

INTENT

1 Nội dung:

- Làm quen và sử dụng Intent chuyển đổi giữa các màn hình.
- Sử dụng AlertDialog.Builder cho phép hỏi đáp với người dùng.
- Truyền dữ liệu từ Activity con về Activity cha.
- Sử dụng control ImageView, ListView.
- Hiểu được cách sử dụng String-Array, ArrayAdapter.

2 Sử dụng ImageView

2.1 Giới thiệu

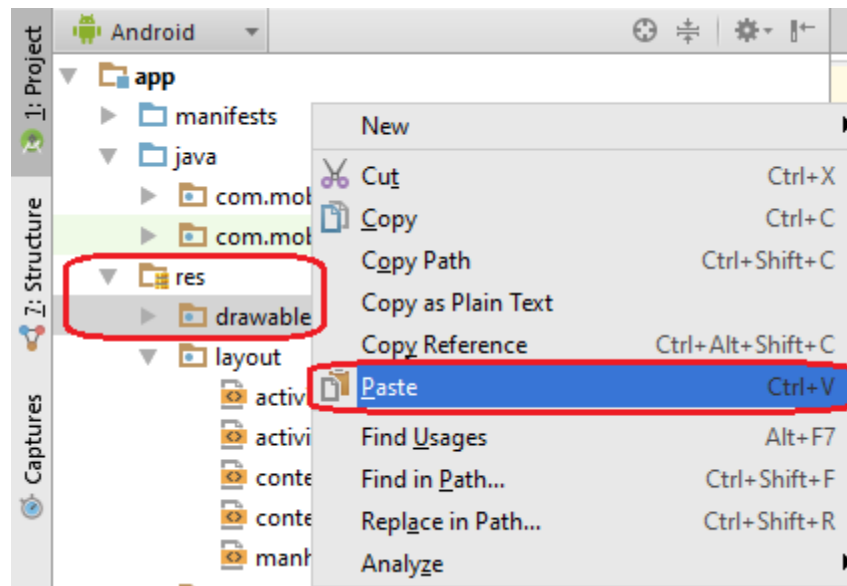
ImageView là thành phần giúp hiển thị một hình ảnh lên giao diện người dùng. Để đặt hình ảnh cho nó ta sử dụng một trong các phương thức sau:

- setImageBitmap(bitmap): sử dụng hiện một hình ảnh Bitmap
- setImageResource(id): lấy một hình ảnh từ trong thư mục drawable và hiển thị nó
- setImageLevel(Level): sử dụng Level
- setImageMatrix(Matrix): sử dụng ma trận ảnh
- setImageURI(URI): sử dụng thông qua một URI

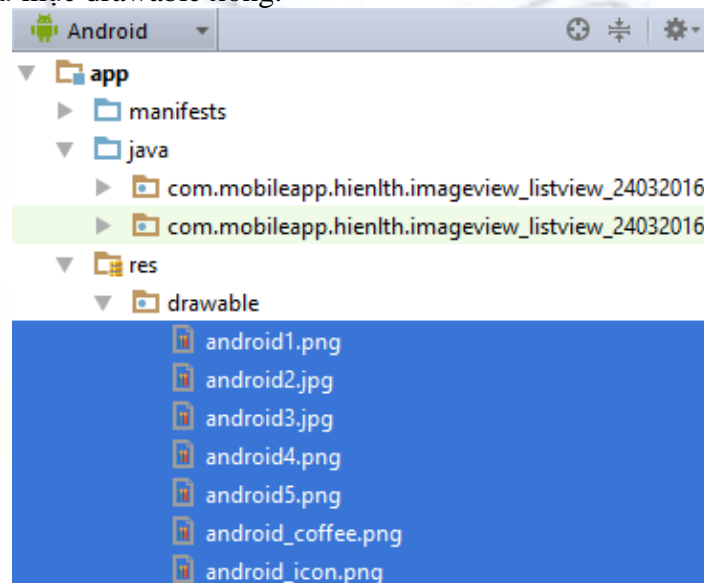
2.2 Hiển thị hình lên ImageView

Đầu tiên chọn 1 hoặc nhiều file ảnh và Chép các file này vào 1 trong thư mục con có tên **drawable** của thư mục **res**.



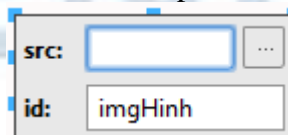


Sau khi paste vào thư mục drawable xong:

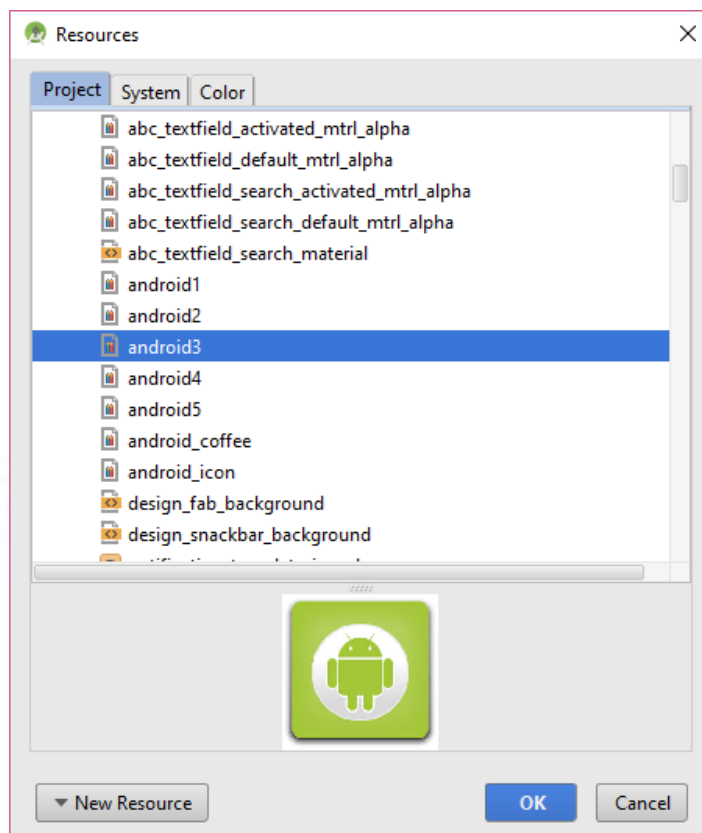


2.2.1 Cách 1: Hiển thị ảnh bằng ImageView trực tiếp từ giao diện

Double click lên control ImageView sẽ hiển thị hộp thoại sau:



