



AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools">
    <application android:allowBackup="true"
        android:icon="@mipmap/ic_launcher" android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportRtl="true" android:theme="@style/AppTheme">
        <activity android:name=".MainActivity"
            android:exported="true"> <intent-filter>
            <action android:name="android.intent.action.MAIN" />
            <category android:name="android.intent.category.LAUNCHER" />
        </intent-filter> </activity>
    </application> </manifest>
```

Activity_main.java

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <LinearLayout android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical"
        android:gravity="center"
        android:background="@drawable/aaa">
        <TextView android:id="@+id/textView"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="ARUNA"
            android:textSize="25sp"
            android:textColor="@android:color/white" />
        <Button android:id="@+id/changeColorButton"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Change Color"
            android:layout_marginTop="16dp"
            android:backgroundTint="@android:color/white"
            android:textColor="@android:color/black" />
        <Button android:id="@+id/changeFontButton"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Change Font"
            android:layout_marginTop="16dp"
            android:backgroundTint="@android:color/white"
            android:textColor="@android:color/black" />
    </LinearLayout></RelativeLayout>
```

MainActivity.xml

```
package com.example.myapplication;
import android.content.DialogInterface;
import android.graphics.Color;
import android.graphics.Typeface;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;
import androidx.annotation.NonNull;
import androidx.appcompat.app.AlertDialog;
import androidx.appcompat.app.AppCompatActivity;
import com.google.android.material.color.MaterialColors;
import java.util.ArrayList;
import java.util.List;
public class MainActivity extends AppCompatActivity {
```

```

private TextView textView;
private Button changeColorButton;
private Button changeFontButton;
@Override protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState); setContentView(R.layout.activity_main);
    textView = findViewById(R.id.textView);
    changeColorButton = findViewById(R.id.changeColorButton);
    changeFontButton = findViewById(R.id.changeFontButton);
    changeColorButton.setOnClickListener(new View.OnClickListener() {
        @Override public void onClick(View v) {
            showColorPicker(); });
    changeFontButton.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            showFontPicker(); });
    private void showColorPicker() {
        final List<String> colors = new ArrayList<>();
        colors.add("Red"); colors.add("Green");
        colors.add("Blue"); colors.add("Black");
        colors.add("Yellow");
        AlertDialog.Builder builder = new AlertDialog.Builder(this);
        builder.setTitle("Choose a Color")
        .setItems(colors.toArray(new String[0]), new DialogInterface.OnClickListener() {
            @Override public void onClick(DialogInterface dialog, int which) {
                switch (which) {
                    case 0: textView.setTextColor(Color.RED);
                    break;
                    case 1: textView.setTextColor(Color.GREEN);
                    break;
                    case 2: textView.setTextColor(Color.BLUE);
                    break;
                    case 3: textView.setTextColor(Color.BLACK);
                    break;
                    case 4: textView.setTextColor(Color.YELLOW);
                    break; }
                Toast.makeText(MainActivity.this, "Color changed!", Toast.LENGTH_SHORT).show(); } });
        builder.show(); }
    private void showFontPicker() {
        final List<String> fonts = new ArrayList<>();
        fonts.add("Normal");
        fonts.add("Bold");
        fonts.add("Italic");
        fonts.add("Monospace");
        AlertDialog.Builder builder = new AlertDialog.Builder(this);
        builder.setTitle("Choose a Font")
        .setItems(fonts.toArray(new String[0]), new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog, int which) {
                switch (which) {
                    case 0: textView.setTypeface(Typeface.DEFAULT);
                    break;
                    case 1: textView.setTypeface(Typeface.DEFAULT_BOLD);
                    break;
                    case 2: textView.setTypeface(Typeface.defaultFromStyle(Typeface.ITALIC));
                    break;
                    case 3: textView.setTypeface(Typeface.MONOSPACE);
                    break; }
                Toast.makeText(MainActivity.this, "Font changed!", Toast.LENGTH_SHORT).show(); } });
        builder.show();
    }
}

```