on Tables

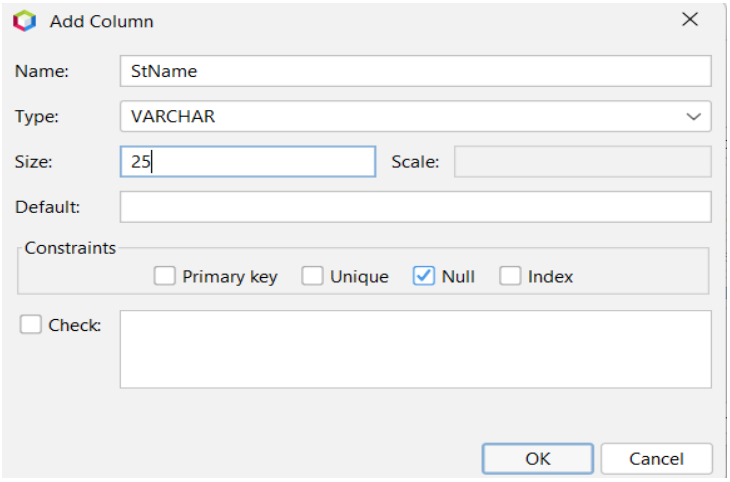


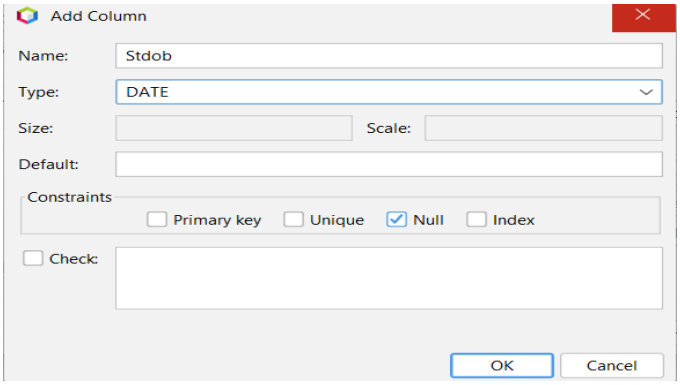


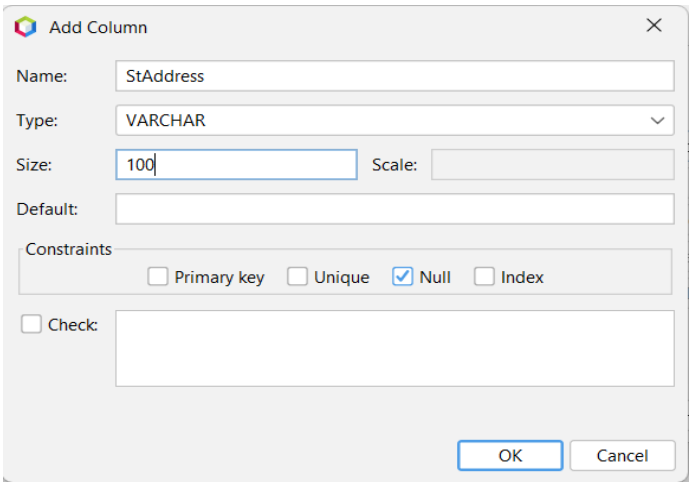
Give table name and click Add column.