Chọn ... trên phần src, sau đó chọn Project, mục drawable và chọn file hình tương ứng.



2.2.2 Cách 2: Hiển thị ảnh bằng ImageView tạo bằng file java

Tình huống: Click chọn button để hiển thị hình.

```
btnLoadHinh = (Button)findViewById(R.id.btnLoad);
imgHinh = (ImageView)findViewById(R.id.imgHinh);

btnLoadHinh.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        imgHinh.setImageResource(R.drawable.android2);
    }
});
```

↑ file hình trong drawable

2.2.3 Tải hình từ Internet

```
URL url = new
URL("http://hcmup.edu.vn/images/stories/site_18/mainsite/daotao.png");
Bitmap bm = BitmapFactory.decodeStream(url.openConnection().getInputStream());
imgHinh.setImageBitmap(bm);
```

```
try {
    URL url = new URL("http://hcmup.edu.vn/images/stories/site_18/mainsite/daotao.png");
    Bitmap bm;
    try {
        bm = BitmapFactory.decodeStream(url.openConnection().getInputStream());
        imgHinh.setImageBitmap(bm);
    }
    catch (IOException ex) {
        ex.printStackTrace();
    }
} catch (MalformedURLException e) {
    e.printStackTrace();
}
```

```
HttpURLConnection httpConn = (HttpURLConnection) url.openConnection();
```

```
httpConn.connect();
```

```
int resCode = httpConn.getResponseCode();
```

```
if (resCode == HttpURLConnection.HTTP_OK) {
    InputStream in = httpConn.getInputStream();
    Bitmap bitmap = BitmapFactory.decodeStream(in);
    this.imageView.setImageBitmap(bitmap);
}
```

2.3 Game bầu cua

Cho hình sau:



a0.png



a1.png



a2.png



a3.png



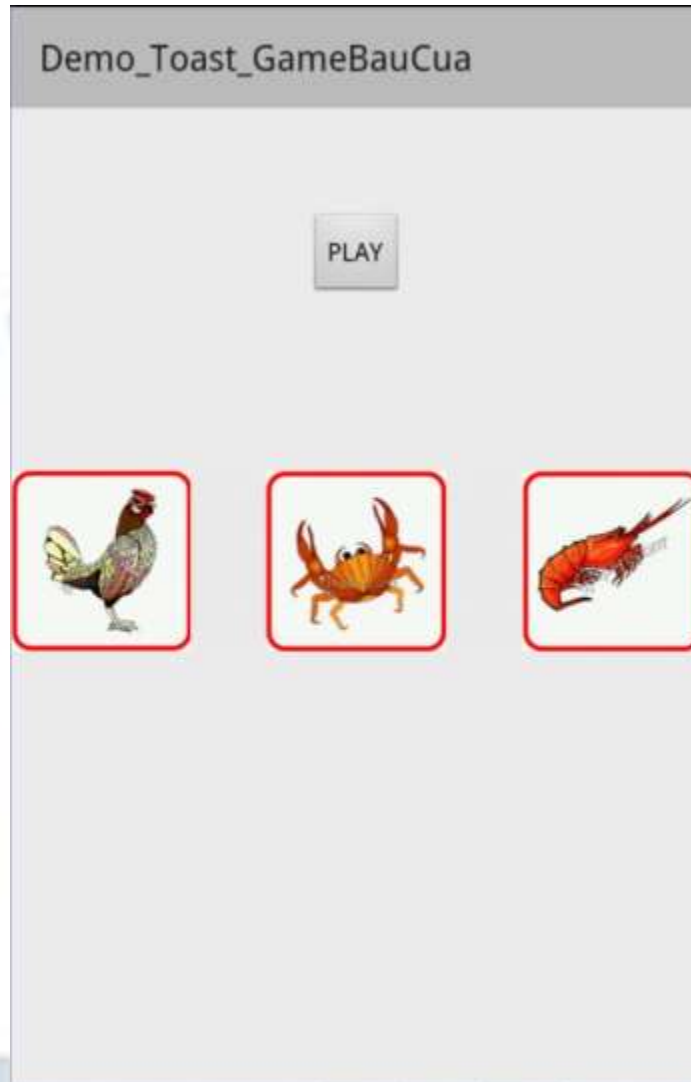
a4.png



a5.png

Hãy viết code khi click nút PLAY, lấy random 3 hình bất kì trong 6 hình trên. Hiện thị trong 1 Toast:

CENTER_HORIZONTAL, CENTER_VERTICAL, LENGTH



Hướng dẫn:

Lấy random từ 0 đến 5:

```
Random r = new Random();  
int r1 = r.nextInt(6);
```

