

```

package com.example.application;

import android.app.Dialog;
import android.content.Context;
import android.content.*;
import android.view.*;
import android.view.Window;
import android.view.View;
import android.view.ViewGroup;
import android.widget.*;
import android.widget.TextView;
import android.widget.RadioGroup.LayoutParams;
import android.widget.LinearLayout;
import android.app.Activity;
import android.os.Bundle;
import android.util.AttributeSet;
import android.animation.ValueAnimator;
import android.graphics.LinearGradient;
import android.graphics.Shader;
import android.os.Bundle;
import android.view.animation.LinearInterpolator;
import android.os.*;
import android.graphics.Typeface;
import android.graphics.*;
import android.graphics.BitmapShader;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.graphics.Canvas;
import android.graphics.LinearGradient;
import android.graphics.Matrix;
import android.graphics.Paint;
import android.graphics.Shader;
import android.graphics.*;
import android.graphics.Typeface;
import android.graphics.drawable.*;
import android.graphics.LinearGradient;
import android.graphics.Matrix;
import android.graphics.Shader;
import android.util.AttributeSet;
import android.view.animation.LinearInterpolator;
import android.widget.*;
import android.animation.*;
import android.animation.ValueAnimator;
import android.os.Bundle;
import java.io.*;

public class CustomRainbowDialog {

    private static Dialog dialog;
    private static final String titleDialog = "CustomDialog";
    private static final String messageDialog = "This is Custom Dialog!";
    private static final String buttonOk = "OK";
    private static int bgColors = 0xffffffff;

    public static void showDialogs(Context context) {

        dialog = new Dialog(context);
        dialog.requestWindowFeature(Window.FEATURE_NO_TITLE);
        final LinearLayout.LayoutParams params = new LinearLayout.LayoutParams(ViewGroup.LayoutParams.MATCH_PARENT,ViewGroup.LayoutParams.WRAP_CONTENT);
        params.setMargins(8,8,8,8);
        final LinearLayout root = new LinearLayout(context);
        root.setOrientation(1);
        root.setBackgroundColor(bgColors);

        final LinearLayout ContentRoot = new LinearLayout(context);
        ContentRoot.setOrientation(1);
        ContentRoot.setBackgroundColor(Color.TRANSPARENT);
        root.addView(ContentRoot,640,ViewGroup.LayoutParams.WRAP_CONTENT);

        final LinearLayout Contents = new LinearLayout(context);
        Contents.setOrientation(1);
        Contents.setBackgroundColor(Color.TRANSPARENT);
        ContentRoot.addView(Contents,params);

        final TextView title = new TextView(context);
        title.setText(titleDialog);
        title.setShadowLayer(3,4,0,Color.GRAY);
        title.setTextColor(0xff000000);
        title.setTextSize(25);
        title.setTypeface(Typeface.DEFAULT_BOLD);
        Contents.addView(title,params);

        final TextView message = new TextView(context);
        message.setText(messageDialog);
        message.setShadowLayer(3,4,0,Color.GRAY);
        message.setTextColor(0xff000000);
        message.setTextSize(15);
        message.setTypeface(Typeface.DEFAULT_BOLD);
        message.setGravity(1);
        Contents.addView(message,params);

        final Button button_ok = new Button(context);
        button_ok.setHint(buttonOk);
        button_ok.setHintTextColor(0xff000000);
        button_ok.setShadowLayer(3,4,0,Color.GRAY);
        Contents.addView(button_ok,params);
        setCornerRadius(button_ok,10);

        animateRainbowText(title);
        animateRainbowText(message);
        animateRainbowText(button_ok);

        dialog.setContentView(root);
        dialog.setCancelable(false);

        button_ok.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                dialog.dismiss();
            }
        });
    }
}

```

```

    }
});

    dialog.show();
}
private static View setCornerRadius(View view, float radius) {
    // Create a GradientDrawable
    final GradientDrawable gradientDrawable = new GradientDrawable();

    // Set the corner radius (in pixels, convert dp to pixels if needed)
    //float cornerRadius = 10f; // 10dp
    float cornerRadius = radius; // get float Radius
    gradientDrawable.setCornerRadius(cornerRadius);
    int[] Colors = new int[] {0xff3298f2, 0xff6742d7};

    // Set the background color
    gradientDrawable.setColors(new int[] {0xff3298f2, 0xff6742d7}); // Replace with your color

    gradientDrawable.setGradientType(GradientDrawable.LINEAR_GRADIENT);

    // Set the gradient direction (orientation)
    gradientDrawable.setOrientation(GradientDrawable.Orientation.TL_BR); // Top-Left to Bottom-Right
    // Apply the GradientDrawable as the button's background
    view.setBackground(gradientDrawable);
    view.setElevation(10);
    return view;
}
private static void animateRainbowText(final TextView textView) {
    // Define the colors for the gradient
    final int[] colors = {

        0xFFFF0000, // Red
        0xFFFF7F00, // Orange
        0xFFFFFF00, // Yellow
        0xFF00FF00, // Green
        0xFF00FFFF, // Cyan
        0xFF0000FF, // Blue
        0xFF4B0082, // Indigo
        0xFF9400D3, // Violet
        0xFFC12BFF, // Light Pink
        0xFFFF246A, // Light Red
        Color.RED, Color.YELLOW, Color.GREEN, Color.CYAN, Color.BLUE, Color.CYAN, Color.GREEN, Color.YELLOW, Color.RED
    };

    final int[] colors2 = {
        //Best For Rainbow Colors
        Color.RED, Color.YELLOW, Color.GREEN, Color.CYAN, Color.BLUE, Color.CYAN, Color.GREEN, Color.YELLOW, Color.RED
    };

    //final float[] positions = {0.0f, 0.13f, 0.26f, 0.39f, 0.52f, 0.65f, 0.78f, 0.91f, 1.0f}; /* set ColorGradient Positions */

    // Create a ValueAnimator to animate the gradient
    ValueAnimator animator = ValueAnimator.ofFloat(0, 1);
    animator.setDuration(5000); // Duration of the animation in milliseconds
    animator.setRepeatCount(ValueAnimator.INFINITE); // Repeat indefinitely
    animator.setInterpolator(new LinearInterpolator());
    animator.addUpdateListener(new ValueAnimator.AnimatorUpdateListener() {
        @Override
        public void onAnimationUpdate(ValueAnimator animation) {
            float value = (float) animation.getAnimatedValue();

            // Calculate the start and end positions for the gradient
            int startX = (int) (value * textView.getWidth());
            int endX = startX + textView.getWidth();

            // Create a LinearGradient shader
            LinearGradient gradient = new LinearGradient(
                startX, 0, endX, 0,
                colors,
                (null),
                //positions, /* If float position Enabled then (null) Remove||Comment */
                Shader.TileMode.REPEAT
            );

            // Apply the shader to the TextView's paint
            textView.getPaint().setShader(gradient);

            // Invalidate the TextView to redraw it
            textView.invalidate();
        }
    });

    // Start the animation
    animator.start();
}
}

/* onCreate Java Code For ShowDialogs context is equal to activity*/
// CustomRainbowDialog.showDialogs(context.this); && showDialogs(this);

```