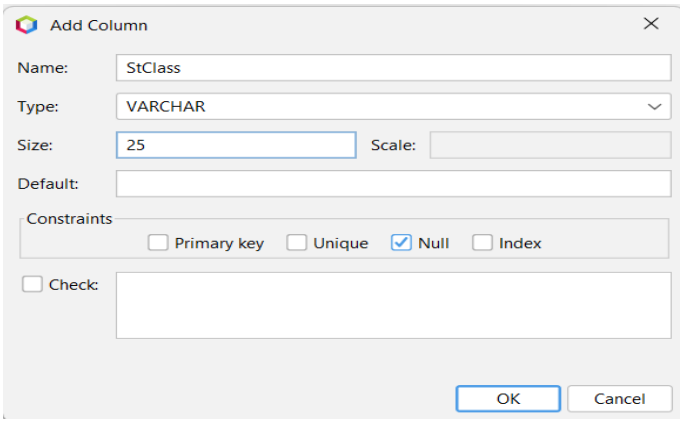


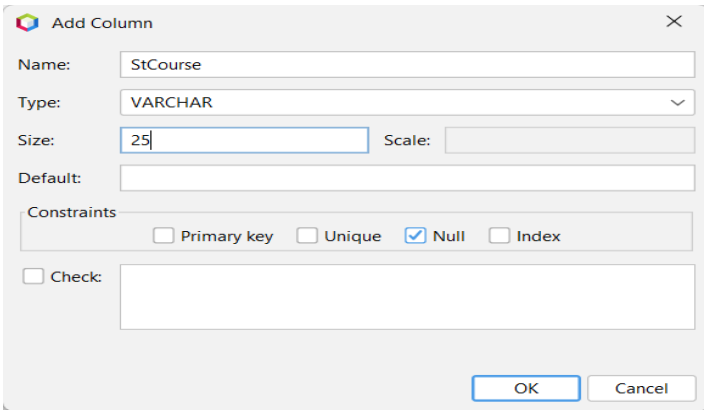
Again click Add column.



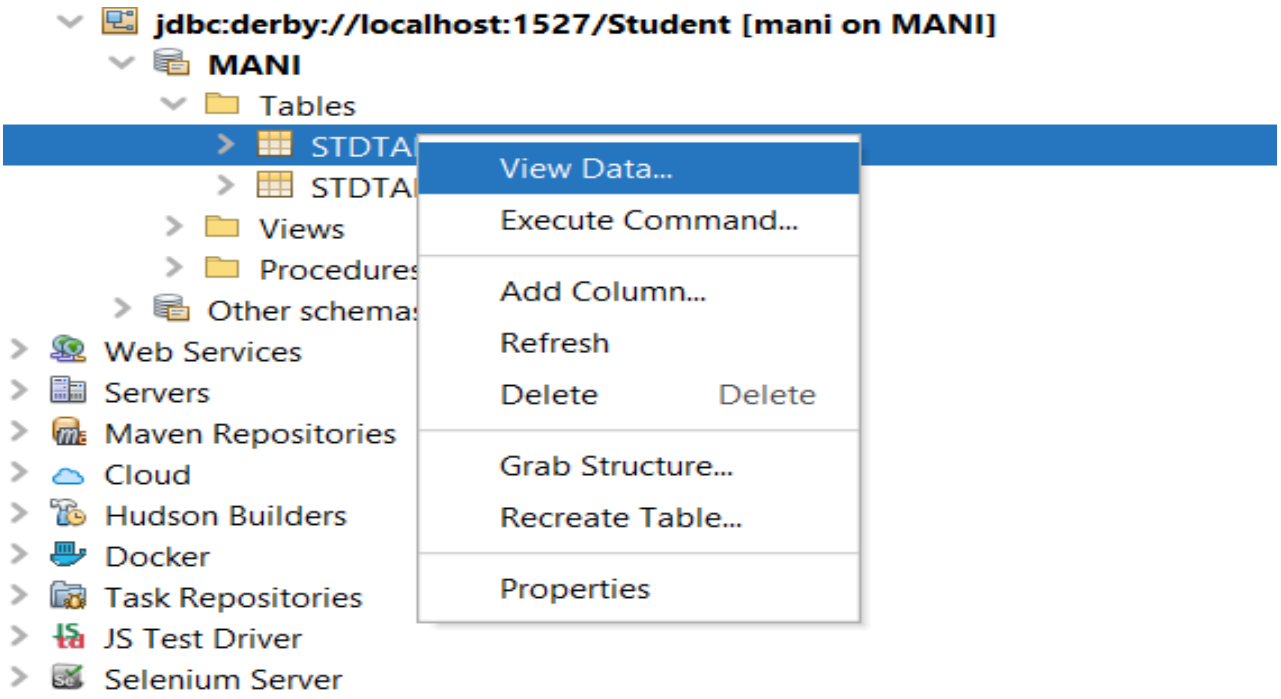



Add Column
Name: StAddress
Type: VARCHAR
Size: 100
Scale:
Default:
Constraints
Primary key
Unique
Null
Index
Check:
OK
Cancel