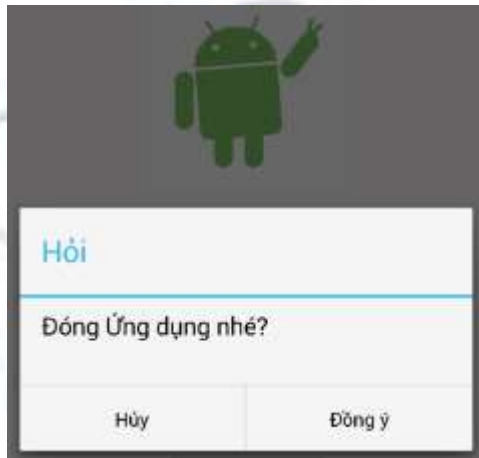
Hiện thị file hình qua ImageView:

```
Resources res = getResources();  
String mDrawableName = "ten_file_hinh"; //a1.png
```

```
int resID = res.getIdentifier(mDrawableName, "drawable", getPackageName());  
Drawable drawable = res.getDrawable(resID);  
icon.setImageDrawable(drawable);
```

3 Alert Dialog

Alert Dialog là hộp thoại tương tác người dùng tùy theo ngữ cảnh người dùng.



Hướng dẫn:

```
//Khai báo hộp thoại thông báo  
AlertDialog.Builder dlg = new AlertDialog.Builder(MainActivity.this);  
//Set tiêu đề và nội dung  
dlg.setTitle("Hỏi");  
dlg.setMessage("Đóng Ứng dụng nhé?");  
//set button Positive - OK  
dlg.setPositiveButton("Đồng ý", new DialogInterface.OnClickListener() {  
    @Override  
    public void onClick(DialogInterface dialog, int which) {  
        finish();  
    }  
});  
//set button Negative - Cancel  
dlg.setNegativeButton("Hủy", new DialogInterface.OnClickListener() {  
    @Override  
    public void onClick(DialogInterface dialog, int which) {  
        dialog.cancel();  
    }  
});
```

//tạo và hiển thị
dlg.create().show();

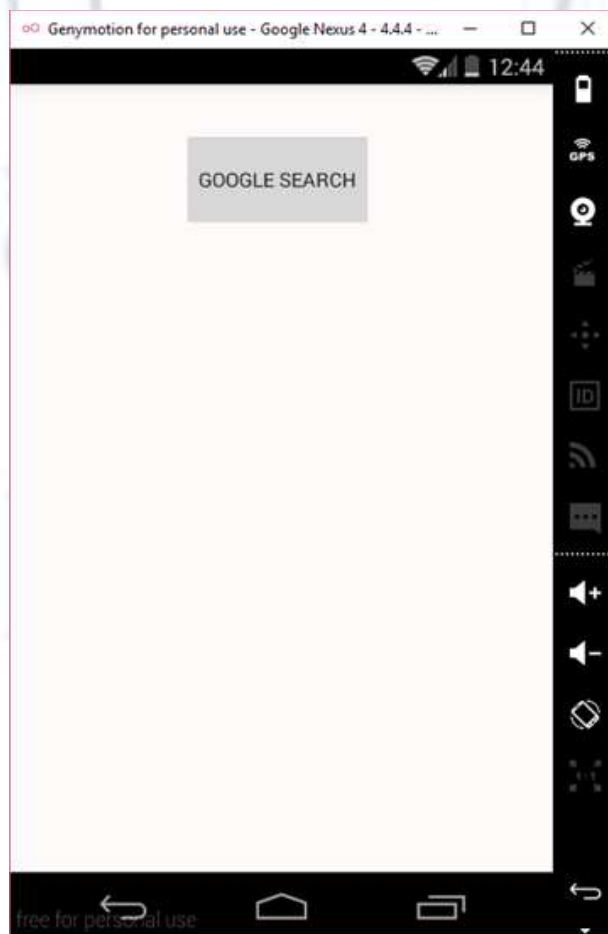
Ý nghĩa của các hàm :

- **setTitle** : thiết lập tiêu đề cho Dialog.
- **setMessage**: Thiết lập nội dung cho Dialog.
- **setIcon** : để thiết lập Icon.
- **setPositiveButton**, **setNegativeButton** thiết lập hiển thị Nút chọn cho Dialog.
Chú ý là ở đối số thứ 2 của các hàm này sẽ là **DialogInterface.OnClickListener**
- **create()** để tạo Dialog
- **show()** để hiển thị Dialog.

