Program 2. Create an application to concatenate two given Strings. (Consider changing the color of the result string to GREEN*)

Activity_main.xml

```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="fill_parent"
    android:layout_height="fill_parent"
    android:background="#E1B04C"
    android:orientation="vertical"
    android:weightSum="1">

    <LinearLayout
        android:id="@+id/linearLayout5"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="5pt"
        android:layout_marginTop="5pt"
        android:layout_marginRight="5pt">

        <TextView
            android:id="@+id/id1"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:capitalize="words"
            android:text="Concatination"
            android:textAlignment="center"
            android:textAllCaps="true"
            android:textSize="20sp"
            tools:ignore="InvalidId" />
    </LinearLayout>

    <LinearLayout
        android:id="@+id/linearLayout1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="10pt"
        android:layout_marginTop="3pt"
```

```xml
        android:layout_marginRight="10pt">

    <EditText
        android:id="@+id/etNum1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginRight="5pt"
        android:layout_weight="1"
        android:hint="String 1"
        android:inputType="text"></EditText>

    <EditText
        android:id="@+id/etNum2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="5pt"
        android:layout_weight="1"
        android:hint="String 2"
        android:inputType="text"></EditText>
</LinearLayout>

<LinearLayout
    android:id="@+id/linearLayout2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="5pt"
    android:layout_marginTop="3pt"
    android:layout_marginRight="5pt">

    <Button
        android:id="@+id/button"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:text="concat"
        android:textSize="8pt"></Button>
</LinearLayout>

<TextView
    android:id="@+id/tvResult"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="5pt"
    android:layout_marginTop="3pt"
    android:layout_marginRight="5pt"
    android:layout_weight="0.07"

    android:gravity="center_horizontal"
```

```
        android:textColor="#19AC2D"
        android:textSize="12pt"></TextView>
</LinearLayout>
```

MainActivity.java

```java
package com.example.concatination;

import android.os.Bundle;
import android.text.TextUtils;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import androidx.appcompat.app.AppCompatActivity;
public class MainActivity extends AppCompatActivity implements
        View.OnClickListener{
    EditText etNum1;
    EditText etNum2;
    Button button;
    TextView tvResult;
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
// find the elements
        etNum1 = (EditText) findViewById(R.id.etNum1);
        etNum2 = (EditText) findViewById(R.id.etNum2);
        button = (Button) findViewById(R.id.button);
        tvResult = (TextView) findViewById(R.id.tvResult);
// set a listener
        button.setOnClickListener(this);
    }
    @Override
    public void onClick(View v) {
// TODO Auto-generated method stub

        String num1;
        String num2;
// check if the fields are empty
        if (TextUtils.isEmpty(etNum1.getText().toString())
                || TextUtils.isEmpty(etNum2.getText().toString())) {
            return;
        }
// read EditText and fill variables with numbers
        num1 = etNum1.getText().toString();
```

```
        num2 = etNum2.getText().toString();

// write operation into oper, we will use it later for output
        switch (v.getId()) {
           case R.id.button:
              tvResult.setText(num1 + num2 );
              break;
           default:
              break;
        }
// form the output line
    }
}
```

Output