

Lab Assignment: Android App Development

Objective:

Develop a simple Android app to understand the basics of activities, layouts, and user interaction.

Requirements:

1. Create a new Android project in Android Studio with the following components:
 - An activity named `MainActivity`.
 - A layout with:
 - An `EditText` to input the user's name.
 - A `Button` labeled "Greet Me".
 - A `TextView` to display the greeting message.
2. Write Java or Kotlin code in `MainActivity` to:
 - Display a greeting message like "Hello, [UserName]!" when the button is clicked.
 - Validate that the input field is not empty before displaying the message.

Code Starter:

```
// MainActivity.java
package com.example.greetapp;

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

        EditText nameInput = findViewById(R.id.nameInput);
        TextView greetingView = findViewById(R.id.greetingView);
        Button greetButton = findViewById(R.id.greetButton);

        greetButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String userName = nameInput.getText().toString();
                // TODO: Validate input and set greeting message
            }
        });
    }
}
```

```

        }
    });
}

```

Exercises:

1. Add a second button to clear the input field and reset the greeting message.
2. Use a **Toast** to show a short message if the input field is left empty.
3. Create a second activity with a "Goodbye" message and navigate to it using an **Intent**.

Bonus Tasks:

1. Implement a feature to save the user's name using **SharedPreferences** and display it when the app restarts.
2. Add a spinner to select a greeting language (e.g., English, Spanish, French) and update the greeting accordingly.
3. Use a custom style or theme for your app.

Submission:

Submit your Android project folder and screenshots of the app running on an emulator or device.