4 BÀI TẬP INTENT

4.1 Bài tập 1:

Tạo ứng dụng minh họa các intent-filter

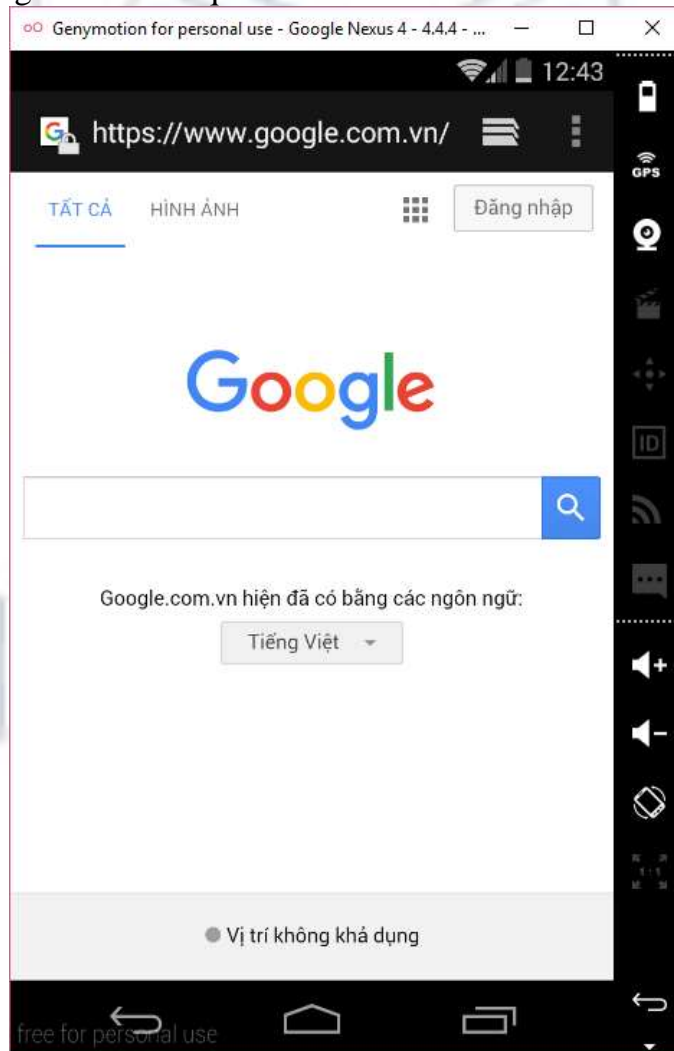


Hướng dẫn:

- Tạo giao diện ứng dụng có 1 button như hình
- Viết xử lý cho button

```
btnGoogle.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        Intent mh = new Intent(Intent.ACTION_VIEW, Uri.parse("http://google.com.vn"));  
        startActivity(mh);  
    }  
});
```

- Giải thích code: ở đây ta đang tạo ra một intent có khả năng View 1 trang web. Constructor của Intent có 2 tham số tương ứng.
- Chạy ứng dụng và xem kết quả



- Thêm Button khác gọi intent dạng **Intent.ACTION_CALL**

```
btnCall.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        Intent mh = new Intent(Intent.ACTION_CALL, Uri.parse("tel: (+84) 888333777"));  
        startActivity(mh);  
    }  
});
```

Missing permissions required by intent Intent.ACTION_CALL: android.permission.CALL_PHONE [more...](#) (Ctrl+F1)

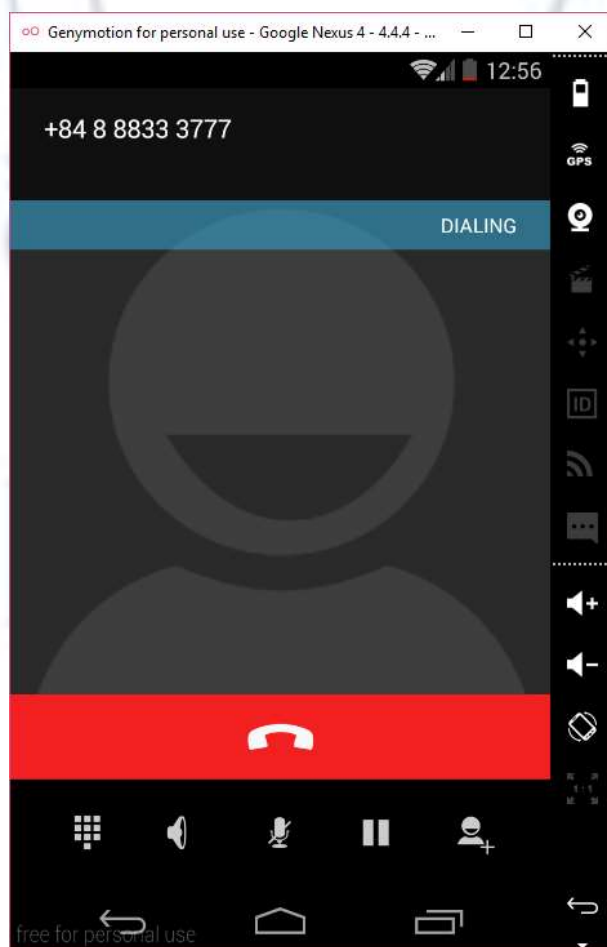
- Thêm permission vào **AndroidManifest.xml**

(xem thêm cấu trúc file tại:

<http://developer.android.com/intl/vi/guide/topics/manifest/manifest-intro.html>)

```
<uses-permission android:name="android.permission.CALL_PHONE">  
</uses-permission>
```

- Chạy ứng dụng



4.2 Bài tập 2:

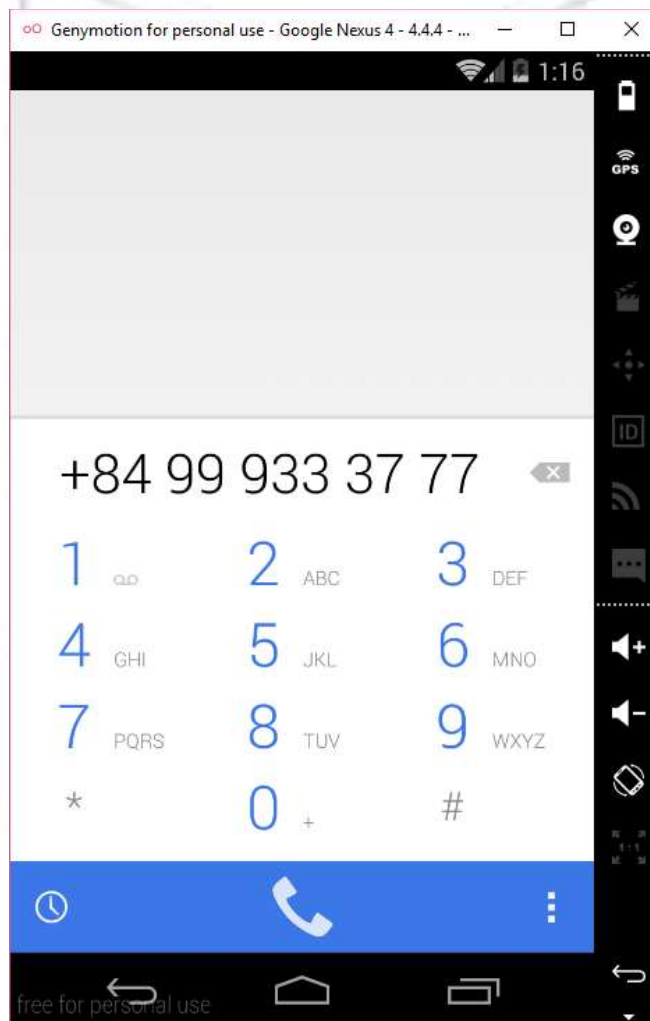
Tiếp tục ứng dụng của bài tập 1, thay thế bằng các intent filter khác và xem kết quả

