








LẬP TRÌNH ANDROID CƠ BẢN

BÀI 2: CÁC THÀNH PHẦN CƠ BẢN CỦA ỨNG DỤNG ANDROID



- ⊙ Kết thúc bài học này bạn có khả năng
 - ⊙ Hiểu về các thành phần cơ bản của ứng dụng Android
 - ⊙ Hiểu về Intent, Context
 - ⊙ Hiểu về Activity và vòng đời của Activity



Phần I: Các thành phần cơ bản của Android

-  Activity
-  Service
-  ContentProvider
-  BroadcastReceiver
-  Intent, Context

Phần II: Activity và vòng đời của Activity

-  Activity - Task, stack
-  Vòng đời của Activity





BÀI 2: CÁC THÀNH PHẦN CƠ BẢN CỦA ỨNG DỤNG ANDROID

PHẦN 1: CÁC THÀNH PHẦN CƠ BẢN

- Các thành phần:
 - Activity
 - Service
 - ContentProvider
 - BroadcastReceiver
 - Intent

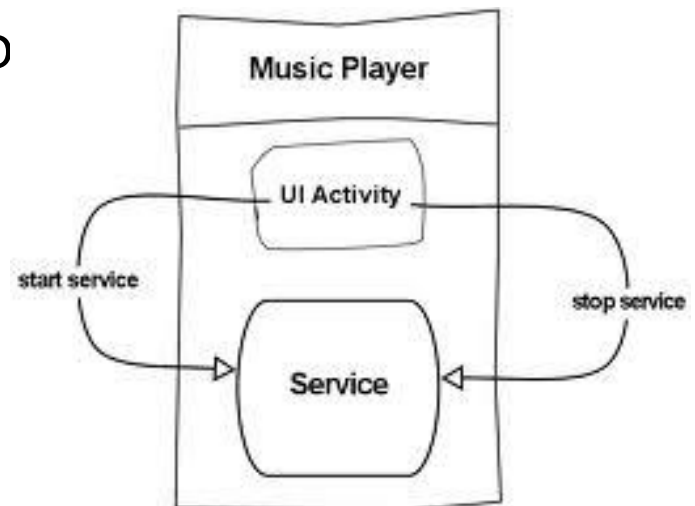


Activity

- Activity là tầng trình diễn của ứng dụng. Giao diện người dùng của ứng dụng được xây dựng dựa trên một hoặc nhiều Activity. Mỗi Activity đều là một lớp kế thừa từ lớp Activity. Activity sử dụng Fragment và View để bố trí và hiển thị thông tin và tương tác với hoạt động của người dùng
- Activity bao gồm 1 Class java và 1 file .xml dùng để thiết kế giao diện người dùng.

Service

- Service là một thành phần chạy ngầm trên hệ điều hành để thực hiện các hoạt động dài hạn mà không cần phải tương tác với người sử dụng và nó hoạt động ngay cả khi ứng dụng bị phá hủy. Một khi được gọi, dịch vụ này có thể chạy ở chế độ nền vô thời hạn, thậm chí cả khi thành phần đã khởi động nó bị phá hủy
- Có hai loại Service:
 - Làm một số công việc nền theo
 - Truyền thông
 - Ví dụ: trình nghe nhạc

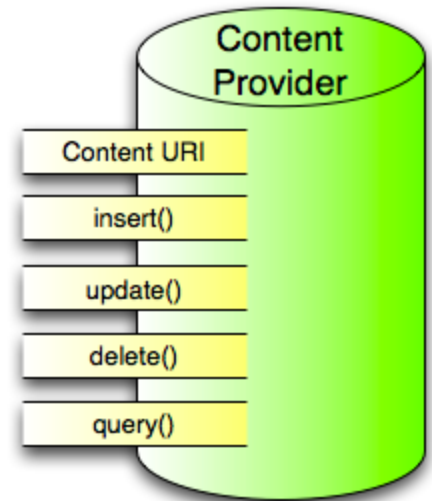


ContentProvider

Truy cập thông qua truy vấn content://URI

Có truy vấn, thêm mới, xóa,...

