```java
package com.zenith.purejava.dialog;

import android.app.AlertDialog;
import android.content.Context;
import android.graphics.Color;
import android.graphics.drawable.GradientDrawable;
import android.os.Build;
import android.view.Gravity;
import android.view.View;
import android.view.ViewGroup;
import android.widget.EditText;
import android.widget.LinearLayout;
import android.widget.TextView;
import android.widget.Toast;

public class CustomMaterialDialog {

    private static final String CORRECT_PASSWORD = "zenith"; // Set the correct password directly
    private static AlertDialog dialog;

    public static void showDialog(final Context context) {
        // Create an AlertDialog.Builder
        AlertDialog.Builder builder = new AlertDialog.Builder(context);

        // Create the parent LinearLayout
        LinearLayout linearLayout = new LinearLayout(context);
        linearLayout.setOrientation(LinearLayout.VERTICAL);
        linearLayout.setPadding(48, 48, 48, 48); // Add padding for a material look
        linearLayout.setGravity(Gravity.CENTER_HORIZONTAL);
        linearLayout.setLayoutParams(new LinearLayout.LayoutParams(
                ViewGroup.LayoutParams.MATCH_PARENT,
                ViewGroup.LayoutParams.WRAP_CONTENT));

        // Create the title TextView
        TextView textViewTitle = new TextView(context);
        textViewTitle.setText("Enter password to continue");
        textViewTitle.setTextSize(20);
        textViewTitle.setTextColor(Color.BLACK);
        textViewTitle.setPadding(0, 0, 0, 40); // Bottom padding for spacing
        textViewTitle.setGravity(Gravity.CENTER_HORIZONTAL);
        textViewTitle.setLayoutParams(new LinearLayout.LayoutParams(
                ViewGroup.LayoutParams.WRAP_CONTENT,
                ViewGroup.LayoutParams.WRAP_CONTENT));

        // Create the EditText for password input
        final EditText editTextPassword = new EditText(context);
        editTextPassword.setHint("Enter Password");
        editTextPassword.setHintTextColor(Color.parseColor("#607D8B"));
        editTextPassword.setTextColor(Color.BLACK);
        editTextPassword.setTextSize(18);
        editTextPassword.setPadding(20, 20, 20, 20);
        editTextPassword.setLayoutParams(new LinearLayout.LayoutParams(
                ViewGroup.LayoutParams.MATCH_PARENT,
                ViewGroup.LayoutParams.WRAP_CONTENT));

        // Set background for EditText
        GradientDrawable editTextBackground = new GradientDrawable();
        editTextBackground.setCornerRadius(12);
        editTextBackground.setStroke(2, Color.parseColor("#BDBDBD"));
        editTextPassword.setBackground(editTextBackground);

        // Create a spacer view for space between EditText and Login Button
        View spacer = new View(context);
        LinearLayout.LayoutParams spacerParams = new LinearLayout.LayoutParams(
                ViewGroup.LayoutParams.MATCH_PARENT,
                16 // Height of the spacer
        );
        linearLayout.addView(spacer, spacerParams);

        // Create the Login Button TextView
        TextView textViewLogin = new TextView(context);
        textViewLogin.setText("Login");
        textViewLogin.setTextSize(18);
        textViewLogin.setTextColor(Color.WHITE);
        textViewLogin.setGravity(Gravity.CENTER);
        textViewLogin.setPadding(20, 20, 20, 20);

        // Add margin to create space between EditText and Login Button
        LinearLayout.LayoutParams loginParams = new LinearLayout.LayoutParams(
                ViewGroup.LayoutParams.MATCH_PARENT,
                ViewGroup.LayoutParams.WRAP_CONTENT
        );
        loginParams.topMargin = 16; // Margin in pixels
        textViewLogin.setLayoutParams(loginParams);

        // Set background for Login Button
        GradientDrawable loginBackground = new GradientDrawable();
        loginBackground.setCornerRadius(12);
        loginBackground.setColor(Color.parseColor("#6200EE")); // Material primary color
        textViewLogin.setBackground(loginBackground);

        // Add shadow to the button if API level >= 21
        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.LOLLIPOP) {
            textViewLogin.setElevation(8);
        }

        // Add views to the LinearLayout
        linearLayout.addView(textViewTitle);
        linearLayout.addView(editTextPassword);
        linearLayout.addView(textViewLogin);

        // Set background for the LinearLayout
        GradientDrawable linearLayoutBackground = new GradientDrawable();
```

```java
        linearLayoutBackground.setCornerRadius(16);
        linearLayoutBackground.setColor(Color.WHITE);
        linearLayout.setBackground(linearLayoutBackground);

        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.LOLLIPOP) {
            linearLayout.setElevation(16);
        }

        // Set the custom view to the AlertDialog
        builder.setView(linearLayout);

        // Create the AlertDialog
        dialog = builder.create();

        // Make the dialog non-cancelable
        dialog.setCancelable(false);

        // Set the click listener for the Login button
        textViewLogin.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                // Get the entered password
                String enteredPassword = editTextPassword.getText().toString();

                // Validate the password
                validatePassword(context, enteredPassword);
            }
        });

        // Show the dialog
        dialog.show();
    }

    private static void validatePassword(Context context, String enteredPassword) {
        if (enteredPassword.equals(CORRECT_PASSWORD)) {
            // If correct, dismiss the dialog and show a success toast
            dialog.dismiss();
            showCustomToast(context, "Password Correct!");
        } else {
            // If incorrect, show a failure toast
            showCustomToast(context, "Wrong Password!");
        }
    }

    private static void showCustomToast(Context context, String message) {
        // Create a new LinearLayout for the custom toast layout
        LinearLayout toastLayout = new LinearLayout(context);
        toastLayout.setOrientation(LinearLayout.HORIZONTAL);
        toastLayout.setPadding(10, 10, 10, 10);

        // Create a GradientDrawable for the background
        GradientDrawable background = new GradientDrawable();
        background.setColor(Color.parseColor("#6200EE")); // Set background color
        background.setCornerRadius(8); // Rounded corners

        // Set the background to the LinearLayout
        toastLayout.setBackground(background);

        // Create a TextView for the message
        TextView toastText = new TextView(context);
        toastText.setText(message);
        toastText.setTextColor(Color.WHITE); // Text color
        toastText.setTextSize(16); // Text size
        toastText.setPadding(20, 10, 20, 10); // Padding for the text

        // Add the TextView to the LinearLayout
        toastLayout.addView(toastText);

        // Create and show the Toast
        Toast toast = new Toast(context);
        toast.setView(toastLayout);
        toast.setGravity(Gravity.CENTER_HORIZONTAL | Gravity.BOTTOM, 0, 100); // Position the toast
        toast.setDuration(Toast.LENGTH_SHORT); // Duration of the toast
        toast.show();
    }
}


/* onCreate Java Code For ShowDialog context is equal to activity*/
// CustomMaterialDialog.showDialog(context.this); && showDialog(this);
```