

INTENT, INTENT FILTER

- Khái niệm intent
- Các loại intent
- Các tham số chính của Intent
- Các cặp giá trị Action/data
- Sử dụng Bundle
- Bộ lọc filter



1. INTENT

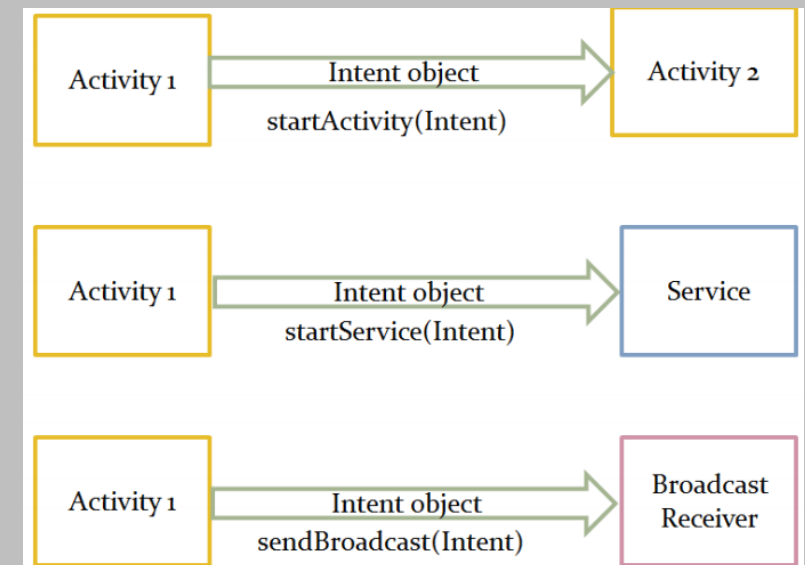
- Các thành phần chính của ứng dụng (activity, service, broadcast receiver, content provider) được kích hoạt thông qua **Intent**.
- **Intent** là những tín nhắn không đồng bộ cho phép các thành phần ứng dụng yêu cầu thực hiện chức năng từ các thành phần android khác.
- **Intent** ràng buộc giữa các thành phần trong cùng ứng dụng hoặc các ứng dụng khác nhau.
- Ví dụ:
 - Activity có thể gọi một Activity bên ngoài để chụp ảnh.

1. INTENT

- **Intent**- đối tượng của **android.content.Intent** sử dụng để:
 - Bắt đầu một Activity
 - Bắt đầu một dịch vụ (Service).
 - Gửi một quảng bá – broadcast receiver
- Có 2 loại Intent:
 - **Intent** tường minh (explicit intent)
 - **Intent** không tường minh (implicit intent).
- **Intent** có thể sử dụng Bundle để chuyển dữ liệu

1. INTENT

- Intent và các thành phần ứng dụng:
 - `startActivity(Intent)` hoặc `startActivityForResult()`: khởi tạo activity/ khởi tạo và nhận kết quả trả về
 - `startService(... Intent)` hoặc `bindService(intent,...)` thiết lập kết nối giữa thành phần gọi và dịch vụ đích
 - `sendBroadcast(... Intent)`: gửi Intent tới thành phần `BroadcastReceiver` quan tâm đến



1.1 EXPLICIT INTENT – INTENT TƯỜNG MINH

- Intent tường minh (Explicit intents): Là những intent **chỉ định rõ** tên của các thành phần mục tiêu sẽ nhận và xử lý bằng cách thiết lập giá trị phù hợp (setComponent() hoặc setClass()).
- Để khởi động activity dùng phương thức gọi tường minh:
 - **startActivity(context, nameActivity.class);**
- Để khởi động và nhận kết quả trả về ta dùng phương thức :
 - **startActivityForResult().**

1.2. IMPLICIT INTENT (INTENT KHÔNG TƯỞNG MINH)

- Là loại intent không cần chỉ rõ thành phần xử lý mà cần cung cấp đủ các thông tin cần thiết để hệ thống xác định xem nên dùng các thành phần có sẵn nào để chạy tốt nhất
- Ví dụ: khi ứng dụng Android mở một trang web
 1. `Intent i = new Intent(Intent.ACTION_VIEW, Uri.parse("https://www.haui.edu.vn/"));`
 2. `startActivity(i);`

2. DEMO

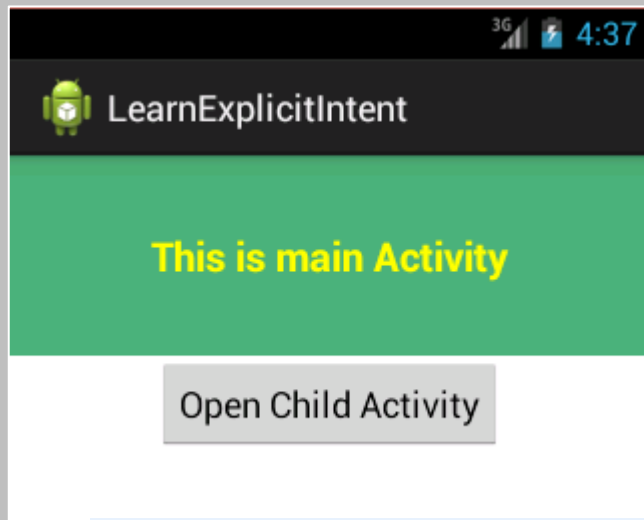
1. Minh họa với Explicit intent

- a. Khởi tạo activity không nhận kết quả trả về
- b. Khởi tạo activity và nhận kết quả trả về

2. Minh họa với Implicit intent

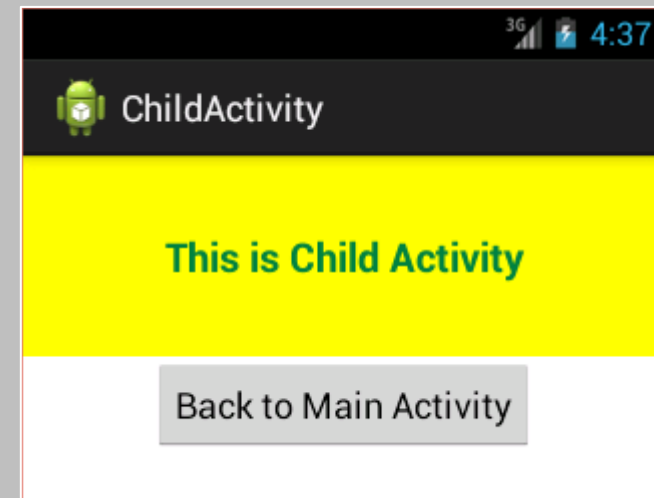
- a. Gọi điện nhắn tin.

2. DEMO EXPLICIT INTENT



btnOpenChildActivity

```
<Button
    android:id="@+id/btnOpenChildActivity"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:text="Open Child Activity" />
```



btnBacktoMainActivity

```
<Button
    android:id="@+id/btnBacktoMainActivity"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:text="Back to Main Activity" />
```


2. DEMO EXPLICIT INTENT (TT)

- Minh họa manifest có 2 activity

```
<manifest ....  
  package="com.vuduong.manhinhchinhphu">  
    <application  
      .....  
      <activity android:name=".MainActivity">  
        <intent-filter>  
          <action android:name="android.intent.action.MAIN" />  
          <category android:name="android.intent.category.LAUNCHER" />  
        </intent-filter>  
      </activity>  
      <activity android:name=".ChidActivity"></activity>  
    </application>  
  </manifest>
```

2.1. DEMO EXPLICIT INTENT

- Minh họa khởi động 1 activity dùng intent

```
public class MainActivity extends Activity {  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
        Button btnOpen=(Button)  
            findViewById(R.id.btnOpenChildActivity);  
        btnOpen.setOnClickListener(new  
            View.OnClickListener() {  
                public void onClick(View arg0) {  
                    doOpenChildActivity();  
                }  
            });  
    }  
    public void doOpenChildActivity()  
    {  
        Intent myIntent=  
            new Intent(this, ChildActivity.class);  
        startActivity(myIntent);  
    }  
    public boolean onCreateOptionsMenu(Menu menu) {  
    }  
}
```

2.1. DEMO EXPLICIT INTENT (TT)

- Dòng lệnh khai báo trong ví dụ trên khi gọi khởi động activity:

```
Intent intent1= new Intent(MainActivity.this,ChildActivity1.class);
```

- Đối tượng Intent là một gói các thông tin xác định ngữ cảnh xuất hiện và đối tượng hiển thị
- Tham số của Intent bao gồm:
 - Context : ngữ cảnh xuất hiện
 - *.class: tên activity muốn hiển thị.