Thêm các permission vào **AndroidManifest.xml**

```
<uses-permission android:name="android.permission.CALL_PHONE"></uses-permission>
<uses-permission android:name="android.permission.CAMERA"></uses-permission>
<uses-permission android:name="android.permission.READ_CONTACTS"></uses-permission>
<uses-permission android:name="android.permission.INTERNET"></uses-permission>
```

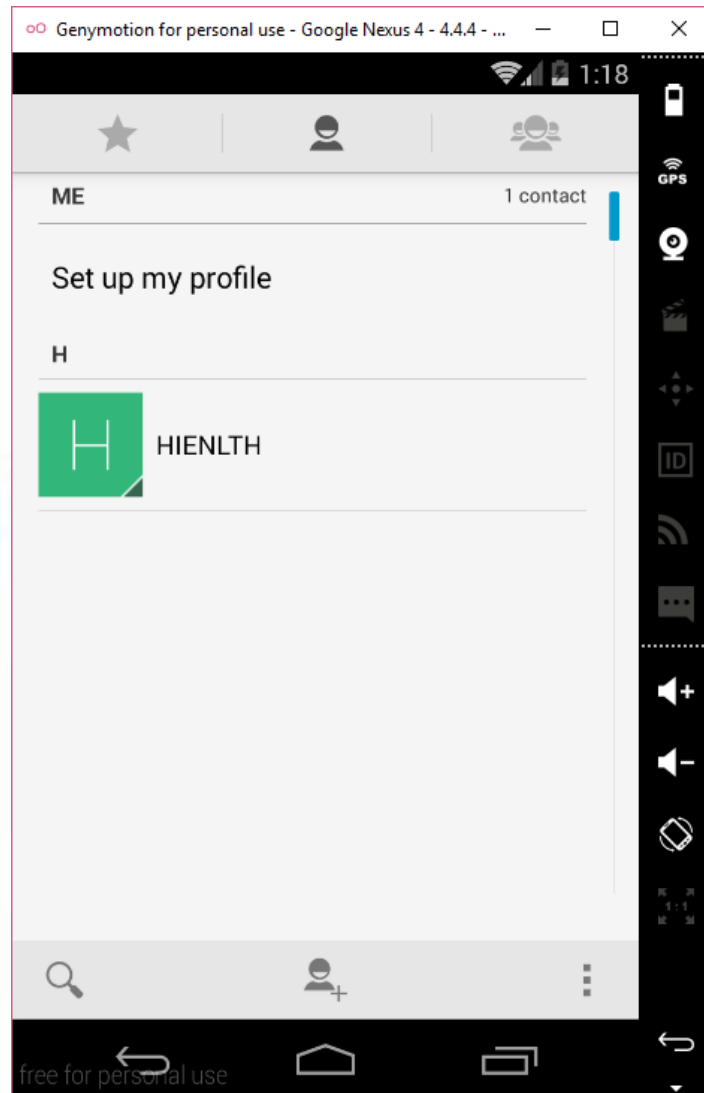
4.2.1 Intent.ACTION_DIAL

```
Intent mh = new Intent(Intent.ACTION_DIAL, Uri.parse("tel:(+84)999333777"));
```



4.2.2 Intent.ACTION_VIEW

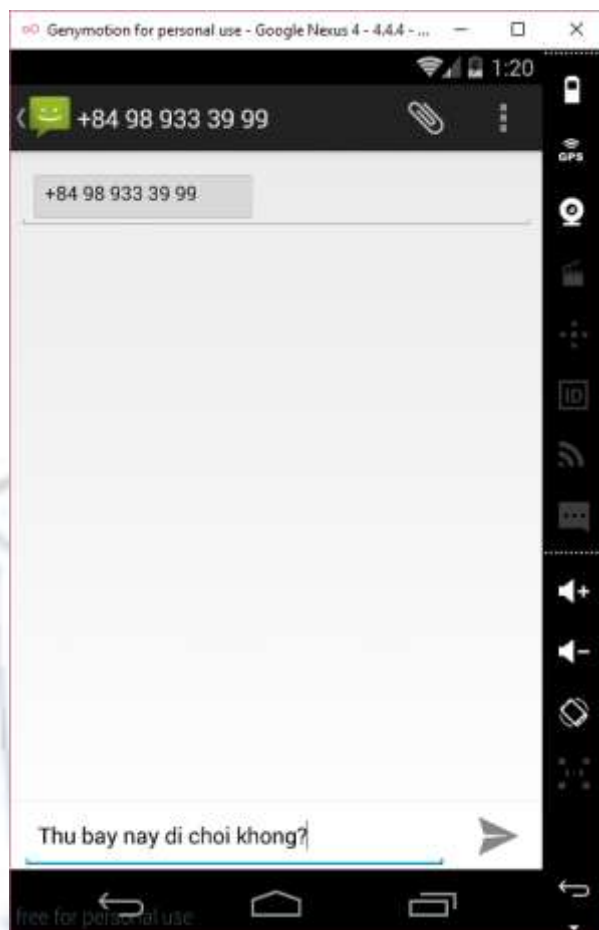
```
Intent intent = new Intent(Intent.ACTION_VIEW, Uri.parse("content://contacts/people/"));
```