Add Column
Name: StClass
Type: VARCHAR
Size: 25
Scale:
Default:
Constraints
Primary key
Unique
Null
Index
Check:
OK
Cancel

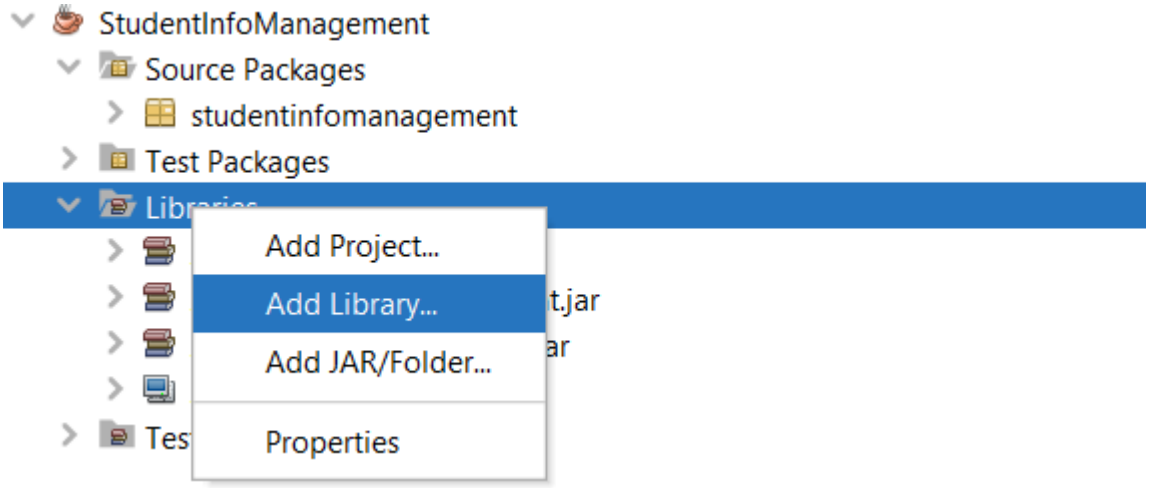

Add Column
Name: StCourse
Type: VARCHAR
Size: 25
Scale:
Default:
Constraints
Primary key
Unique
Null
Index
Check:
OK
Cancel

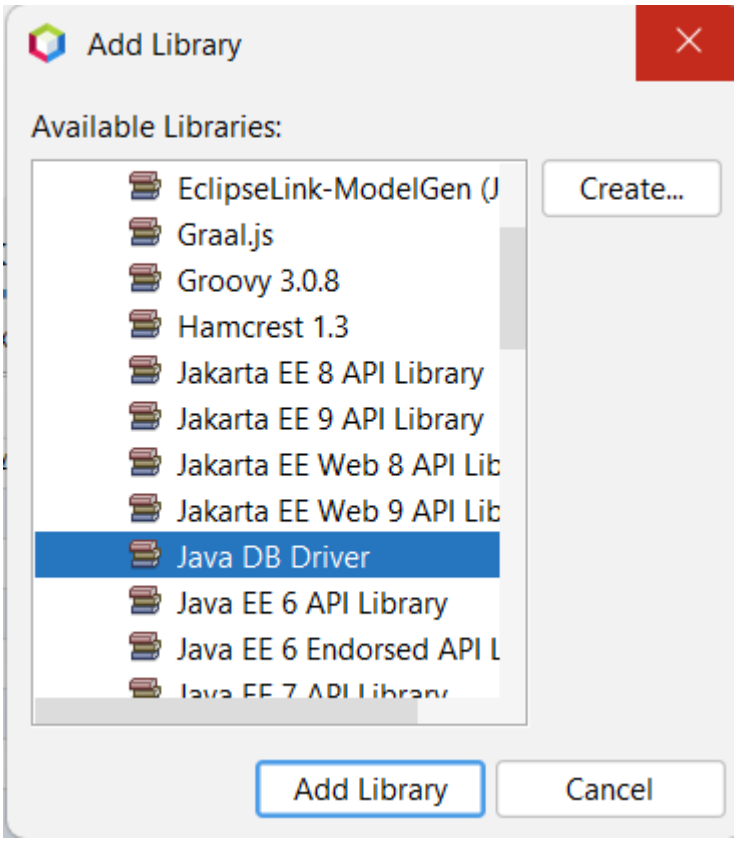
To view the table data-> right click on STDTABLE