2.2 EXPLICIT INTENT GỬI NHẬN KẾT QUẢ TRỰC TIẾP

Hoạt động trong Activity gửi

- Sử dụng phương thức put(key,value)
 - `intent.putExtra(key,value);`
- Thực hiện:
 - `intent.putExtra("soa",5);` sẽ gửi kèm giá trị 5 qua key “soa”.
 - //yêu cầu start Activity chỉ định
 - `this.startActivity(intent);`

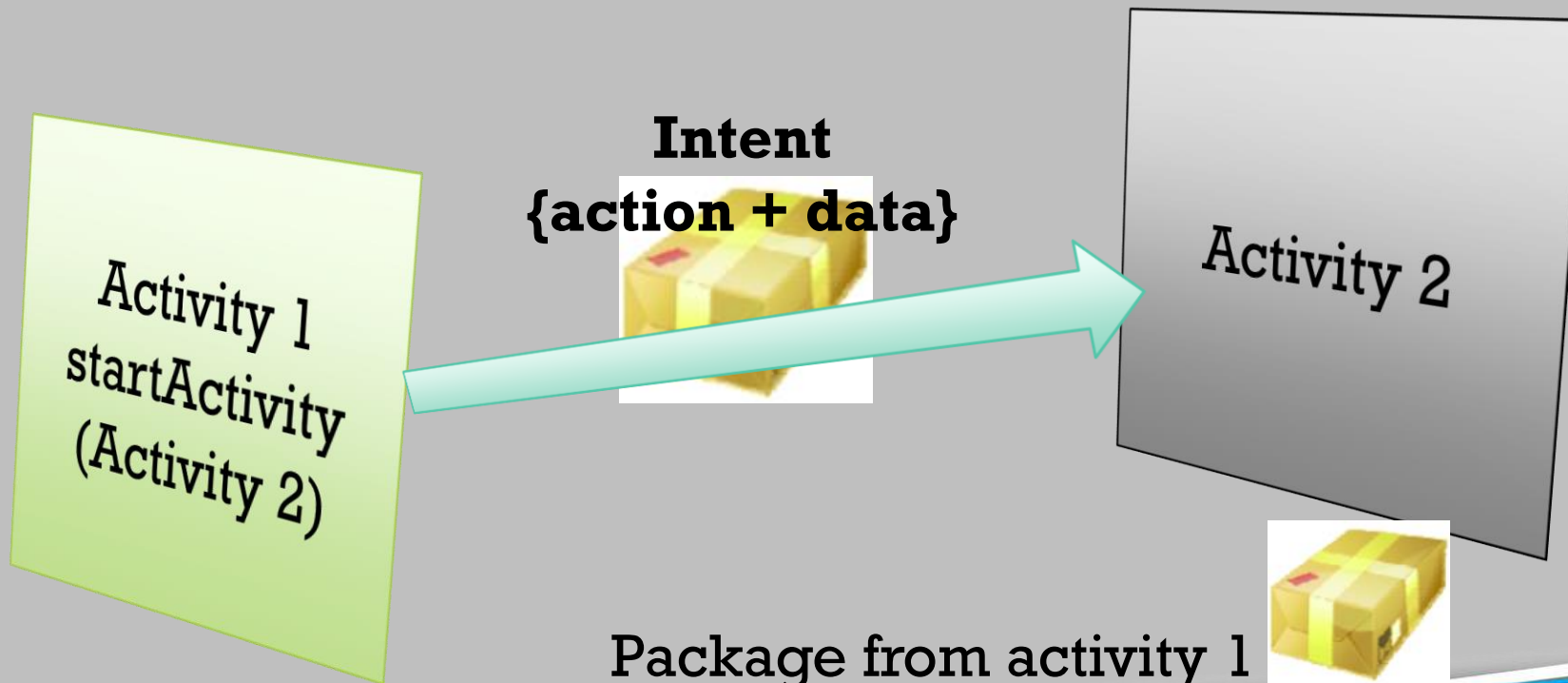
Hoạt động trong Activity nhận

- Sử dụng phương thức get(...)
- Thực hiện:
 - `Intent intent = this getIntent();`
 - `int a = intent.getIntExtra("soa");`



2.2. EXPLICIT INTENT- GỬI NHẬN QUA BUNDLE

- Bundle là cơ chế đơn giản để truyền dữ liệu giữa các Activity



2.2. EXPLICIT INTENT- GỬI NHẬN QUA BUNDLE

- Mỗi Bundle là một tập hợp dạng cặp <name, value>.
- Đưa dữ liệu vào Bundle dùng phương thức **putXXX** và **getXXX** để lấy dữ liệu (dạng đơn trị hay dạng bảng) thuộc các dữ liệu cơ bản.
- Intent sẽ mang dữ liệu: **tênIntent.putExtras(tên_bundle);**
- Ví dụ:

```
Bundle bundle=new Bundle();  
bundle.putInt("soa", 5);  
bundle.putDouble("sob", 5.7);  
//...  
int soa=bundle.getInt("soa");  
double sob=bundle.getDouble("sob");
```



2.2. EXPLICIT INTENT- GỬI NHẬN QUA BUNDLE

- Bundle với đối tượng.
- Sử dụng putSerializable(...)


```
class Person implements Serializable
{
    private int id;
    private String name;
    public Person(int id,String name)
    {
        this.id=id;
        this.name=name;
    }
    public String toString() {
        return this.id+"- "+this.name;
    }
}
```

```
Bundle bundle=new Bundle();
Person p=new Person(1, "teo");
bundle.putSerializable("t1", p);
//...
Person t1=(Person) bundle.getSerializable("t1");
```

2.2 EXPLICIT INTENT – VÍ DỤ PTB2

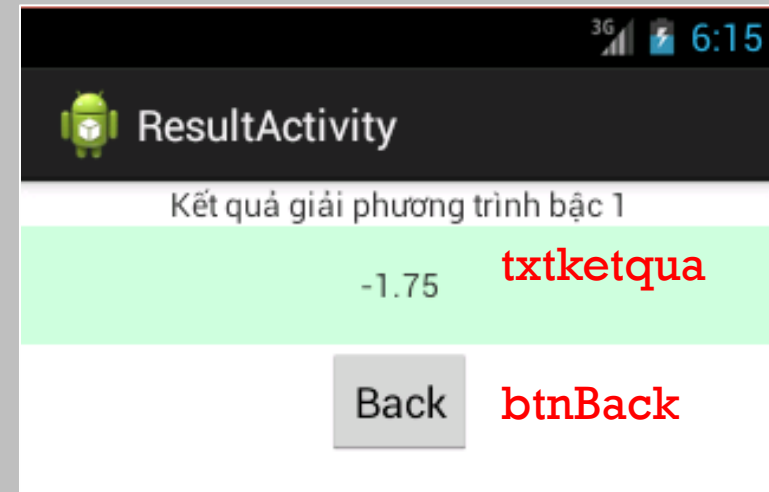
- Ví dụ sử dụng Intent và Bundle:

MainActivity



The screenshot shows the MainActivity interface. At the top, the title bar says "LearnBundle". Below it, a green header bar contains the text "Giải phương trình bậc 1". The main area has two input fields: "Nhập a:" with the value "4" and "Nhập b:" with the value "7". To the right of each input field is a red label: "txta" for the first and "txtb" for the second. Below the input fields is a grey button labeled "Kết quả". At the bottom, there is a red label "btnketqua".

ResultActivity



The screenshot shows the ResultActivity interface. The title bar says "ResultActivity". Below it, a white header bar contains the text "Kết quả giải phương trình bậc 1". The main area has a green bar with the value "-1.75" and a red label "txtketqua" to its right. Below this is a grey button labeled "Back" and a red label "btnBack" to its right.

<http://developer.android.com/reference/android/os/Bundle.html>

2.2 EXPLICIT INTENT – VÍ DỤ PTB2

- Xử lý gửi hai hệ số a,b qua intent sang bên nhận để giải phương trình.

```
public class MainActivity extends Activity {  
    EditText txta,txtb;  
    Button btnketqua;  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
        txta=(EditText) findViewById(R.id.txta);  
        txtb=(EditText) findViewById(R.id.txtb);  
        btnketqua=(Button) findViewById(R.id.btnketqua);  
        btnketqua.setOnClickListener(new View.OnClickListener() {  
            public void onClick(View arg0) {  
                Intent myIntent=new Intent  
                    (MainActivity.this, ResultActivity.class);  
                Bundle bundle=new Bundle();  
                int a=Integer.parseInt(txta.getText().toString());  
                int b=Integer.parseInt(txtb.getText().toString());  
                bundle.putInt("soa", a);  
                bundle.putInt("sob", b);  
                myIntent.putExtra("MyPackage", bundle);  
                startActivity(myIntent);  
            }  
        });  
    }  
}
```



2.2 EXPLICIT INTENT – VÍ DỤ PTB2

- Xử lý nhận dữ liệu và xử lý thông tin nhận được.
- Giải phương trình và hiển thị kết quả tên textview

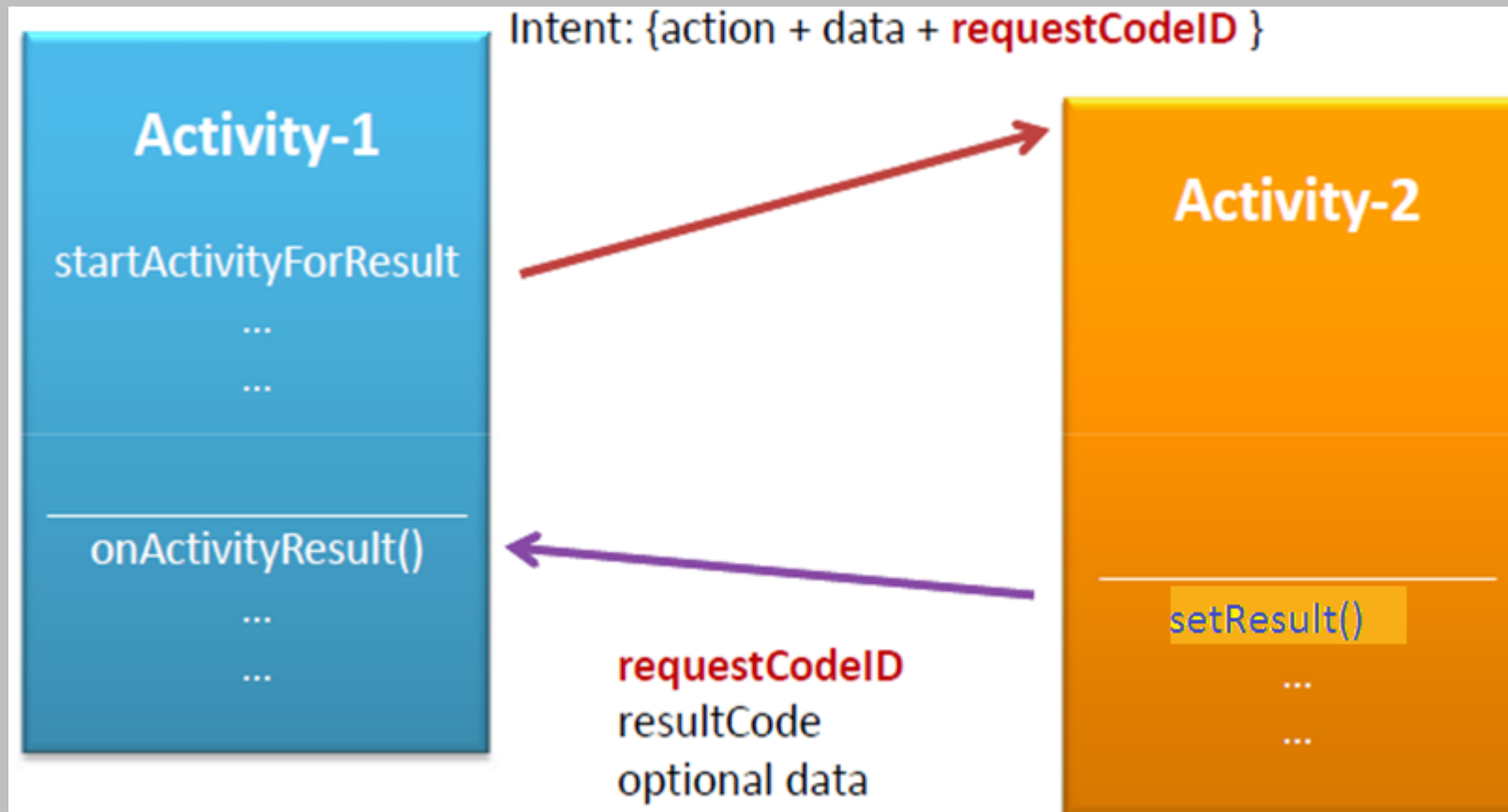
```
public class ResultActivity extends Activity {  
    TextView txtketqua;  
    Button btnBack;  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_result);  
        btnBack=(Button) findViewById(R.id.btnBack);  
        txtketqua=(TextView) findViewById(R.id.txtketqua);  
        Intent callerIntent=getIntent();  
        Bundle packageFromCaller=  
            callerIntent.getBundleExtra("MyPackage");  
        int a=packageFromCaller.getInt("soa");  
        int b=packageFromCaller.getInt("sob");  
        giaipt(a, b);  
        btnBack.setOnClickListener(new View.OnClickListener() {  
            public void onClick(View v) {  
                finish();  
            }  
        });  
    }  
    public void giaipt(int a,int b){..  
    public boolean onCreateOptionsMenu(Menu menu) {..  
}
```