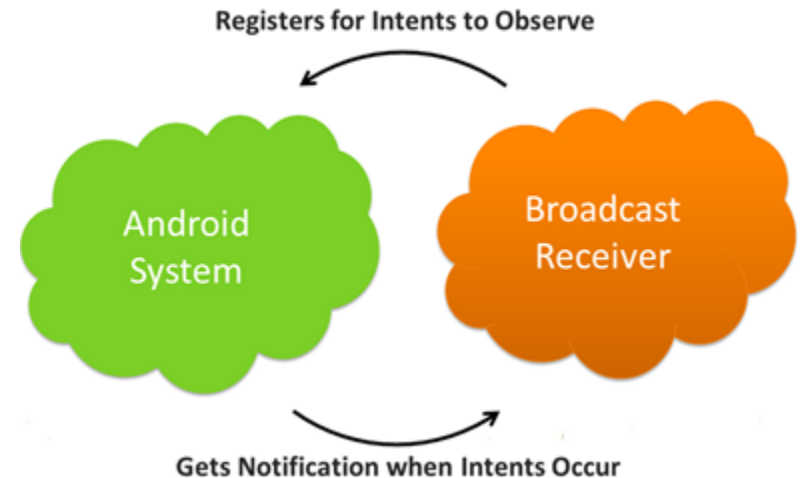
Tìm hiểu sâu hơn các bài sau



```
ContentResolver cr=  
Context.getContentResolver();  
cr.query(content://android.provider.  
Contacts.Phones.CONTACT_URI,...)
```


BroadcastReceiver

- Đánh thức bởi broadcast hệ thống
- Rất đơn giản – chỉ là onReceive handler
 - Nhận context và Intent miêu tả broadcast



Có 2 loại Intent

- Action, Data = Implicit
- Action, Data, Component = Explicit



Thành phần khác của Intent

- Category: miêu tả loại thành phần điều khiển Intent
 - CATEGORY_LAUNCHER: Activity xuất hiện ở launcher
 - CATEGORY_PREFERENCE
- EXTRA: cặp giá trị key-value chứa thông tin bổ sung
 - ACTION_HEADSET_PLUG
- Flags: hướng dẫn hệ thống cách khởi tạo Activity
 - FLAG_ACTIVITY_NO_ANIMATION

Truy cập các thành phần ứng dụng

- Activity và service được khởi tạo như thế nào?
- Java:
 - Viết class để thực hiện một số công việc
 - Giống trong Android
 - Viết phương thức Main. Trong phương thức Main gọi hàm khởi tạo của class và chạy các phương thức
 - Không giống trong Android
 - Phụ thuộc vào kiểu đối tượng, Android sẽ gọi hàm tạo và quản lý vòng đời của đối tượng

- Lớp Context cung cấp truy cập tới chức năng và dịch vụ của hệ thống
- Activity và Service kế thừa Context, do đó có thể gọi các phương thức trong Context trực tiếp
- BroadcastReceiver có chứa tham số Context trong tham số đầu vào ở các hàm quản lý sự kiện
- ContentProvider gọi hàm getContext để lấy đối tượng Context

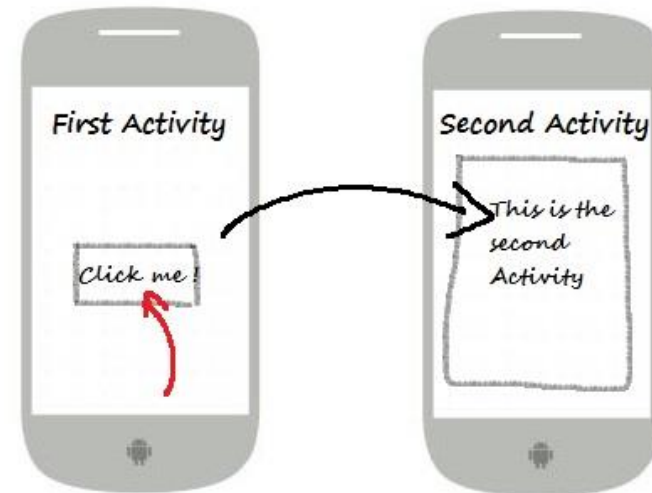




BÀI 2: CÁC THÀNH PHẦN CƠ BẢN CỦA ỨNG DỤNG ANDROID

PHẦN 1: ACTIVITY VÀ VÒNG ĐỜI CỦA

- Khởi tạo Activity bằng cách gọi `startActivity(Intent)`
- Subactivity: Là activity được gọi bởi activity khác.
- Gọi Subactivity sử dụng phương thức `startActivityForResult`
 - Truyền Intent và integer code trong tham số đầu vào
 - Khi subactivity kết thúc, trả lại mã code
 - `startActivityForResult` là phương thức không đồng bộ



