**Exp3**

Navigation: Back-button navigation, Hierarchical navigation patterns, Ancestral navigation (Up button), Descendant navigation, Lateral navigation with tabs and swipes

Task: Design a complete **Student Management Application using Android** and provide effective navigation between various Activities

**Source code for student Management Application Using Android.**

package com.delaroystudios.studentmgt;

import java.security.PublicKey;

import android.os.Bundle;

import android.app.Activity;

import android.app.AlertDialog.Builder;

import android.content.Context;

import android.database.Cursor;

import android.database.sqlite.SQLiteDatabase;

import android.view.Menu;

import android.view.View;

import android.view.View.OnClickListener;

import android.widget.Button;

import android.widget.EditText;

import com.delaroystudios.studentmgt.R;

public class StudentMainActivity extends Activity {

EditText ename,eroll\_no,emarks;

Button add,view,viewall,Show1,delete,modify;

SQLiteDatabase db;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_student\_main);

ename=(EditText)findViewById(R.id.name);

eroll\_no=(EditText)findViewById(R.id.roll\_no);

emarks=(EditText)findViewById(R.id.marks);

add=(Button)findViewById(R.id.addbtn);

view=(Button)findViewById(R.id.viewbtn);

viewall=(Button)findViewById(R.id.viewallbtn);

delete=(Button)findViewById(R.id.deletebtn);

Show1=(Button)findViewById(R.id.showbtn);

modify=(Button)findViewById(R.id.modifybtn);

db=openOrCreateDatabase("Student\_manage", Context.MODE\_PRIVATE, null);

db.execSQL("CREATE TABLE IF NOT EXISTS student(rollno INTEGER,name VARCHAR,marks INTEGER);");

add.setOnClickListener(new OnClickListener()

{

@Override

public void onClick(View v) {

// TODO Auto-generated method stub

if(eroll\_no.getText().toString().trim().length()==0||

ename.getText().toString().trim().length()==0||

emarks.getText().toString().trim().length()==0)

{

showMessage("Error", "Please enter all values");

return;

}

db.execSQL("INSERT INTO student VALUES('"+eroll\_no.getText()+"','"+ename.getText()+

"','"+emarks.getText()+"');");

showMessage("Success", "Record added successfully");

clearText();

}

});

delete.setOnClickListener(new OnClickListener() {

@Override

public void onClick(View v) {

// TODO Auto-generated method stub

if(eroll\_no.getText().toString().trim().length()==0)

{

showMessage("Error", "Please enter Rollno");

return;

}

Cursor c=db.rawQuery("SELECT \* FROM student WHERE rollno='"+eroll\_no.getText()+"'", null);

if(c.moveToFirst())

{

db.execSQL("DELETE FROM student WHERE rollno='"+eroll\_no.getText()+"'");

showMessage("Success", "Record Deleted");

}

else

{

showMessage("Error", "Invalid Rollno");

}

clearText();

}

});

modify.setOnClickListener(new OnClickListener() {

@Override

public void onClick(View v) {

// TODO Auto-generated method stub

if(eroll\_no.getText().toString().trim().length()==0)

{

showMessage("Error", "Please enter Rollno");

return;

}

Cursor c=db.rawQuery("SELECT \* FROM student WHERE rollno='"+eroll\_no.getText()+"'", null);

if(c.moveToFirst())

{

db.execSQL("UPDATE student SET name='"+ename.getText()+"',marks='"+emarks.getText()+

"' WHERE rollno='"+eroll\_no.getText()+"'");

showMessage("Success", "Record Modified");

}

else

{

showMessage("Error", "Invalid Rollno");

}

clearText();

}

});

view.setOnClickListener(new OnClickListener() {

@Override

public void onClick(View v) {

// TODO Auto-generated method stub

if(eroll\_no.getText().toString().trim().length()==0)

{

showMessage("Error", "Please enter Rollno");

return;

}

Cursor c=db.rawQuery("SELECT \* FROM student WHERE rollno='"+eroll\_no.getText()+"'", null);

if(c.moveToFirst())

{

ename.setText(c.getString(1));

emarks.setText(c.getString(2));

}

else

{

showMessage("Error", "Invalid Rollno");

clearText();

}

}

});

viewall.setOnClickListener(new OnClickListener() {

@Override

public void onClick(View v) {

// TODO Auto-generated method stub

Cursor c=db.rawQuery("SELECT \* FROM student", null);

if(c.getCount()==0)

{

showMessage("Error", "No records found");

return;

}

StringBuffer buffer=new StringBuffer();

while(c.moveToNext())

{

buffer.append("Rollno: "+c.getString(0)+"\n");

buffer.append("Name: "+c.getString(1)+"\n");

buffer.append("Marks: "+c.getString(2)+"\n\n");

}

showMessage("Student Details", buffer.toString());

}

});

Show1.setOnClickListener(new OnClickListener() {

@Override

public void onClick(View v) {

// TODO Auto-generated method stub

showMessage("Student Management Application", "Brought To You By code-projects.org");

}

});

}

public void showMessage(String title,String message)

{

Builder builder=new Builder(this);

builder.setCancelable(true);

builder.setTitle(title);

builder.setMessage(message);

builder.show();

}

public void clearText()

{

eroll\_no.setText("");

ename.setText("");

emarks.setText("");

eroll\_no.requestFocus();

}

@Override

public boolean onCreateOptionsMenu(Menu menu) {

// Inflate the menu; this adds items to the action bar if it is present.