2.3 GỬI/NHẬN KẾT QUẢ TỪ INTENTS

- Phương thức **startActivity(Intent)** được dùng để khởi động 1 activity mới. Activity này sẽ ở đỉnh của stack.
- Nhưng khi muốn nhận 1 kết quả của activity khác gửi đến trước kết thúc quá trình hiển thị của mình: **startActivityForResult(Intent)**
- **Với Android 11 phiên bản alpha 09/2020**. Activity Result APIs thay thế startActivityForResult(Intent). Hướng dẫn thực hiện (4/2021).
 - **registerForActivityResult()** để đăng ký nhận kết quả trả về
 - **ActivityResultContract** và **ActivityResultCallback** và trả về **ActivityResultLauncher** để thao tác với kết quả nhận được

2.3 GỬI/NHẬN KẾT QUẢ TỪ INTENTS (TT)

- Người dùng sẽ dựa vào: `requestCodeID` và `resultCode` để xử lý



2.3 GỬI/NHẬN KẾT QUẢ TỪ INTENTS (TT)

- Khởi động 1 activity ta dùng phương thức:

```
startActivityForResult ( Intent, requestCode )
```



- Và thêm hàm xử lý kết quả trả về

```
onActivityResult ( requestCode, resultCode, Intent )
```

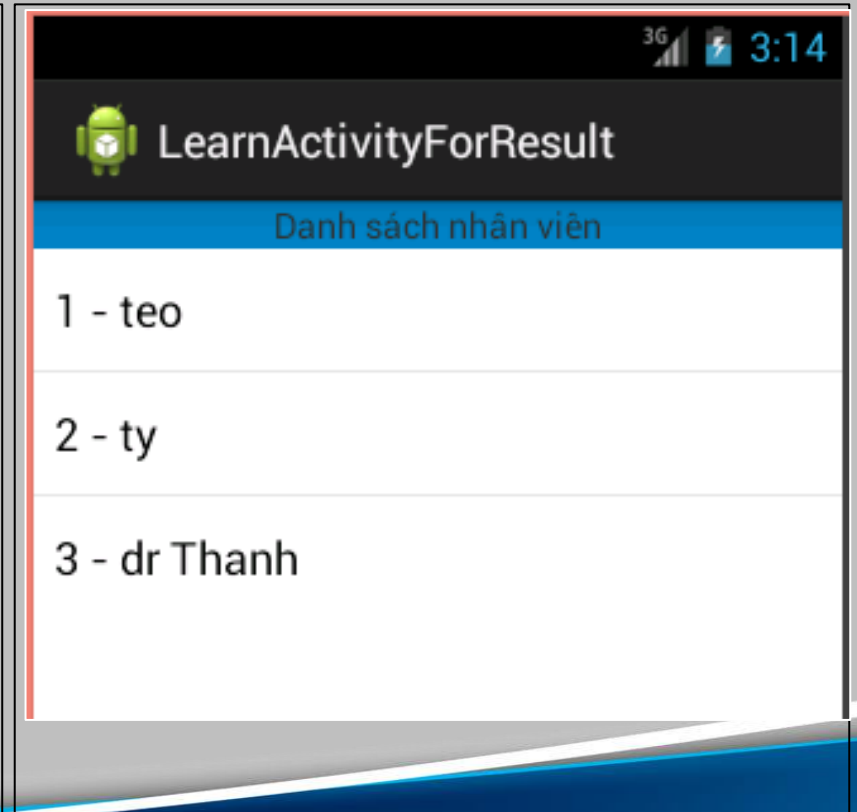
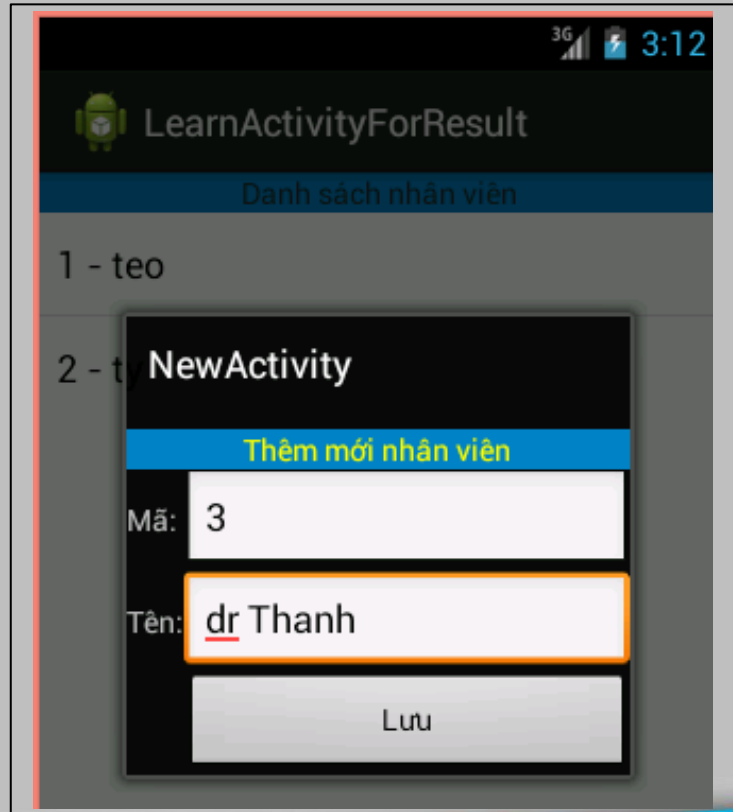
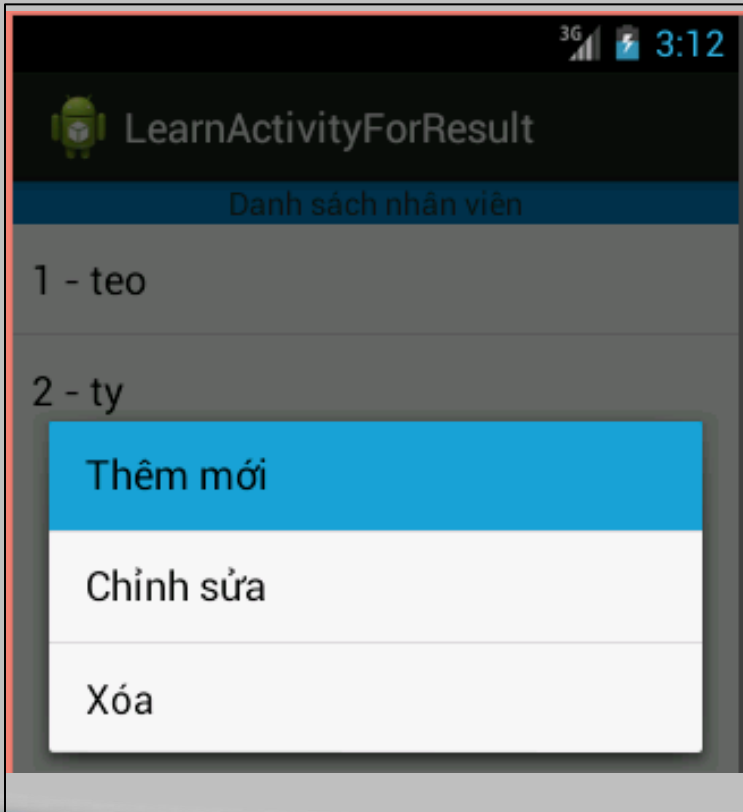