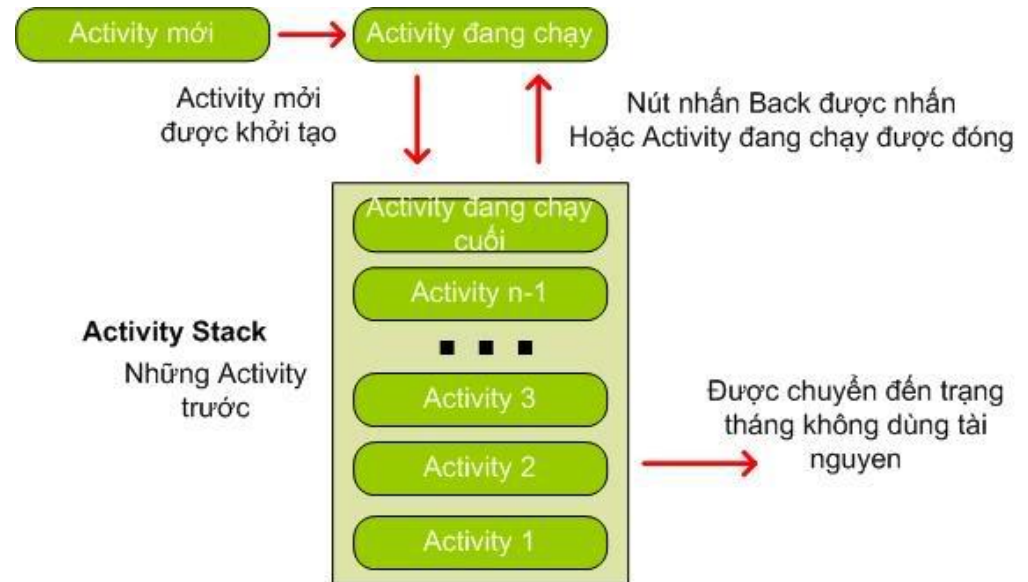
Task

- Android nhóm các activity trong một chương trình vào một công việc chung (hàng đợi các activity liên quan đến nhau)
- Người dùng nhấn nút HOME và khởi tạo một chương trình mới
 - Chuyển task hiện tại sang chế độ nền
 - Bắt đầu task mới, đặt activity mặc định của ứng dụng mới ở đầu Stack
- Nếu ứng dụng được quay lại, task cũ (stack cũ) sẽ được khôi phục



Stack

- Các activity cấu tạo nên Stack
 - Activity mới sẽ xuất hiện ở đầu Stack
 - Thông thường, khi nhấn nút back sẽ quay lại activity trước đó