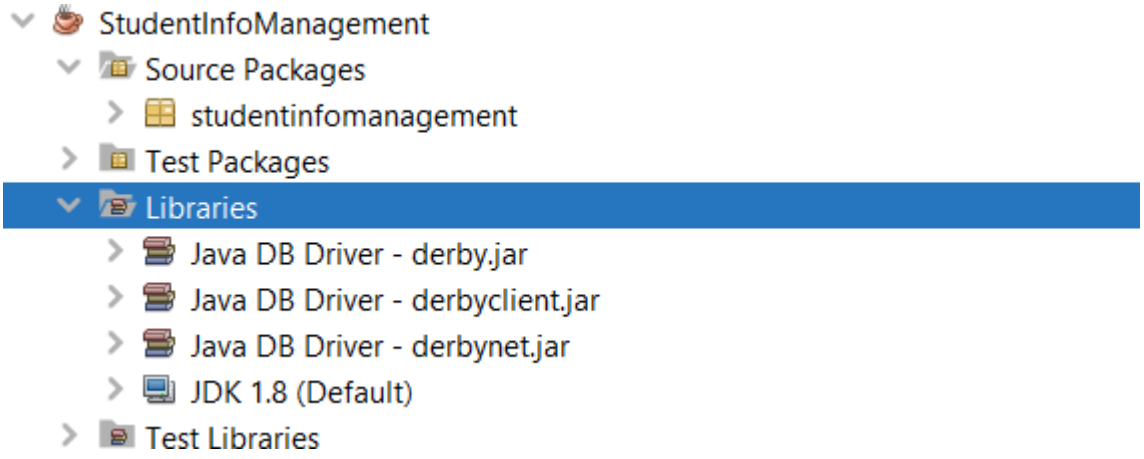


****

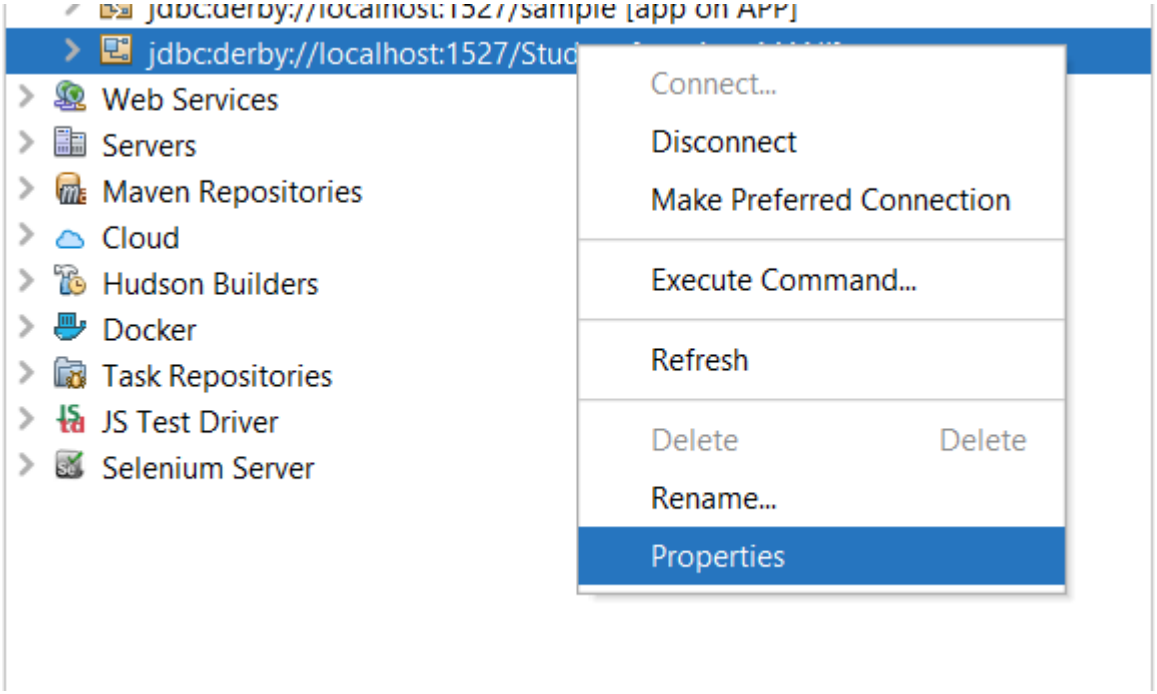
To add Library file

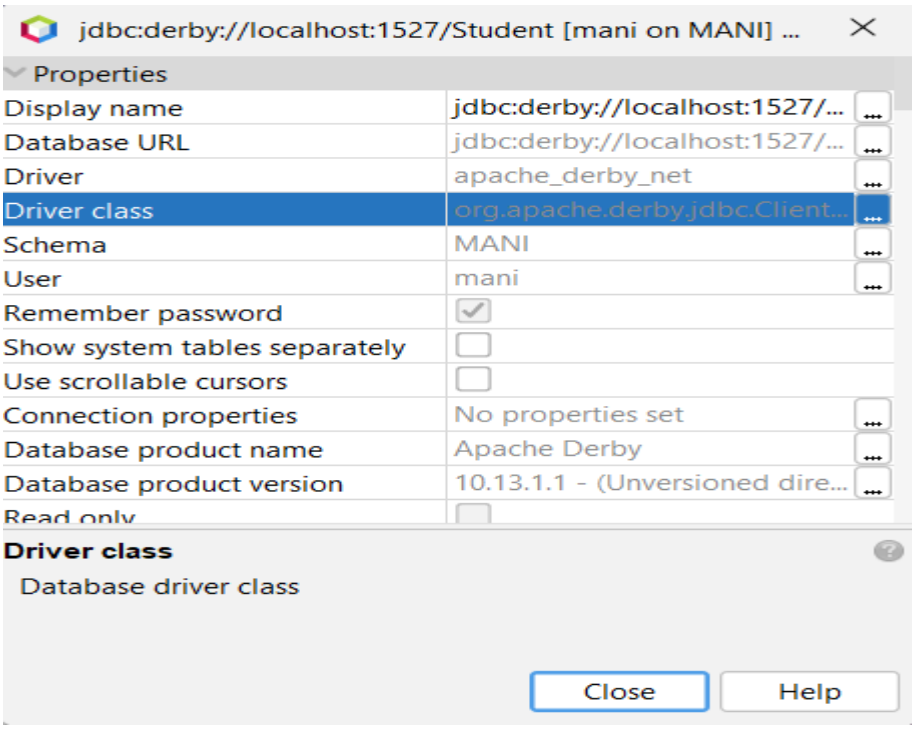


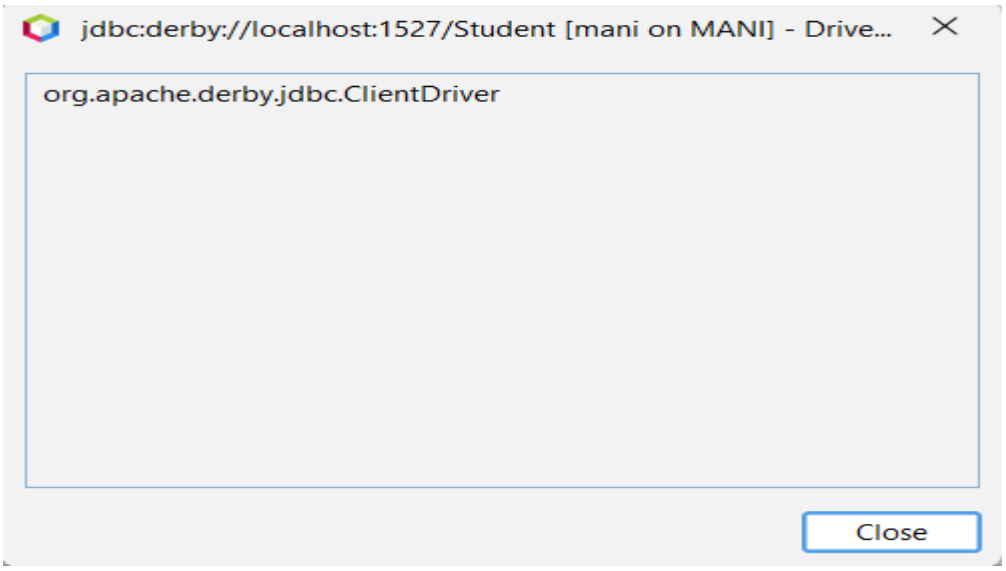

Add Library
Available Libraries:
EclipseLink-ModelGen (J
Graal.js