2.3. VÍ DỤ MINH HỌA GỬI/NHẬN KẾT QUẢ TỪ INTENT

1) Load list person on the ListView

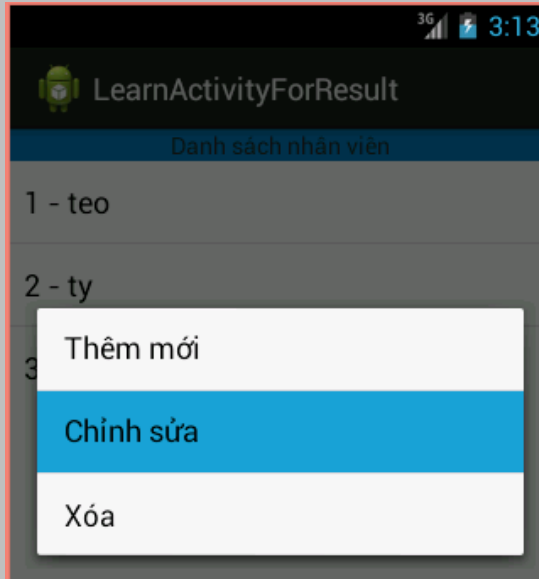
2) Long Item Press on the ListView and choose this menu item: **Thêm mới**