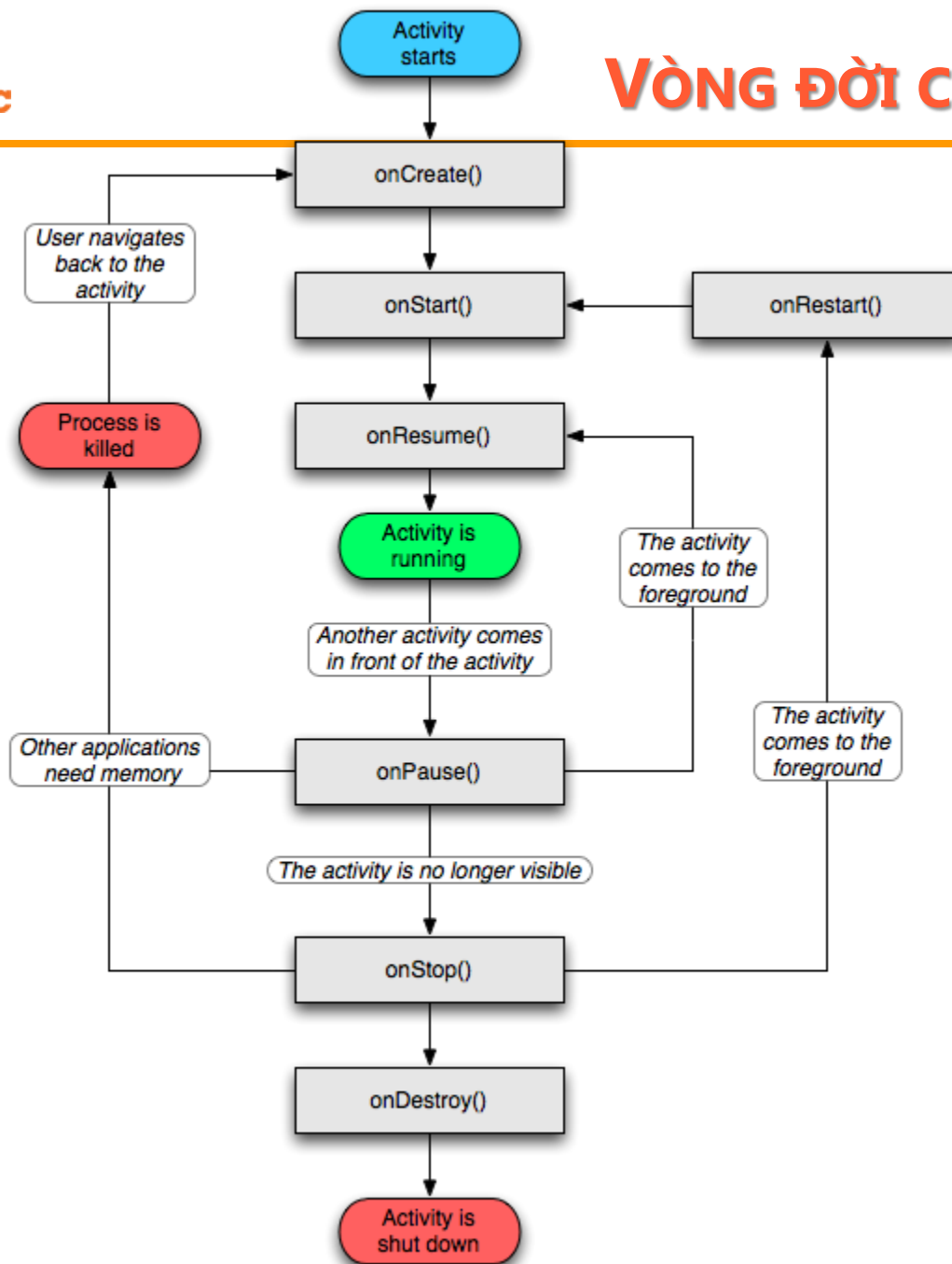
DEMO

- Sử dụng findViewById trong Activity



- startActivity đảm bảo Activity được khởi tạo
 - Nếu Activity được khởi tạo, sẽ được đưa lên đầu
 - Activity được quản lý như thế nào?
- Mô hình hướng sự kiện
 - Activity có một số hàm để điều khiển các sự kiện
 - onCreate, onResume, onPause,...
 - Tất cả Activity phải nạp chồng hàm onCreate để thực hiện một việc gì đó
 - Các hàm nạp chồng phải gọi phương thức của superclass





- Ba trạng thái
 - Kích hoạt (active): ở chế độ nền, đang chạy
 - Tạm dừng (pause): vẫn hiển thị nhưng bị che khuất bởi Activity khác
 - Giống active, nhưng có thể bị hủy nếu dung lượng bộ nhớ thấp
 - Dừng (stop): không hiển thị trên màn hình



- onCreate()
 - Gọi khi Activity đầu tiên được tạo
 - Chuẩn bị GUI và các bước khởi tạo khác
- onResume()
 - Gọi khi Activity ở trên đầu Stack
 - Cập nhật giá trị GUI

Chú ý: được gọi khi Activity đầu tiên được khởi tạo
- onPause()
 - Activity chuẩn bị biến mất
 - Cập nhật các dữ liệu quan trọng, dừng các công việc tốn nhiều tài nguyên





- Để gọi activity khác chạy bạn dùng phương thức startActivity

```
Intent i = new  
Intent(getApplicationContext(),ActivityB.class);  
startActivity(i)
```


Put và get data qua intent:

- Gửi:

