

Lab04

Bài 1.

Yêu cầu:

Xây dựng 2 giao diện theo mẫu:



Trên mỗi giao diện có nút để chuyển đến giao diện kia và ngược lại.

Hướng dẫn:

- Trên activity_main tạo giao diện thứ nhất. Tiếp tục tạo giao diện thứ 2 trong activity_second (id của TextView và Button nên đặt kèm theo tên giao diện)
- Trong thư mục java, MainActivity.java xây dựng chức năng chuyển đến Activity thứ 2:

```
public class MainActivity extends AppCompatActivity {  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
  
        Button buttonMain = findViewById(R.id.buttonMain);  
        buttonMain.setOnClickListener(new View.OnClickListener() {  
            @Override  
            public void onClick(View v) {  
                //Dùng Intent chuyển từ Activity 1 sang Activity 2  
                Intent intent = new Intent(getApplicationContext(), SecondActivity.class);  
                startActivity(intent);  
            }  
        });  
    }  
}
```

id của Button trong activity_main

//Dùng Intent chuyển từ Activity 1 sang Activity 2

Intent intent = new Intent(getApplicationContext(), SecondActivity.class);

startActivity(intent);

class SecondActivity cần tạo thêm

Trong thư mục java, tiếp tục tạo class có tên SecondActivity để thực hiện việc chuyển từ Activity 2 về Activity 1:

```
public class SecondActivity extends AppCompatActivity{
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_second);

        Button buttonSecond = findViewById(R.id.buttonSecond);
        buttonSecond.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                //Dùng Intent chuyển từ Activity 2 sang Activity 1
                Intent intent = new Intent(getApplicationContext(), MainActivity.class);
                startActivity(intent);
            }
        });
    }
}
```

Lưu ý:

- Có thể khai báo thêm các thư viện nếu cần:

```
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import androidx.appcompat.app.AppCompatActivity;
```

- Cần khai báo Activity 2 mới tạo thêm trong [AndroidManifest.xml](#)

```
<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>
<activity android:name=".SecondActivity"/>
```

Chạy thử ứng dụng để xem sự chuyển đổi giữa 2 Activity.

Bài 2.

Yêu cầu

Xây dựng 2 Activity có giao diện như sau:

The image shows two separate Activity screens side-by-side. Both screens have a light gray background with a thin gray border around the content area. The left screen contains the following text:
Họ và tên của bạn là gì?
Nhập vào Họ và tên...

Điểm của bạn là:
0

The right screen contains the following text:
Họ và tên sinh viên là:
Báo điểm cho sinh viên:
Nhập vào số điểm của sinh viên...

Từ Activity 1:

- Nhập vào họ và tên.
- Click nút Gửi, thông tin nhập vào sẽ gửi sang Activity 2.

Tại Activity 2:

- Hiển thị Họ và tên đã nhập bên Activity 1.
- Nhập điểm số của sinh viên rồi click nút Thông báo.

Điểm số sẽ được chuyển và hiển thị trở lại ở Activity 1.

Hướng dẫn

1. Thiết kế layout.

- Có thể sử dụng LinearLayout để xây dựng 2 layout cho các activity.

- Sử dụng thuộc tính `android:gravity` giúp các Widget con luôn nằm giữa Layout theo cả chiều ngang và chiều dọc:

```
android:gravity="center_horizontal|center_vertical"
```

- Trong Activity 1: Bao gồm các TextView, EditText, Button, TextView và TextView.
- Trong Activity 2: Bao gồm: TextView, TextView, TextView, EditText, Button.

Một vài gợi ý

- Đặt id cho các Widget để có thể phân biệt rõ ràng.
- `android:hint` tạo text gợi ý, ví dụ Nhập vào Họ và tên...
- `android:layout_marginXXX` tạo khoảng cách giữa các widget
- `android:inputType` quy định kiểu dữ liệu nhập vào, ví dụ
`android:inputType="number"` quy định kiểu dữ liệu nhập vào là số

2. Xử lý yêu cầu.

- ❖ Để gửi dữ liệu ta làm như sau: Trong sự kiện onClick của Button khởi tạo:

```
Intent intent = new Intent(<Class đang tương tác>.this,
<Class cần chuyển đến>.class);
```

- ❖ Để gửi dữ liệu ta dùng:

```
intent.putExtra("Key",<Dữ liệu cần gửi>);
```

Đồng thời để nhận dữ liệu từ một Activity khác trả về Activity gọi nó, ta sử dụng `registerForActivityResult`

- ❖ Sử dụng `registerForActivityResult` để tạo hàm `getResult` để lắng nghe dữ liệu trả về

Các xử lý trong MainActivity.java (Activity 1):

- Khai báo các biến chứa các giá trị trên layout:

```
editTextHoTen = findViewById(R.id.editTextHoTen);
buttonGui = findViewById(R.id.buttonGui);
textViewDiem = findViewById(R.id.textViewDiem);
```

- Xử lý Gửi thông tin từ Activity 1 sang Activity 2: Gửi HoTen

```
//Sự kiện click vào nút gửi
buttonGui.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        //Dùng Intent chuyển từ Activity 1 sang Activity 2
        Intent intent = new Intent( MainActivity.this,SecondActivity.class);
        //Chuỗi nhập vào sẽ lưu trong biến HoTen và gửi đi
        intent.putExtra("HoTen",editTextHoTen.getText().toString());
        nhanDienSV.launch(intent);
    }
});
```

- Xử lý thông tin nhận được từ Activity 2: Nhận diemGui

```
//Xử lý kết quả trả về từ Activity2
ActivityResultLauncher<Intent> nhanDienSV = registerForActivityResult(
    new ActivityResultContracts.StartActivityForResult(),
    new ActivityResultCallback<ActivityResult>() {
        @Override
        public void onActivityResult(ActivityResult ketQuaDien) {
            if (ketQuaDien.getResultCode()==RESULT_OK){
                Intent intent = ketQuaDien.getData();
                if(intent != null){
                    Bundle bundle = intent.getExtras();
                    String diemNhan = bundle.getString("diemGui");
                    //diemGui là giá trị điểm gửi về từ Activity2
                    textViewDiem.setText(diemNhan);
                }
            }
        }
    });
};
```

Các xử lý trong SecondActivity.java (Activity 2)

- Khai báo biến chứa các giá trị trên layout:

```
buttonThongBao = findViewById(R.id.buttonThongBao);
textViewHoTenDaNhap = findViewById(R.id.textViewHoTenDaNhap);
editTextDiem = findViewById(R.id.editTextDiem);
```

- Xử lý giá trị nhận được từ Activity 1: Nhận HoTen

```
//Hiển thị tên đã nhập ở Activity nhận thông qua biến HoTen từ Activity_main đã gửi
Intent intent = getIntent();
textViewHoTenDaNhap.setText(intent.getStringExtra("HoTen"));
```

- Xử lý gửi Điểm từ Activity 2 về Activity 1: Gửi đi diemGui

```
//Gửi thông tin từ Activity 2 về Activity 1
buttonThongBao.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        //Dùng Intent chuyển từ Activity 2 sang Activity 1
        String diemSV = editTextDiem.getText().toString();
        Intent intent = new Intent();
        Bundle bundle = new Bundle();
        bundle.putString("diemGui", diemSV);
        intent.putExtras(bundle);
        setResult(RESULT_OK, intent);
        finish();
    }
});
```

Thực thi ứng dụng để kiểm tra kết quả.