3) Click Lưu Button

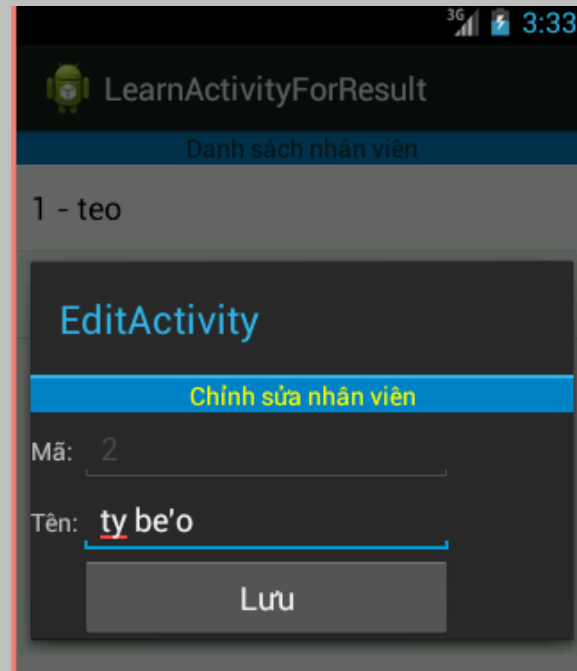


2.3. VÍ DỤ MINH HỌA GỬI/NHẬN KẾT QUẢ TỪ INTENT

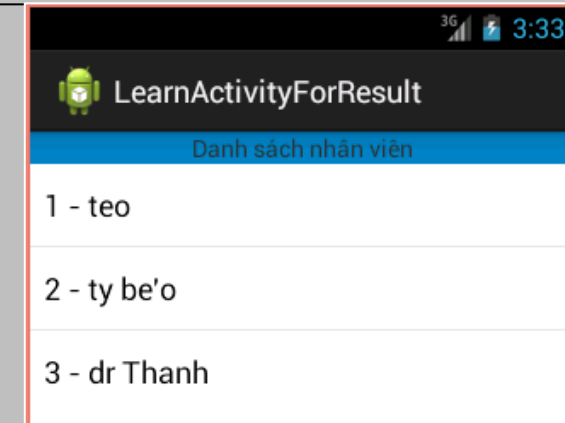
1) Load list person on the ListView



2) Long Item Press on the ListView and choose this menu item: **Chỉnh sửa**

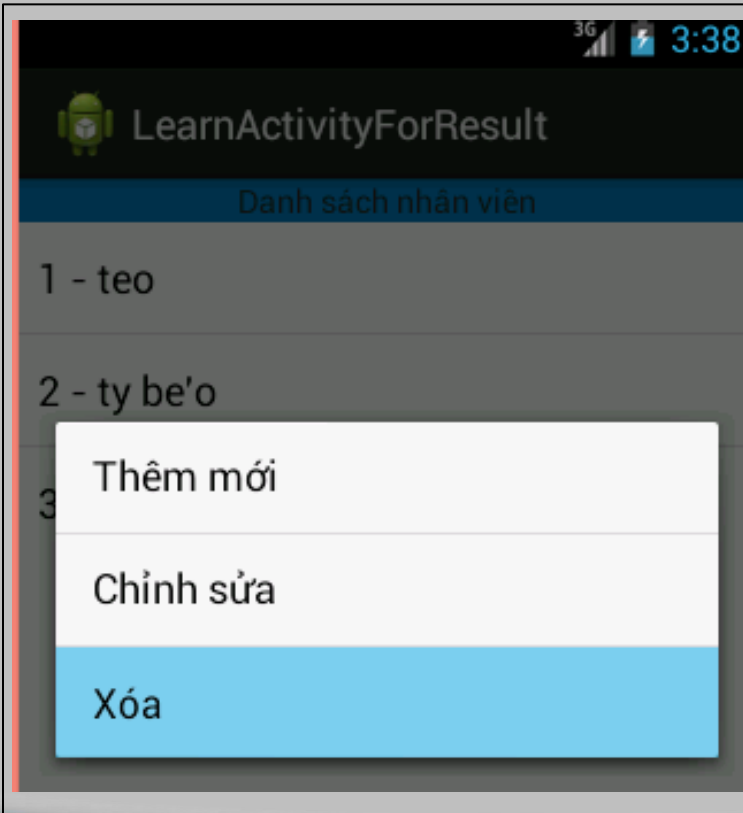


3) Click Lưu Button

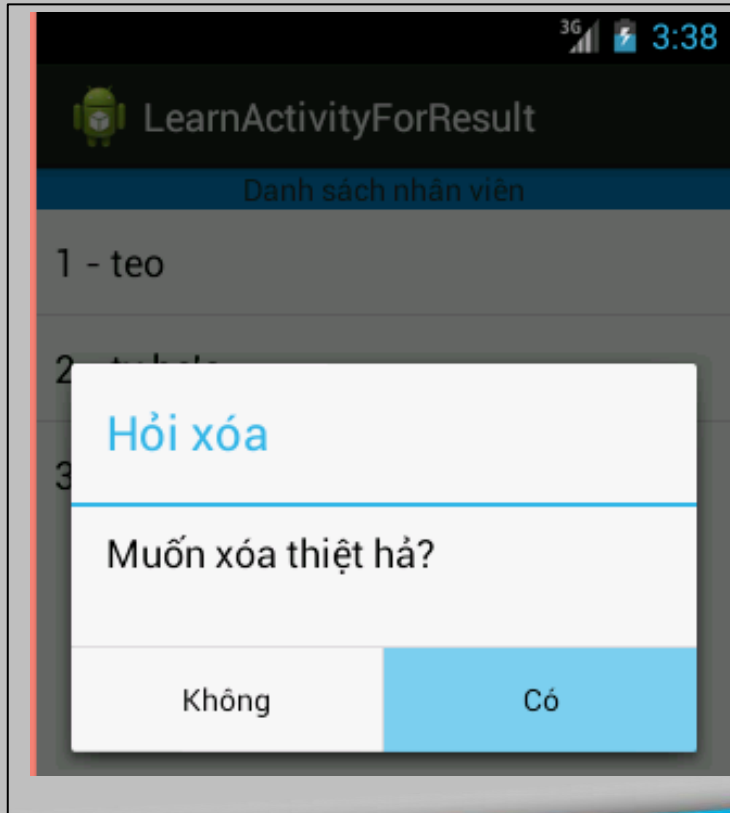


2.3. VÍ DỤ MINH HỌA GỬI/NHẬN KẾT QUẢ TỪ INTENT

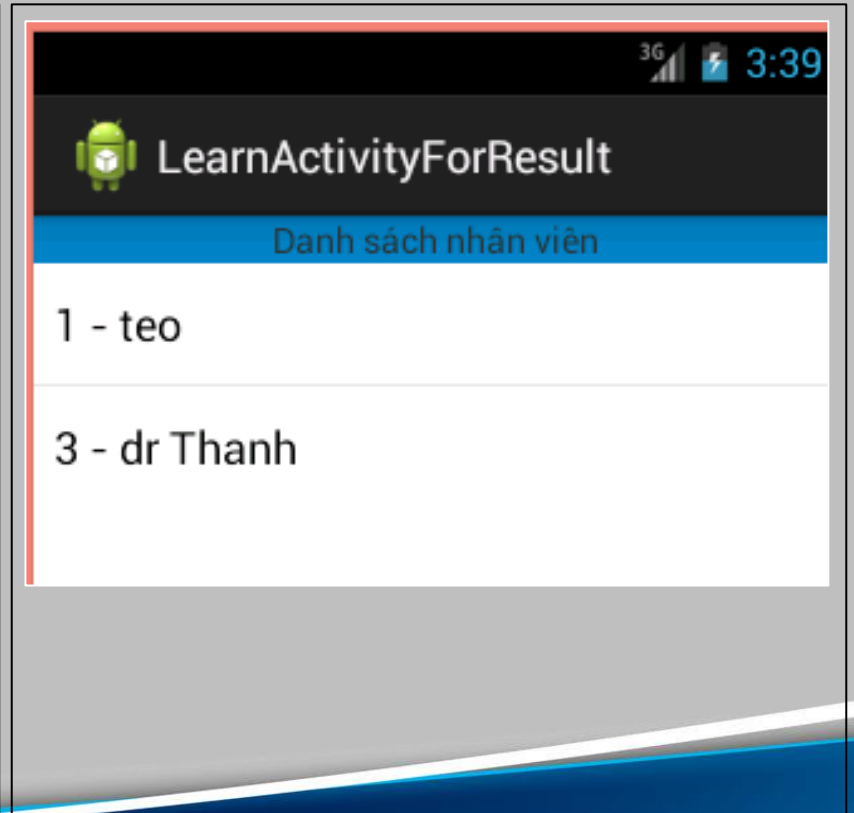
1) Load list person on the ListView



2) Long Item Press on the ListView and choose this menu item: **Xóa**



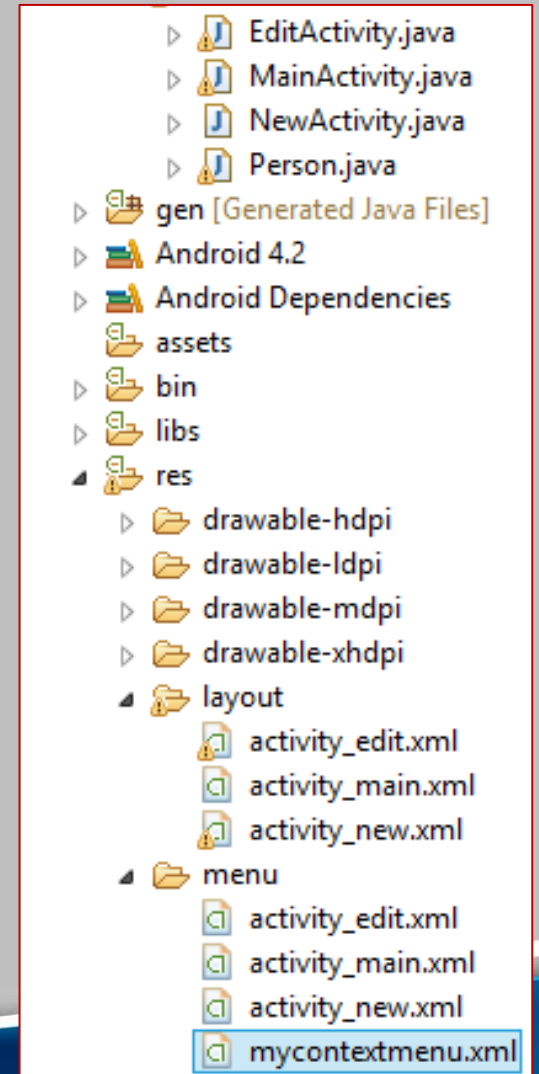
3) Click Lưu Button



2.3. VÍ DỤ MINH HỌA GỬI/NHẬN KẾT QUẢ TỪ INTENT

• Tạo Project

- Prj có 3 activity như hình vẽ
- Lớp person cụ thể hóa giao tiếp Serializable
- Đối tượng lớp này sẽ được sử dụng để gửi thông tin từ MainActivity tới các activity khác và ngược lại
- Tạo file contextmenu.xml



2.3. VÍ DỤ MINH HỌA GỬI/NHẬN KẾT QUẢ TỪ INTENT

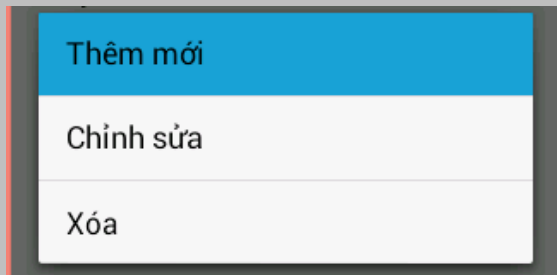
• Person Serializable

- Đối tượng được định nghĩa để gửi qua lại giữa Activity A to another Activity B
- Danh sách đối tượng lưu trữ trong ListView trong MainActivity

```
public class Person implements Serializable {  
    private int id;  
    private String name;  
    public int getId() {  
        return id;  
    }  
    public void setId(int id) {  
        this.id = id;  
    }  
    public String getName() {  
        return name;  
    }  
    public void setName(String name) {  
        this.name = name;  
    }  
    public Person(int id, String name) {  
        public Person() {super();}  
    }  
    public String toString() {  
        return this.id+" - "+this.name;  
    }  
}
```

2.3. VÍ DỤ MINH HỌA GỬI/NHẬN KẾT QUẢ TỪ INTENT

- mycontextmenu XML



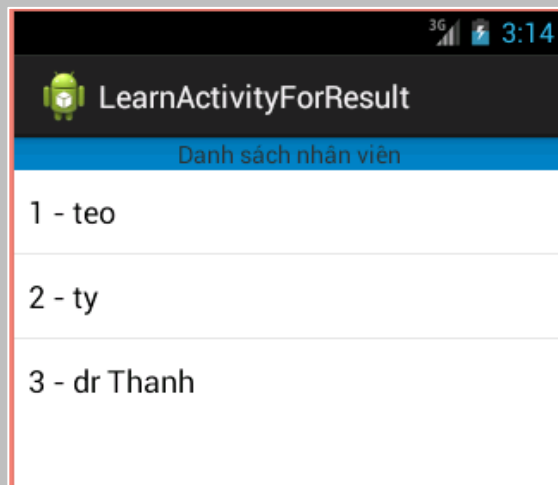
```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android"
    <item
        android:id="@+id/mnnew"
        android:title="Thêm mới">
    </item>
    <item
        android:id="@+id/mnuedit"
        android:title="Chỉnh sửa">
    </item>
    <item
        android:id="@+id/mnudelete"
        android:title="Xóa">
    </item>
</menu>
```

The

2.3. VÍ DỤ MINH HỌA GỬI/NHẬN KẾT QUẢ TỪ INTENT

- MainActivity XML

Person object



```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/LinearLayout1"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity" >
    <TextView
        android:id="@+id/textView1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="#0080C0"
        android:gravity="center"
        android:text="Danh sách nhân viên" />
    <ListView
        android:id="@+id/lvnhanvien"
        android:layout_width="match_parent"
        android:layout_height="wrap_content" >
    </ListView>
</LinearLayout>
```



2.3 GỬI/NHẬN KẾT QUẢ TỪ INTENT (TT)

Kết quả gửi và nhận khi chuyển từ Activity A -

Activity B

```
public class MainActivity extends Activity {  
    public static final int NEW_EMPLOYEE=113;  
    public static final int EDIT_EMPLOYEE=114;  
    public static final int SAVE_NEW_EMPLOYEE=115;  
    public static final int SAVE_EDIT_EMPLOYEE=116;  
    ListView lv;  
    int posselected=-1;  
    ArrayList<Person>list=new ArrayList<Person>();  
    ArrayAdapter<Person>adapter=null;  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
        list.add(new Person(1, "teo"));  
        list.add(new Person(2, "ty"));  
        lv=(ListView) findViewById(R.id.lvnhanvien);  
        adapter=new ArrayAdapter<Person>  
            (this, android.R.layout.simple_list_item_1, list);  
        lv.setAdapter(adapter);  
        lv.setOnItemLongClickListener(  
            new AdapterView.OnItemClickListener() {  
                public boolean onItemClick(  
                    AdapterView<?> arg0, View arg1,int arg2, long arg3) {  
                    posselected=arg2;  
                    return false;}  
            });  
        registerContextMenu(lv);  
    }  
}
```

For Requestcode

For Resultcode

Track the Item
clicked

Make
contextmenu

2.3 GỬI/NHẬN KẾT QUẢ TỪ INTENT (TT)

MainActivity xử lý mã nguồn qua menu

```
public void onCreateContextMenu(ContextMenu menu, View v,
    ContextMenuInfo menuInfo) {
    super.onCreateContextMenu(menu, v, menuInfo);
    getMenuInflater().inflate(R.menu.mycontextmenu, menu);
}
public boolean onOptionsItemSelected(MenuItem item) {
    switch(item.getItemId())
    {
        case R.id.mnunew:
            doStartNew();
            break;
        case R.id.mnuedit:
            doStartEdit();
            break;
        case R.id.mnudelete:
            doDelete();
            break;
    }
    return super.onOptionsItemSelected(item);
}
public void doStartNew()
{
    Intent intentNew=new Intent(this, NewActivity.class);
    startActivityForResult(intentNew, MainActivity.NEW_EMPLOYEE);
}
```

2.3 GỬI/NHẬN KẾT QUẢ TỪ INTENT (TT)

- Nhận kết quả trả về từ Intnetns
- MainActivity

```
public void doStartEdit()
{
    Intent intentEdit=new Intent(this, EditActivity.class);
    Person p=list.get(posselected);
    Bundle bundle=new Bundle();
    bundle.putSerializable("per", p);
    intentEdit.putExtra("DATA", bundle);
    startActivityForResult(intentEdit, MainActivity.EDIT_EMPLOYEE);
}
public void doDelete()
{
    AlertDialog.Builder builder=new Builder(MainActivity.this);
    builder.setTitle("Hỏi xóa");
    builder.setMessage("Muốn xóa thiết hả?");
    builder.setPositiveButton("Có", new DialogInterface.OnClickListener() {
        public void onClick(DialogInterface arg0, int arg1) {
            list.remove(posselected);
            adapter.notifyDataSetChanged();
        }
    });
    builder.setNegativeButton("Không", new DialogInterface.OnClickListener() {
        public void onClick(DialogInterface dialog, int which) {
            dialog.dismiss();
        }
    });
    builder.create().show();
}
```


2.3 GỬI/NHẬN KẾT QUẢ TỪ INTENT (TT)

Main activity

```
protected void onActivityResult(int requestCode, int resultCode,
    Intent data) {
    super.onActivityResult(requestCode, resultCode, data);
    switch(requestCode)
    {
        case MainActivity.NEW_EMPLOYEE:
            if(resultCode==MainActivity.SAVE_NEW_EMPLOYEE) {
                Bundle bundle= data.getBundleExtra("DATA");
                Person p=(Person) bundle.getSerializable("per");
                list.add(p);
                adapter.notifyDataSetChanged();
            }
            break;
        case MainActivity.EDIT_EMPLOYEE:
            if(resultCode==MainActivity.SAVE_EDIT_EMPLOYEE) {
                Bundle bundle= data.getBundleExtra("DATA");
                Person p=(Person) bundle.getSerializable("per");
                list.set(posselected, p);
                adapter.notifyDataSetChanged();
            }
            break;
    }
}

public boolean onCreateOptionsMenu(Menu menu) {
}

} //end MainActivity
```