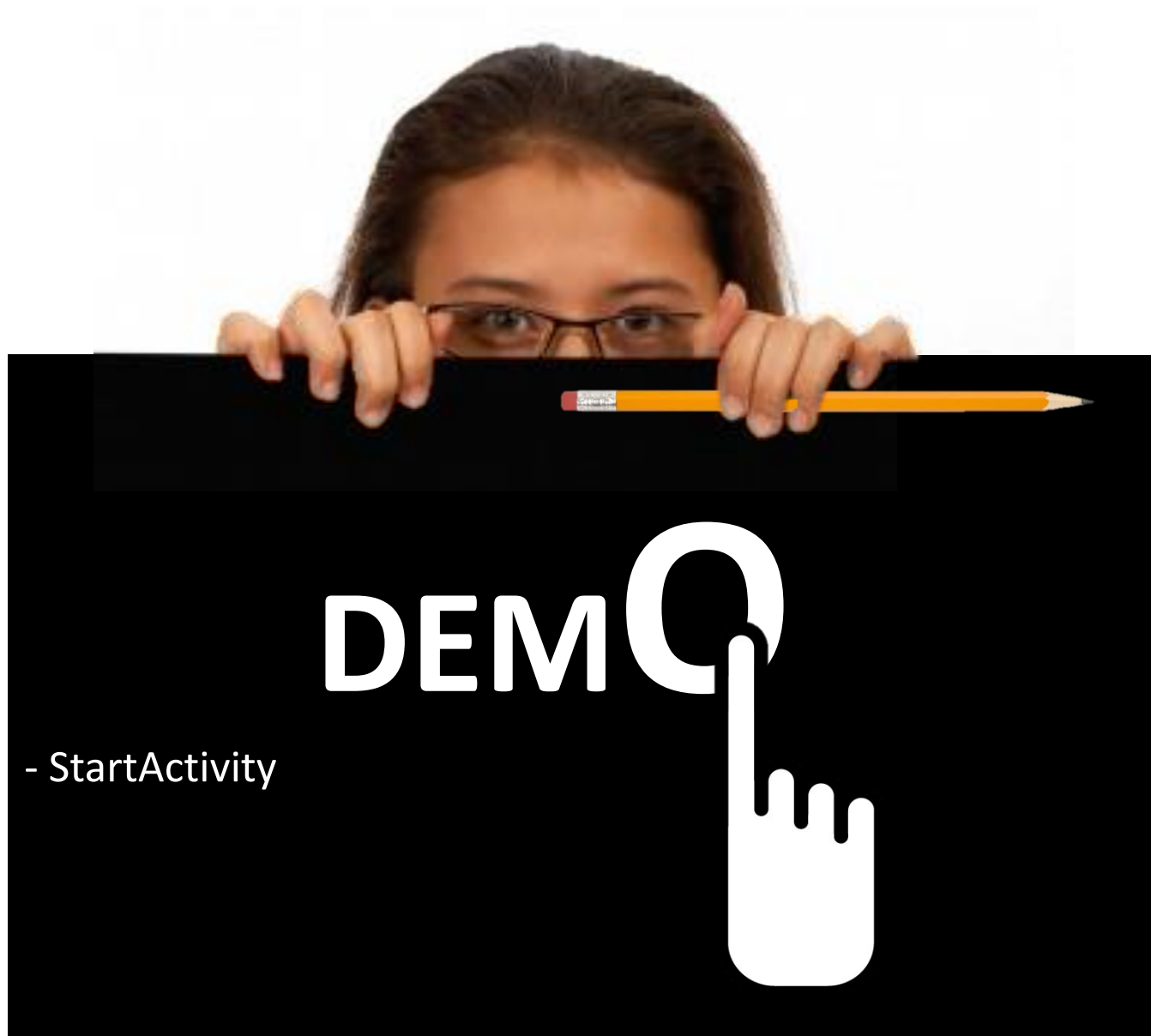
```
Intent i = new Intent(getBaseContext(), ActivityB.class);

//Set the Data to pass
EditText txtInput = (EditText) findViewById(R.id.txtInput);
String txtData = txtInput.getText().toString();
i.putExtra("txtData", txtData);

startActivity(i);
```

- Nhận

```
Intent i = getIntent();
//The second parameter below is the default string returned
String txtData = i.getExtras().getString("txtData", "");
EditText txtInput2 = (EditText) findViewById(R.id.txtInput2);
txtInput2.setText(txtData);
```





DEMO

GIẢNG VIÊN HƯỚNG
DẪN SINH VIÊN
THIẾT KẾ GIAO DIỆN
ACTIVITY ĐƠN GIẢN



Phần I: Các thành phần cơ bản của Android

 Activity


 Service


 ContentProvider

 BroadcastReceiver

 Intent, Context

Phần II: Activity và vòng đời của Activity

 Activity - Task, stack

 Vòng đời của Activity





Cảm ơn