4.2.3 Intent.ACTION_SENDTO

```
intent = new Intent( Intent.ACTION_SENDTO, Uri.parse("sms:(+84)989333999"));
```

```
intent.putExtra("sms_body", "Thu bay nay di choi khong?");
```



4.2.4 Xem ảnh

```
Intent myIntent = new Intent();
myIntent.setType("image/pictures/*");
myIntent.setAction(Intent.ACTION_GET_CONTENT);
startActivity(myIntent);
```

4.2.5 Nghe nhạc

```
Intent myActivity2 = new Intent("android.intent.action.MUSIC_PLAYER");
startActivity(myActivity2);
```

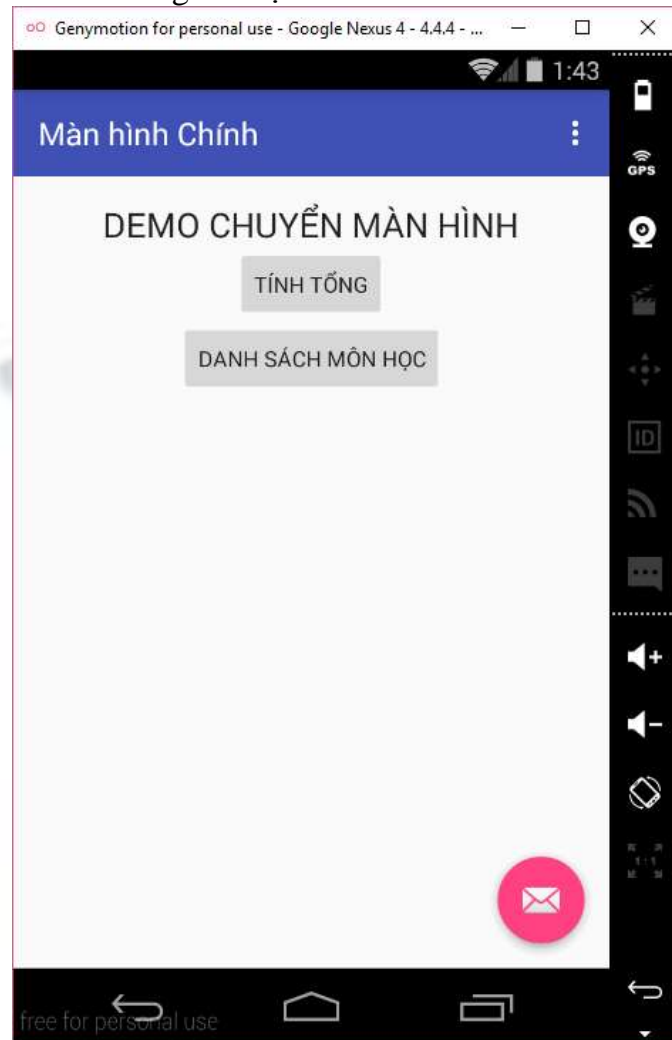
4.2.6 Hiển thị địa điểm Đại học Sư phạm TpHCM:

```
String url = "https://goo.gl/maps/uFyzc2qigiM2";
Intent intent = new Intent(Intent.ACTION_VIEW, Uri.parse(url));
startActivity(intent);
```

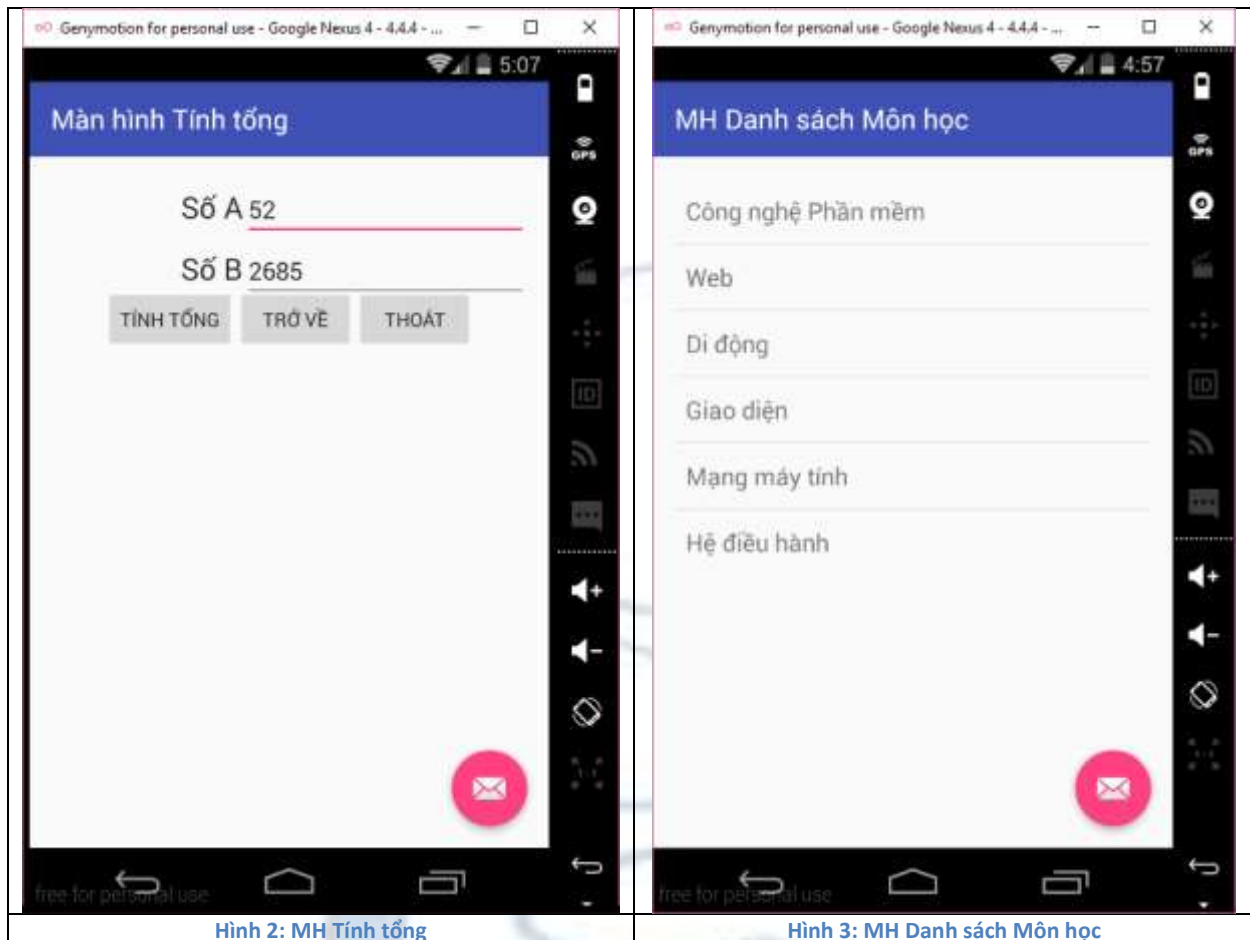
- Sinh viên tự tìm hiểu và tham khảo thêm các intent khác