This method use to get data from sub activity callback

2.3 GỬI/NHẬN KẾT QUẢ TỪ INTENT (TT)

AndroidManifest.xml

```
<application
    android:allowBackup="true"
    android:icon="@mipmap/ic_launcher"
    android:label="@string/app_name"
    android:roundIcon="@mipmap/ic_launcher_round"
    android:supportsRtl="true"
    android:theme="@style/AppTheme">
    <activity android:name=".MainActivity">
        <intent-filter>
            <action android:name="android.intent.action.MAIN" />
            <category android:name="android.intent.category.LAUNCHER" />
        </intent-filter>
    </activity>
    <activity
        android:name=".EditActivity"
        android:label="@string/edit_activity"
        android:theme="@android:style/Theme.Dialog">
    </activity>
    <activity
        android:name=".NewActivity"
        android:label="@string/new_activity"
        android:theme="@android:style/Theme.Holo.Dialog">
    </activity>
</application>
```

2.3 GỬI/NHẬN KẾT QUẢ TỪ INTENT (TT)

New Activity.java

```
public class NewActivity extends Activity {
    EditText txtid,txtname;
    Button btnsave;
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_new);
        txtid=(EditText) findViewById(R.id.txtma);
        txtname=(EditText) findViewById(R.id.txtten);
        btnsave=(Button) findViewById(R.id.btnluu);
        btnsave.setOnClickListener(new View.OnClickListener() {
            public void onClick(View arg0) {
                Intent intent=getIntent();
                Bundle bundle=new Bundle();
                int id=Integer.parseInt(txtid.getText().toString());
                String name=txtname.getText().toString();
                Person p=new Person(id,name);
                bundle.putSerializable("per", p);
                intent.putExtra("DATA", bundle);
                setResult(MainActivity.SAVE_NEW_EMPLOYEE, intent);
                finish();
            }
        });
    }
    public boolean onCreateOptionsMenu(Menu menu) {
    }
```

2.3 GỬI/NHẬN KẾT QUẢ TỪ INTENT (TT)

Edit Activity

```
public class EditActivity extends Activity {
    EditText txtid,txtname;
    Button btnsave;
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_edit);
        final Intent intent=getIntent();
        Bundle bundle=intent.getBundleExtra("DATA");
        final Person p=(Person) bundle.getSerializable("per");
        btnsave =(Button) findViewById(R.id.btnluu);
        txtid=(EditText) findViewById(R.id.txtma);
        txtname=(EditText) findViewById(R.id.txtten);
        txtid.setText(p.getId()+"");
        txtname.setText(p.getName());
        txtid.setEnabled(false);
        btnsave.setOnClickListener(new View.OnClickListener() {
            public void onClick(View arg0) {
                p.setName(txtname.getText()+"");
                setResult(MainActivity.SAVE_EDIT_EMPLOYEE, intent);
                finish();
            }
        });
    }
    public boolean onCreateOptionsMenu(Menu menu) {
    }
}
```



3. IMPLICIT INTENT- INTENT KHÔNG TƯỞNG MINH

- Implicit Intents:

- Là những intent **không** chỉ định rõ một mục tiêu thành phần
- Implicit intent **bao gồm đầy đủ thông tin** cho hệ thống tự xác định các thành phần có sẵn nào là tốt nhất để thực hiện mục đích đó.
- Hệ thống sẽ tìm kiếm tất cả các thành phần đã được đăng ký thông qua bộ lọc thỏa mãn với hành động xác định và tương thích với kiểu dữ liệu.
 - Nếu chỉ **một thành phần được tìm thấy** thì hệ thống Android sẽ khởi động thành phần này trực tiếp.
 - Nếu nhiều thành phần thỏa mãn thì hệ thống sẽ yêu cầu xác nhận từ phía người **dùng lựa chọn** thông qua hộp thoại

3. IMPLICIT INTENT- INTENT KHÔNG TƯỜNG MINH

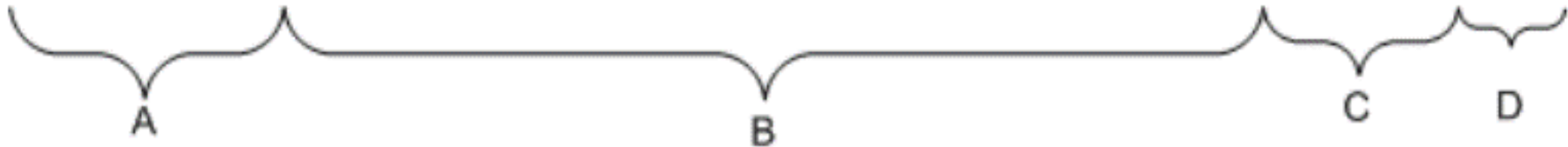
- Khi tạo Implicit Intent, hệ thống cần xác định rõ hành động (action) và dữ liệu (data) tương ứng.
 - **Action**: kiểu chuỗi chứa thông tin hành động sẽ diễn ra. Hệ thống đã định nghĩa sẵn một **số hằng chuẩn trong lớp Intent**.
 - **Data**: dữ liệu chính cần thao tác (số điện thoại, trang web, vvv được mô tả dạng **URI**)
 - **Extra**: Cặp giá trị key-value chứa thông tin bổ sung được truyền tới thành phần điều khiển intent.

<http://developer.android.com/reference/android/content/Intent.html>

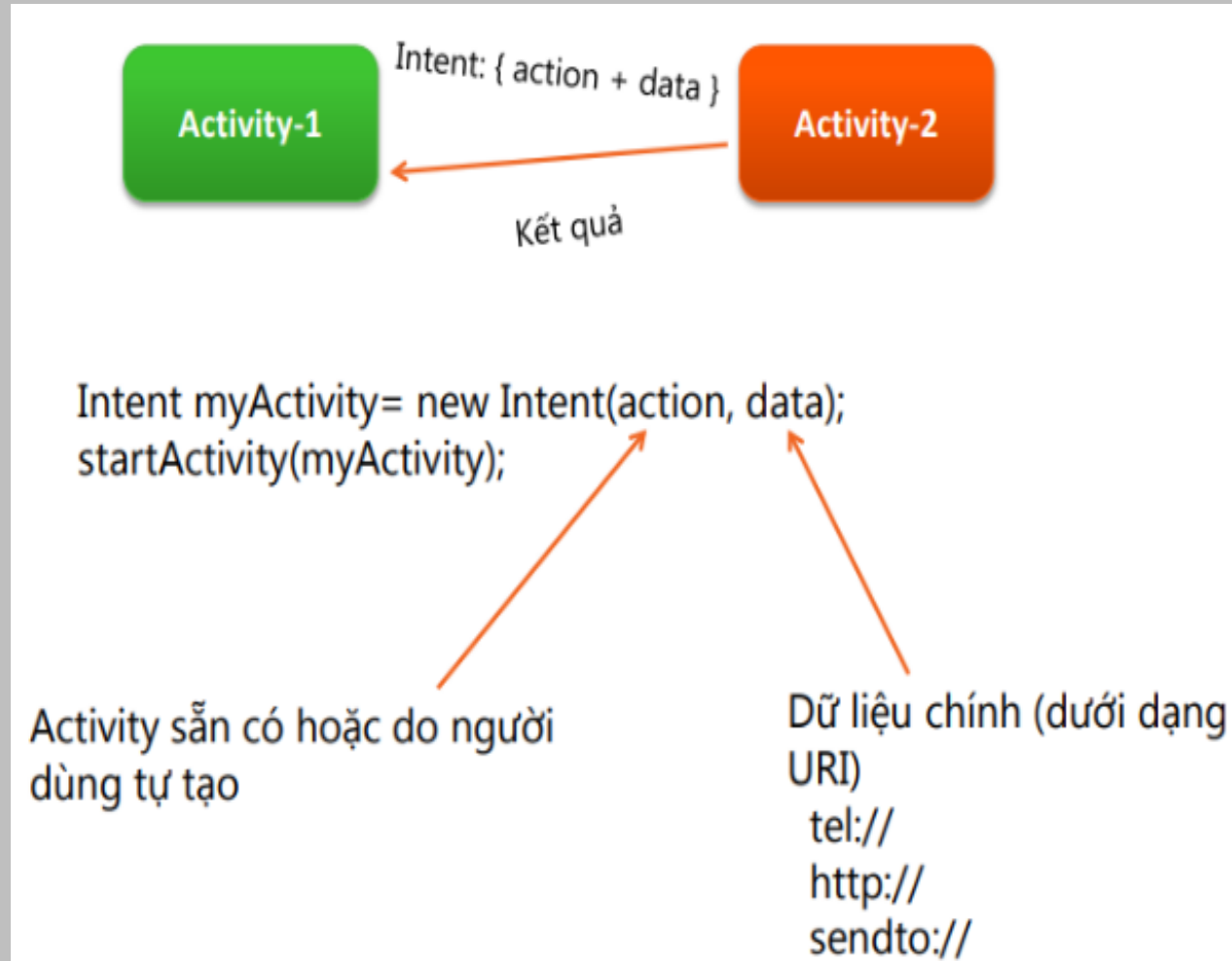
3. IMPLICIT INTENT-(TT) - URI

- URI (Uniform Resource Identifier) là một dãy các ký tự dùng để
 - Mở một trang web trong trình duyệt, bắt đầu với “http:”
 - Chỉ tới các file tài nguyên có sẵn trong máy (ví dụ 1 file video, danh sách contacts,... bắt đầu với “content:”

`content://com.example.transportationprovider/trains/122`



3. (TT)-THAM SỐ CHÍNH CỦA INTENT



3. (TT)- THAM SỐ ACTION

STT	Tên hành động	Mô tả
1	ACTION_ALL_APPS	Liệt kê tất cả các ứng dụng được cài đặt trong thiết bị.
2	ACTION_ANSWER	Xử lý cuộc gọi đến
3	ACTION_ATTACH_DATA	Dùng để báo rằng một phần của dữ liệu được đính kèm vào một vị trí khác.
4	ACTION_BATTERY_CHANGED	Dùng để hiển thị tình trạng sạc và các thông tin khác liên quan đến pin của thiết bị.
5	ACTION_CALL	Dùng để thực hiện cuộc gọi đến một số xác định được chỉ ra trong dữ liệu
6	ACTION_CALL_BUTTON	Người dùng bấm vào nút call để kích hoạt danh bạ để thực hiện cuộc gọi
7	ACTION_CAMERA_BUTTON	Dùng để báo hiệu nút Camera đã được bấm.
8	ACTION_CHOOSER	Hiển thị activity lựa chọn để người sử dụng lựa chọn cái mình cần rồi mới xử lý tiếp.
9	ACTION_CONFIGURATION_CHANGED	Báo cấu hình hiện tại của thiết bị thay đổi.