Groovy 3.0.8
Hamcrest 1.3
Jakarta EE 8 API Library
Jakarta EE 9 API Library
Jakarta EE Web 8 API Lib
Jakarta EE Web 9 API Lib
Java DB Driver
Java EE 6 API Library
Java EE 6 Endorsed API L
Create...
Add Library
Cancel


StudentInfoManagement
Source Packages
studentinfomanagement
Test Packages
Libraries
Java DB Driver - derby.jar
Java DB Driver - derbyclient.jar
Java DB Driver - derbynet.jar
JDK 1.8 (Default)
Test Libraries

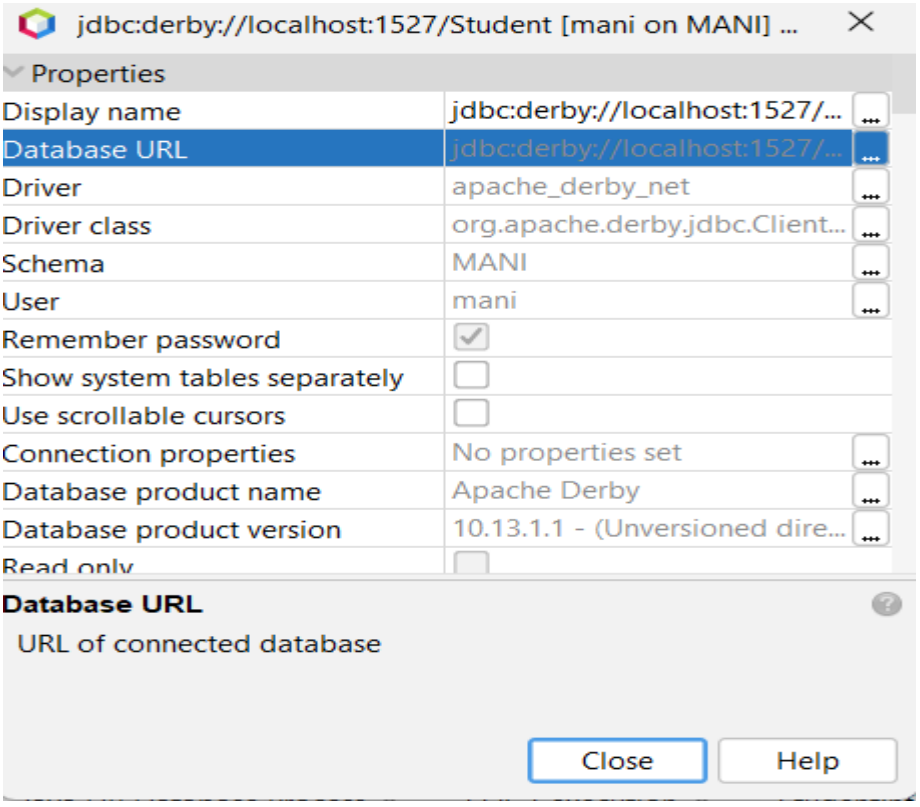
Class.forName("org.apache.derby.jdbc.ClientDriver");

