

```

package com.example.myapplicationa;

import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

public class ResultActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_result);

        Button buttonReturn = findViewById(R.id.buttonReturn);
        buttonReturn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                finish(); // Return to previous activity
            }
        });

        // Get the intent that started this activity
        Bundle extras = getIntent().getExtras();
        if (extras != null) {
            double interpolatedValue = extras.getDouble("interpolatedValue");
            String interpolationSteps =
extras.getString("interpolationSteps");

            // Display result
            TextView textViewResult = findViewById(R.id.textViewResult);
            textViewResult.setText("Interpolated y value: " +
interpolatedValue);

            // Display solution
            TextView textViewSolution = findViewById(R.id.textViewSolution);
            textViewSolution.setText(interpolationSteps);
        }
    }
}

```

Result.java

```

package com.example.myapplicationa;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Spinner;
import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    EditText editTextXValues, editTextYValues, editTextXValue;
    Spinner spinnerInterpolationMethod;
    Button buttonInterpolate;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        editTextXValues = findViewById(R.id.editTextXValues);
        editTextYValues = findViewById(R.id.editTextYValues);
        editTextXValue = findViewById(R.id.editTextXValue);
        spinnerInterpolationMethod =
findViewById(R.id.spinnerInterpolationMethod);
        buttonInterpolate = findViewById(R.id.buttonInterpolate);

        ArrayAdapter<CharSequence> adapter =
ArrayAdapter.createFromResource(this,
            R.array.interpolation_methods,
            android.R.layout.simple_spinner_item);

        adapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);

        spinnerInterpolationMethod.setAdapter(adapter);

        buttonInterpolate.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                interpolate();
            }
        });

        private void interpolate() {
            try {
                String[] xValuesStr =
editTextXValues.getText().toString().split(",");
                String[] yValuesStr =

```

```

editTextYValues.getText().toString().split(",");
    double[] x = new double[xValuesStr.length];
    double[] y = new double[yValuesStr.length];

    for (int i = 0; i < xValuesStr.length; i++) {
        x[i] = Double.parseDouble(xValuesStr[i].trim());
        y[i] = Double.parseDouble(yValuesStr[i].trim());
    }

    double xValue =
Double.parseDouble(editTextXValue.getText().toString());

    String selectedMethod =
spinnerInterpolationMethod.getSelectedItem().toString();
    double result;
    String steps;
    if (selectedMethod.equals("Newton")) {
        NewtonInterpolation newtonInterpolation = new
NewtonInterpolation(x, y);
        result = newtonInterpolation.interpolate(xValue);
        steps = newtonInterpolation.getInterpolationSteps(xValue,
result);
    } else {
        LagrangeInterpolation lagrangeInterpolation = new
LagrangeInterpolation(x, y);
        result = lagrangeInterpolation.interpolate(xValue);
        steps = lagrangeInterpolation.getInterpolationSteps(xValue,
result);
    }

    // Start ResultActivity and pass data via Intent
    Intent intent = new Intent(MainActivity.this,
ResultActivity.class);
    intent.putExtra("interpolatedValue", result);
    intent.putExtra("interpolationSteps", steps);
    startActivity(intent);

    } catch (NumberFormatException e) {
        e.printStackTrace();
        // Handle invalid input
    }
}
}

```

main.java

```

package com.example.myapplicationa;

public class LagrangeInterpolation {
    private double[] x;
    private double[] y;

    public LagrangeInterpolation(double[] x, double[] y) {
        this.x = x;
        this.y = y;
    }

    public double interpolate(double xValue) {
        double result = 0;

        for (int i = 0; i < x.length; i++) {
            double term = y[i];
            for (int j = 0; j < x.length; j++) {
                if (j != i) {
                    term *= (xValue - x[j]) / (x[i] - x[j]);
                }
            }
            result += term;
        }

        return result;
    }

    public String getInterpolationSteps(double xValue, double result) {
        StringBuilder steps = new StringBuilder();

        steps.append("Lagrange interpolation steps:\n\n");

        // Display Lagrange basis polynomials
        steps.append("Lagrange Basis Polynomials:\n");
        for (int i = 0; i < x.length; i++) {
            steps.append("L").append(i).append("(x) = ");
            double denominator = 1;
            for (int j = 0; j < x.length; j++) {
                if (j != i) {
                    steps.append("(x - ").append(x[j]).append(")");
                    denominator *= (x[i] - x[j]);
                    if (j < x.length - 1) {
                        steps.append(" * ");
                    }
                }
            }
            steps.append(" / ").append(denominator).append("\n");
        }
        steps.append("\n");

        // Display interpolation formula
        steps.append("Interpolation Formula:\n");
        steps.append("f(x) = ");
        for (int i = 0; i < x.length; i++) {
            steps.append(y[i]);