3. (TT)- THAM SỐ ACTION/DATA

- Khi sử dụng Implicit Intent, hành động khác nhau được kết hợp theo cặp với các loại đặc tả dữ liệu.

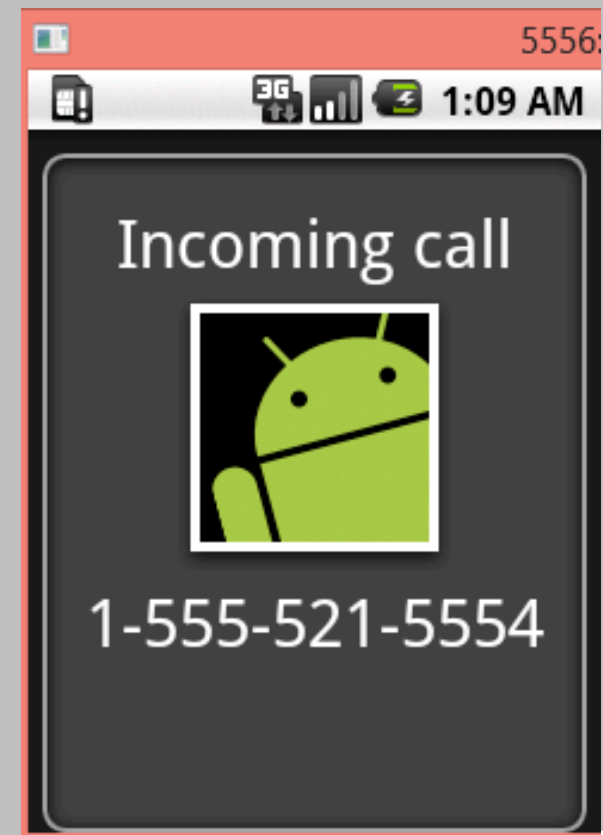
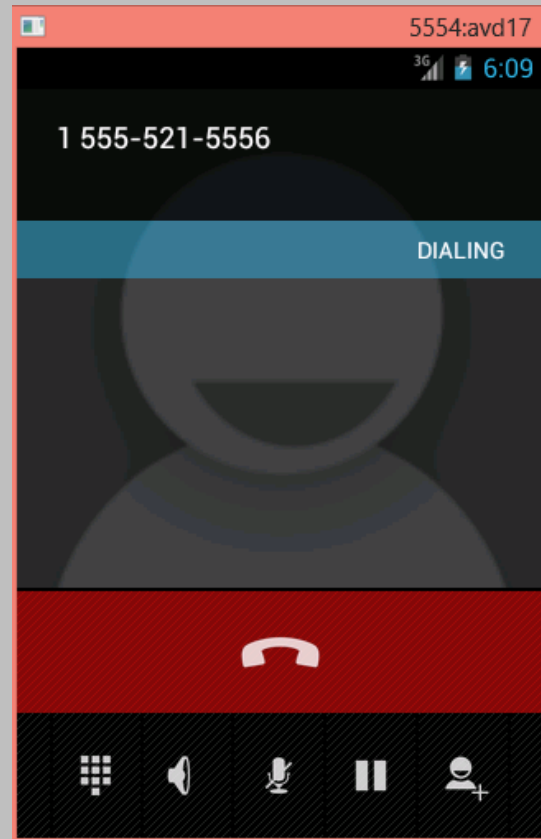
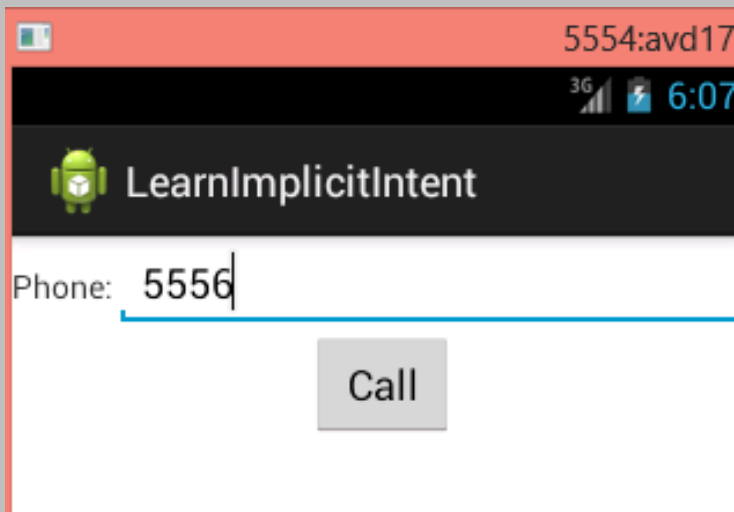
- Nếu trường action là ACTION_EDIT thì trường dữ liệu sẽ chứa định dạng tài nguyên thống nhất (URI) của tài liệu được hiển thị để chỉnh sửa.
- Nếu ACTION_CALL thì trường dữ liệu sẽ là tel: URI là số điện thoại để gọi.

Một số cặp hành động và dữ liệu mẫu được liệt kê trong bảng 4.2 sau:

STT	Hành động/Dữ liệu	Mô tả
1	ACTION_VIEW content://contacts/people/1	Hiển thị thông tin về người trong sổ địa chỉ có định danh là “1”
2	ACTION_DIAL content://contacts/people/1	Hiển thị danh bạ điện thoại với thông tin của người có định danh được điền vào
3	ACTION_VIEW tel:123	Hiển thị danh bạ điện thoại với số điện thoại cụ thể được điền vào
4	ACTION_DIAL tel:123	Quay số đến số xác định được điền vào
5	ACTION_EDIT content://contacts/people/1	Sửa thông tin của người trong danh bạ có định danh là “1”
6	ACTION_VIEW content://contacts/people/	Hiển thị danh sách tất cả người trong danh bạ và người sử dụng có thể duyệt qua.