5 Chuyển đổi Activity

Thiết kế màn hình chính với giao diện như hình bên dưới:



Hình 1: MH chính



Hình 2: MH Tính tổng

Hình 3: MH Danh sách Môn học

- Bấm nút “Tính tổng” ở màn hình chính sẽ mở màn hình như hình 2.

Trong đó khi khởi tạo màn hình truyền tham số là 2 số ngẫu nhiên sang cho màn hình Tính tổng làm giá trị mặc định.

- Khi bấm vào button “Danh sách Môn học” ở màn hình chính, hiển thị màn hình Danh sách Môn học có thiết kế như hình 3.

5.1 Chuyển đổi Activity



```
Intent manhinh_1 = new Intent(MainActivity.this, SecondActivity.class);
startActivity(manhinh_1);
```

Nếu có truyền tham số:

```
Intent mh2 = new Intent(MainActivity.this, ManHinh2.class);
//Truyền tham số
mh2.putExtra("AA", A);
mh2.putExtra("BB", B);
startActivity(mh2);
```

5.2 Nhận giá trị tham số từ Activity khác chuyển qua

```
//lấy lại giá trị truyền từ màn hình
Bundle bd = getIntent().getExtras();
if(bd != null)
{
    int A = bd.getInt("AA");
    int B = bd.getInt("BB");
    txtA.setText(Integer.toString(A));
    txtB.setText(Integer.toString(B));
}
```

6 ListView

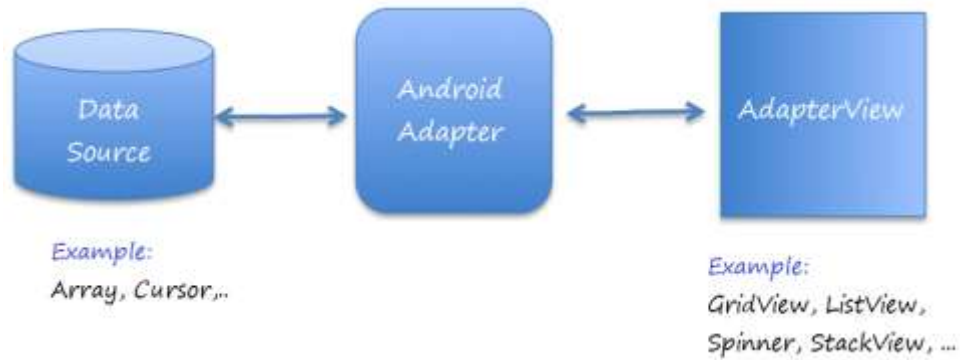
6.1 Giới thiệu

ListView là một view group, hiển thị các thành phần (elements) theo một danh sách, có thể cuộn được theo chiều thẳng đứng. ListView là một view quan trọng, nó được sử dụng rộng rãi trong các ứng dụng Android.

Một ListView được tạo từ một danh sách các ListItem. ListItem là một dòng (row) riêng lẻ trong listview nơi mà dữ liệu sẽ được hiển thị. Bất kỳ dữ liệu nào trong listview chỉ được hiển thị thông qua listItem.