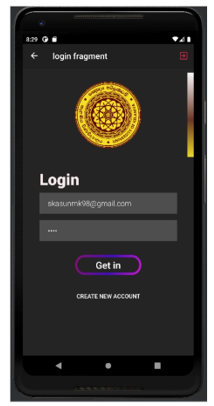
getMenuInflater().inflate(R.menu.student\_main, menu);

return true;

}

}

**OUTPUT:**





**Exp 4**

Connect to the Internet: Security best practices for network operations, Including permissions in the manifest, Performing network operations on a worker thread, Making an HTTP connection, Parsing the results, Managing the network state

**Task: Develop an Android Application that stores Student Details into the hosting server and retrieve student details from the server**

**Code for Activity\_main.xml:**

[?](https://www.codingconnect.net/android-application-makes-use-database/)

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99  100  101  102  103  104  105  106  107  108  109 | <?**xml** version="1.0" encoding="utf-8"?>  <**AbsoluteLayout** xmlns:android="http://schemas.android.com/apk/res  /android"      android:layout\_width="match\_parent"      android:layout\_height="match\_parent">      <**TextView**          android:layout\_width="wrap\_content"          android:layout\_height="wrap\_content"          android:layout\_x="50dp"          android:layout\_y="20dp"          android:text="Student Details"          android:textSize="30sp" />       <**TextView**          android:layout\_width="wrap\_content"          android:layout\_height="wrap\_content"          android:layout\_x="20dp"          android:layout\_y="110dp"          android:text="Enter Rollno:"          android:textSize="20sp" />       <**EditText**          android:id="@+id/Rollno"          android:layout\_width="150dp"          android:layout\_height="wrap\_content"          android:layout\_x="175dp"          android:layout\_y="100dp"          android:inputType="number"          android:textSize="20sp" />       <**TextView**          android:layout\_width="wrap\_content"          android:layout\_height="wrap\_content"          android:layout\_x="20dp"          android:layout\_y="160dp"          android:text="Enter Name:"          android:textSize="20sp" />       <**EditText**          android:id="@+id/Name"          android:layout\_width="150dp"          android:layout\_height="wrap\_content"          android:layout\_x="175dp"          android:layout\_y="150dp"          android:inputType="text"          android:textSize="20sp" />       <**TextView**          android:layout\_width="wrap\_content"          android:layout\_height="wrap\_content"          android:layout\_x="20dp"          android:layout\_y="210dp"          android:text="Enter Marks:"          android:textSize="20sp" />       <**EditText**          android:id="@+id/Marks"          android:layout\_width="150dp"          android:layout\_height="wrap\_content"          android:layout\_x="175dp"          android:layout\_y="200dp"          android:inputType="number"          android:textSize="20sp" />        <**Button**          android:id="@+id/Insert"          android:layout\_width="150dp"          android:layout\_height="wrap\_content"          android:layout\_x="25dp"          android:layout\_y="300dp"          android:text="Insert"          android:textSize="30dp" />       <**Button**          android:id="@+id/Delete"          android:layout\_width="150dp"          android:layout\_height="wrap\_content"          android:layout\_x="200dp"          android:layout\_y="300dp"          android:text="Delete"          android:textSize="30dp" />       <**Button**          android:id="@+id/Update"          android:layout\_width="150dp"          android:layout\_height="wrap\_content"          android:layout\_x="25dp"          android:layout\_y="400dp"          android:text="Update"          android:textSize="30dp" />       <**Button**          android:id="@+id/View"          android:layout\_width="150dp"          android:layout\_height="wrap\_content"          android:layout\_x="200dp"          android:layout\_y="400dp"          android:text="View"          android:textSize="30dp" />        <**Button**          android:id="@+id/ViewAll"          android:layout\_width="200dp"          android:layout\_height="wrap\_content"          android:layout\_x="100dp"          android:layout\_y="500dp"          android:text="View All"          android:textSize="30dp" />   </**AbsoluteLayout**>  **Code for MainActivity.java:** |

**package** com.example.exno5;

**import** android.app.Activity;

**import** android.app.AlertDialog.Builder;

**import** android.content.Context;

**import** android.database.Cursor;

**import** android.database.sqlite.SQLiteDatabase;

**import** android.os.Bundle;

**import** android.view.View;

**import** android.view.View.OnClickListener;

**import** android.widget.Button;

**import** android.widget.EditText;

**public** **class** MainActivity **extends** Activity **implements** OnClickListener

{

    EditText Rollno,Name,Marks;