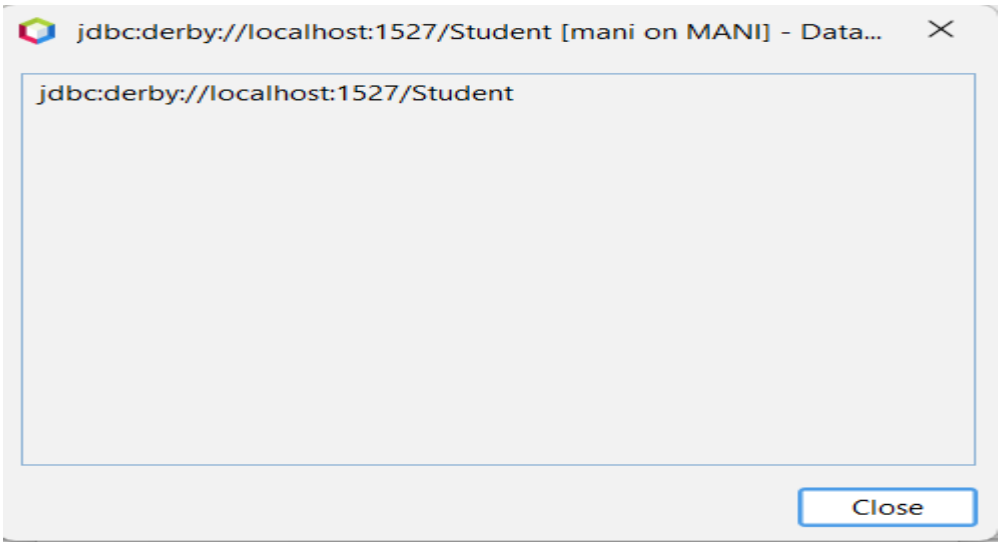






Connection con=DriverManager.getConnection("jdbc:derby://localhost:1527/Student





Output:

MENU

1.Add student
2.Delete student
3.Update student
4.Search student
5.Exit

Enter your choice-> 1
------Enter student details-------
Reg.No: 101
Name: Abc
DOB [yyyy-mm-dd]: 2005-06-04
Address: Kundapura
Class: III BCA
Course: BCA
Student details are saved
MENU

1.Add student
2.Delete student
3.Update student
4.Search student
5.Exit

Enter your choice-> 3
Enter RegNo: 101
Old address is:Kundapura
Enter new address: Udupi
Address updated
MENU

1.Add student
2.Delete student
3.Update student
4.Search student

5.Exit

Enter your choice-> 4
Enter Reg No: 101
Student details are

Reg No: 101
Name: Abc
DOB: 2005-06-04
Address: Udupi
Class: III BCA
Course: BCA

MENU

1.Add student
2.Delete student
3.Update student
4.Search student
5.Exit

Enter your choice-> 2
Enter student Reg.No: 101
Record deleted
MENU

1.Add student
2.Delete student
3.Update student
4.Search student
5.Exit

Enter your choice-> 5
Thank you