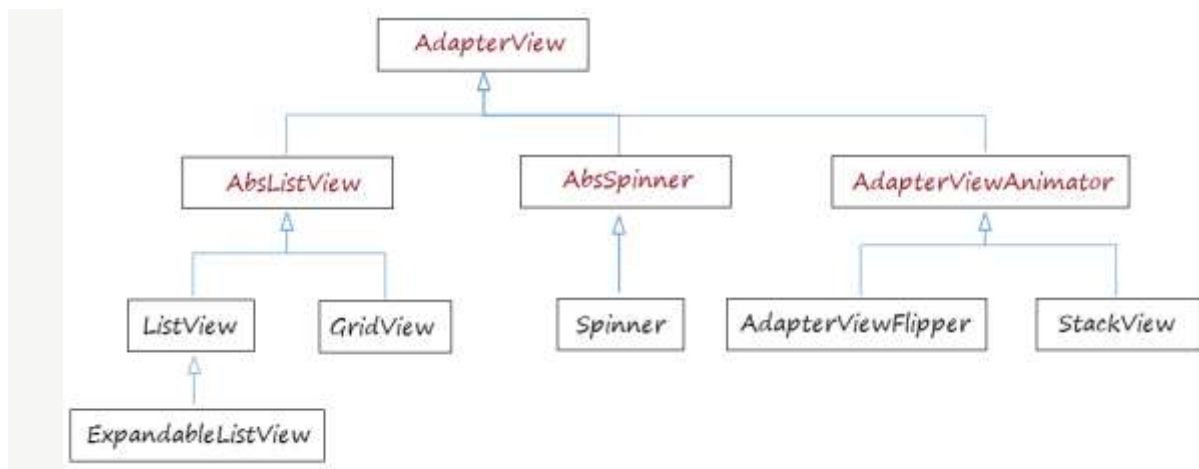
Android Adapter (bộ tiếp nối) là một cầu nối giữa các View (ví dụ như **ListView**) và các dữ liệu cơ bản cho View đó. Một Adapter quản lý dữ liệu và ghép nối với các dòng riêng lẻ (ListItems) của view.

Chúng ta ràng buộc các Adapter với Android ListView thông qua phương thức *setAdapter*.

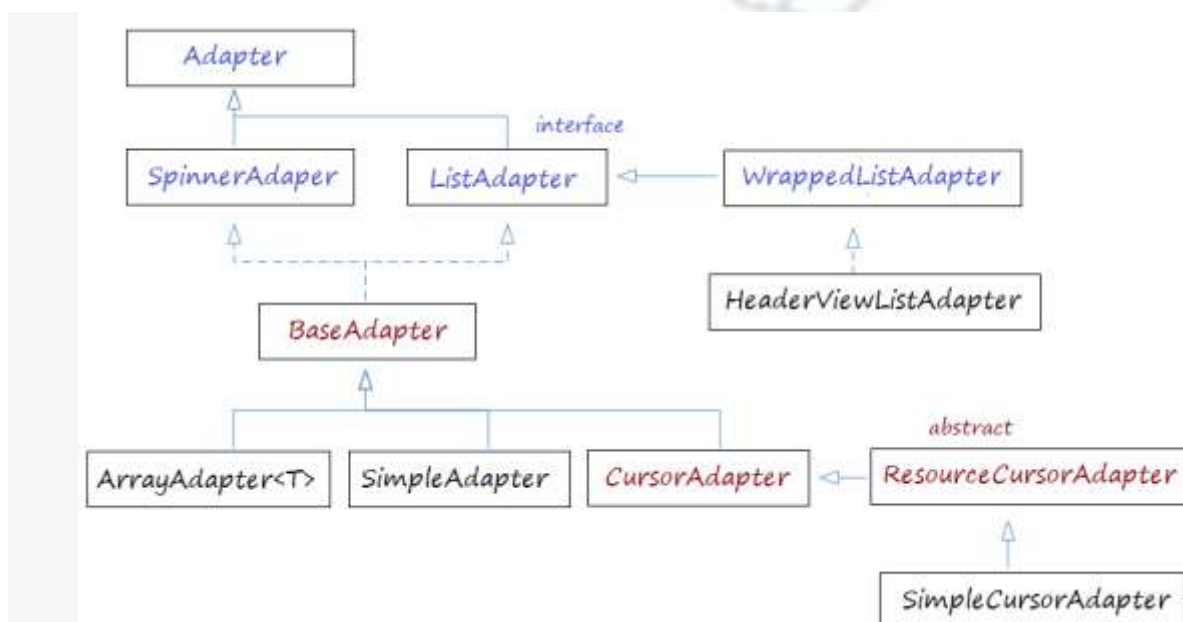


6.2 AdapterView

Có nhiều **View** cần tới **Android Adapter** để quản lý dữ liệu hiển thị, các View này là con của class **AdapterView**, bạn có thể xem ở hình minh họa dưới đây:



6.3 Android Adapter



6.4 Các Layout có sẵn làm việc với ArrayAdapter

Android xây dựng sẵn một số Layout (của ListItem) có thể làm việc với ArrayAdapter.

android.R.layout.simple_list_item_1

Đây là layout đơn giản của ListItem, được tạo bởi duy nhất một **TextView**

android.R.layout.simple_list_item_checked và

android.R.layout.simple_list_item_multiple_choice

2 Layout trên là layout đơn giản để tạo ra một **ListView** với **ListItem** có một checkbox.

(sẽ có bài tập về ListItem có checkbox và ListItem tùy biến sau)

6.5 Thêm dữ liệu cho ListView (Màn hình 3)

//Xử lý thêm dữ liệu

```
lvMonHoc = (ListView)findViewById(R.id.listViewMonHoc);  
final String[] dsmon = new String[]{"Công nghệ Phần mềm", "Web", "Đi  
động", "Giao diện", "Mạng máy tính", "Hệ điều hành"};  
ArrayAdapter adap = new ArrayAdapter(  
    MHDSMonHoc.this,  
    android.R.layout.simple_list_item_1,//kiểu hiển thị  
    dsmon  
);  
lvMonHoc.setAdapter(adap);
```

6.6 Thêm sự kiện click lên Item của ListView

```
lvMonHoc.setOnClickListener(new AdapterView.OnItemClickListener() {  
    @Override  
    public void onItemClick(AdapterView<?> parent, View view, int position, long id) {  
        String kq = "Position :" + position + ", value =" + dsmon[position];  
        Toast.makeText(MHDSMonHoc.this, kq, Toast.LENGTH_LONG).show();  
    }  
});
```

----Kết thúc Lab----