    Button Insert,Delete,Update,View,ViewAll;

    SQLiteDatabase db;

    /\*\* Called when the activity is first created. \*/

    @Override

**public** **void** onCreate(Bundle savedInstanceState)

    {

**super**.onCreate(savedInstanceState);

        setContentView(R.layout.activity\_main);

        Rollno=(EditText)findViewById(R.id.Rollno);

        Name=(EditText)findViewById(R.id.Name);

        Marks=(EditText)findViewById(R.id.Marks);

        Insert=(Button)findViewById(R.id.Insert);

        Delete=(Button)findViewById(R.id.Delete);

        Update=(Button)findViewById(R.id.Update);

        View=(Button)findViewById(R.id.View);

        ViewAll=(Button)findViewById(R.id.ViewAll);

         Insert.setOnClickListener(**this**);

        Delete.setOnClickListener(**this**);

        Update.setOnClickListener(**this**);

        View.setOnClickListener(**this**);

        ViewAll.setOnClickListener(**this**);

         // Creating database and table

        db=openOrCreateDatabase("StudentDB", Context.MODE\_PRIVATE, **null**);

        db.execSQL("CREATE TABLE IF NOT EXISTS student(rollno VARCHAR,name VARCHAR,marks VARCHAR);");

    }

**public** **void** onClick(View view)

    {

        // Inserting a record to the Student table

**if**(view==Insert)

        {

            // Checking for empty fields

**if**(Rollno.getText().toString().trim().length()==0||

                    Name.getText().toString().trim().length()==0||

                    Marks.getText().toString().trim().length()==0)

            {

                showMessage("Error", "Please enter all values");

**return**;

            }

            db.execSQL("INSERT INTO student VALUES('"+Rollno.getText()+"','"+Name.getText()+

                    "','"+Marks.getText()+"');");

            showMessage("Success", "Record added");

            clearText();

        }

        // Deleting a record from the Student table

**if**(view==Delete)

        {

            // Checking for empty roll number

**if**(Rollno.getText().toString().trim().length()==0)

            {

                showMessage("Error", "Please enter Rollno");

**return**;

            }

            Cursor c=db.rawQuery("SELECT \* FROM student WHERE rollno='"+Rollno.getText()+"'", **null**);

**if**(c.moveToFirst())

            {

                db.execSQL("DELETE FROM student WHERE rollno='"+Rollno.getText()+"'");

                showMessage("Success", "Record Deleted");

            }

**else**

            {

                showMessage("Error", "Invalid Rollno");

            }

            clearText();

        }

        // Updating a record in the Student table

**if**(view==Update)

        {

            // Checking for empty roll number

**if**(Rollno.getText().toString().trim().length()==0)

            {

                showMessage("Error", "Please enter Rollno");

**return**;

            }

            Cursor c=db.rawQuery("SELECT \* FROM student WHERE rollno='"+Rollno.getText()+"'", **null**);

**if**(c.moveToFirst()) {

                db.execSQL("UPDATE student SET name='" + Name.getText() + "',marks='" + Marks.getText() +

                        "' WHERE rollno='"+Rollno.getText()+"'");

                showMessage("Success", "Record Modified");

            }

**else** {

                showMessage("Error", "Invalid Rollno");

            }

            clearText();

        }

        // Display a record from the Student table

**if**(view==View)

        {

            // Checking for empty roll number

**if**(Rollno.getText().toString().trim().length()==0)

            {

                showMessage("Error", "Please enter Rollno");

**return**;

            }

            Cursor c=db.rawQuery("SELECT \* FROM student WHERE rollno='"+Rollno.getText()+"'", **null**);

**if**(c.moveToFirst())

            {

                Name.setText(c.getString(1));

                Marks.setText(c.getString(2));

            }

**else**

            {

                showMessage("Error", "Invalid Rollno");

                clearText();

            }

        }

        // Displaying all the records

**if**(view==ViewAll)

        {

            Cursor c=db.rawQuery("SELECT \* FROM student", **null**);

**if**(c.getCount()==0)

            {

                showMessage("Error", "No records found");

**return**;

            }

            StringBuffer buffer=**new** StringBuffer();

**while**(c.moveToNext())

            {

                buffer.append("Rollno: "+c.getString(0)+"\n");

                buffer.append("Name: "+c.getString(1)+"\n");

                buffer.append("Marks: "+c.getString(2)+"\n\n");

            }

            showMessage("Student Details", buffer.toString());

        }

    }

**public** **void** showMessage(String title,String message)

    {

        Builder builder=**new** Builder(**this**);

        builder.setCancelable(**true**);

        builder.setTitle(title);

        builder.setMessage(message);

        builder.show();

    }

**public** **void** clearText()

    {

        Rollno.setText("");

        Name.setText("");

        Marks.setText("");

        Rollno.requestFocus();

    }

}

**OUTPUT:**