```

```

        steps.append(" * L").append(i).append("(x)");
        if (i < x.length - 1) {
            steps.append(" + ");
        }
    }
    steps.append("\n\n");

    // Display interpolation steps
    steps.append("Interpolation Steps:\n");
    steps.append("f(").append(xValue).append(") = ").append(result);

    return steps.toString();
}

```

lagrangeinterpolation.java

```

package com.example.myapplicationa;

public class NewtonInterpolation {
    private double[] x;
    private double[] y;
    private double[][] dividedDifferences;

    public NewtonInterpolation(double[] x, double[] y) {
        this.x = x;
        this.y = y;
        calculateDividedDifferences();
    }

    private void calculateDividedDifferences() {
        int n = x.length;
        dividedDifferences = new double[n][n];

        // Initialize first column with y values
        for (int i = 0; i < n; i++) {
            dividedDifferences[i][0] = y[i];
        }

        // Calculate divided differences
        for (int j = 1; j < n; j++) {
            for (int i = 0; i < n - j; i++) {
                dividedDifferences[i][j] = (dividedDifferences[i + 1][j - 1]
- dividedDifferences[i][j - 1]) / (x[i + j] - x[i]);
            }
        }
    }

    public double interpolate(double xValue) {
        double result = dividedDifferences[0][0];
        double term = 1;

        for (int i = 1; i < x.length; i++) {
            term *= (xValue - x[i - 1]);
            result += term * dividedDifferences[0][i];
        }

        return result;
    }

    public String getInterpolationSteps(double xValue, double result) {
        StringBuilder steps = new StringBuilder();

        steps.append("Newton's divided difference interpolation steps:\n\n");

        // Display divided differences table
        steps.append("Divided Differences Table:\n");
        for (int i = 0; i < x.length; i++) {
            for (int j = 0; j < x.length - i; j++) {
                steps.append(dividedDifferences[j][i]).append("\t");
            }
            steps.append("\n");
        }
    }
}

```

```

    }
    steps.append("\n");

    // Display interpolation formula
    steps.append("Interpolation Formula:\n");
    steps.append("f(x) = ");
    for (int i = 0; i < x.length; i++) {
        if (i != 0) {
            steps.append(" + ");
        }
        steps.append(dividedDifferences[0][i]);
        for (int j = 0; j < i; j++) {
            steps.append(" * (x - ").append(x[j]).append(")");
        }
    }
    steps.append("\n\n");

    // Display interpolation steps
    steps.append("Interpolation Steps:\n");
    steps.append("f(").append(xValue).append(") = ").append(result);

    double term = 1;
    for (int i = 0; i < x.length - 1; i++) {
        steps.append(" + (").append(xValue).append(" - ").append(x[i]).append(")");
        term *= (xValue - x[i]);
        steps.append(" * ").append(dividedDifferences[0][i + 1]);
    }

    return steps.toString();
}
}

```

newtoninterpolation.java

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16dp"
    tools:context=".ResultActivity">

    <Button
        android:id="@+id/buttonReturn"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Return"/>

    <TextView
        android:id="@+id/textViewResult"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textSize="18sp"
        android:textStyle="bold"
        android:layout_marginTop="16dp"/>

    <ScrollView
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_weight="1">

        <TextView
            android:id="@+id/textViewSolution"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:textSize="18sp"
            android:textStyle="bold"
            android:layout_marginTop="16dp"/>

    </ScrollView>

</LinearLayout>
```

Activityresult.xml


```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16dp"
    tools:context=".MainActivity">

    <EditText
        android:id="@+id/editTextXValues"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter X values (comma-separated)"/>

    <EditText
        android:id="@+id/editTextYValues"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter Y values (comma-separated)"/>

    <EditText
        android:id="@+id/editTextXValue"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter X value to interpolate"/>

    <Spinner
        android:id="@+id/spinnerInterpolationMethod"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"/>

    <Button
        android:id="@+id/buttonInterpolate"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Interpolate"/>

</LinearLayout>
```

Activitymain.xml