3.(TT) DEMO IMPLICIT INTENT

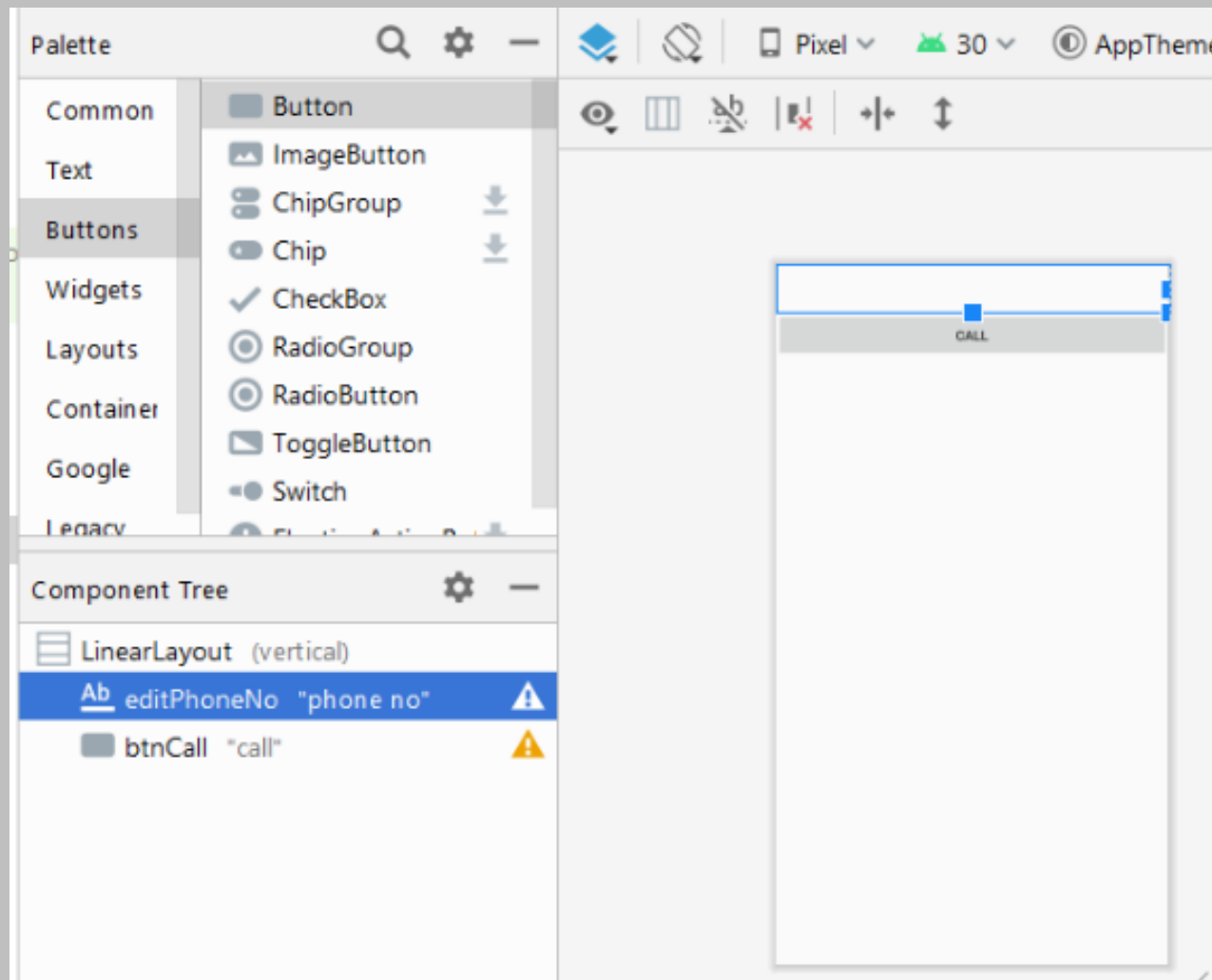


3.(TT) DEMO IMPLICIT INTENT

AndroidManifest.xml

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

3.(TT) DEMO IMPLICIT INTENT

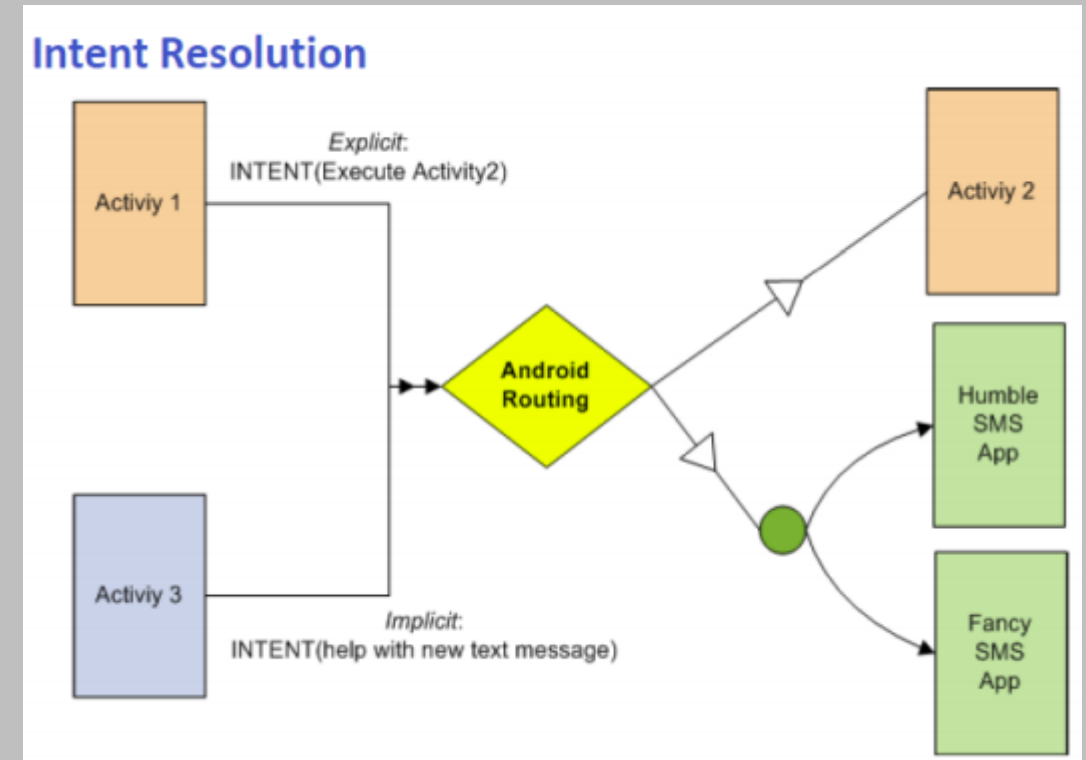


3.(TT) DEMO IMPLICIT INTENT

```
public class MainActivity extends AppCompatActivity {  
    EditText editPhoneNo;  
    Button btnCall;  
    String phoneNo="1234";  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
        getWidGet();  
        btnCall.setOnClickListener(new View.OnClickListener() {  
            @Override  
            public void onClick(View view) {  
                phoneNo=editPhoneNo.getText().toString();  
                Intent activityCall=new Intent( Intent.ACTION_CALL, Uri.parse("tel:"+phoneNo));  
                startActivity(activityCall);  
            }  
        });  
    }  
    private void getWidGet() {  
        editPhoneNo=findViewById(R.id.editPhoneNo);  
        btnCall=findViewById(R.id.btnCall);  
    }  
}
```

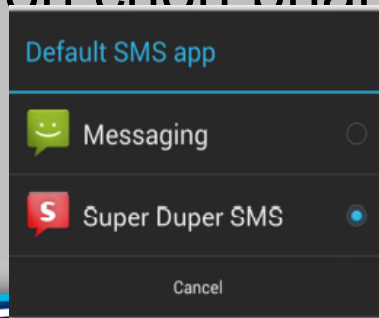
INTENT FILTER

- Để thông báo cho hệ thống implicit intent nào được sử dụng, activity, service và broadcast receiver có thể có một hoặc nhiều bộ lọc intent (intent filter)



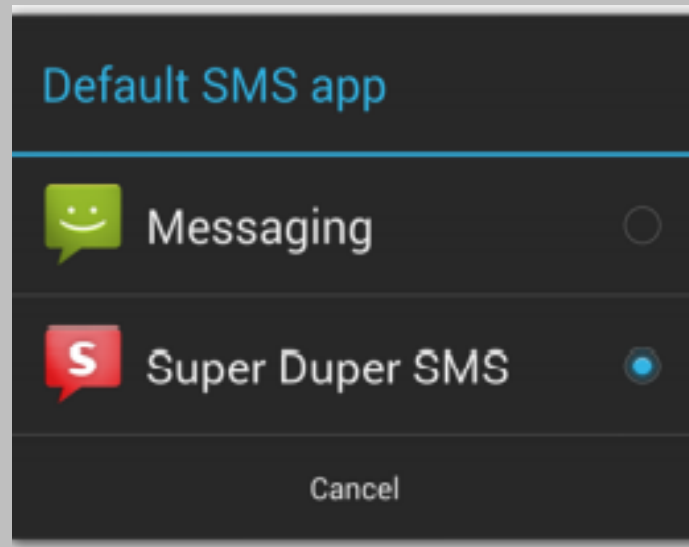
VÍ DỤ VỀ INTENT FILTER

- Giả sử người dùng cài ứng dụng SMS “Super Duper SMS” để thay thế cho ứng dụng Messaging mặc định của máy
- Với sự ra đời của Implicit Intent, Android sẽ “hỏi” người dùng câu hỏi:
 - “Bạn có một tin nhắn mới. Bạn có hai ứng dụng nhắn tin ”Super Duper SMS” và ứng dụng Messaging mặc định.
 - Bạn chọn phần mềm nào, bạn có muốn chọn phần mềm này là phần mềm mặc định khi đọc tin nhắn không?



VÍ DỤ VỀ INTENT FILTER

- Một activity được gọi khi thỏa mãn các điều kiện sau:
 - Hỗ trợ action cho trước
 - Hỗ trợ type of data đã được miêu tả (nếu cung cấp)
 - Hỗ trợ tất cả category được nêu trong Intent

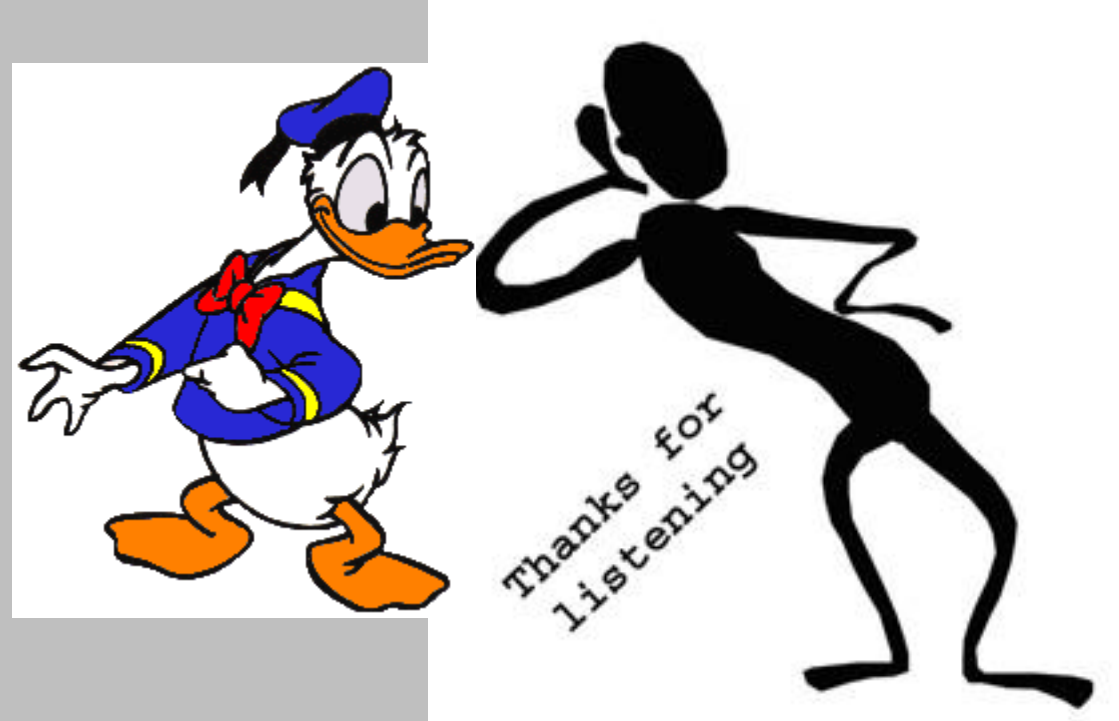


VÍ DỤ INTENT FILTER

- Phần tử `<data>` như minh họa dưới đây sẽ thông báo cho Android biết rằng component có thể nhận dữ liệu video từ trên mạng và có thể hiển thị `<data android:scheme="http" android:type="video/*">`
- Android sẽ chọn một trình duyệt khi người dùng click vào một link web
- Đầu tiên, android sẽ hiển thị dữ liệu (nếu link trên trang web). Nếu không thể hiển thị dữ liệu, nó sẽ đẩy implicit intent cùng với scheme và data type và thử khởi tạo một activity đáp ứng được công việc này
- Nếu không có Activity đáp ứng được, trình download manager sẽ tải dữ liệu về

REFERENCE

- <http://devlup.com/mobile/android/what-is-an-android-activity/3634/>
- Beginning Android Application Development, *Wei-Meng Lee*
- Android Built-in Layout Class, *M.Sc. Bui Tan Loc*, University of Sciences- Ho Chi Minh City, Viet Nam



END