**Intent :**

Android Intent can be defined as a simple message objects which is used to communicate from one activity to another activity.

Or

* Intents are asynchronous messages which allow application components(activity) to request functionality from other Android components(activity).
* Intents allow you to interact with components(activity) from the same applications as well as with components(activities) of other applications.

Example : Camera, Wots up(attaching photo to wots from gallery)….etc….

* Intents are objects of the android.content.Intent type.
* An intent can hold the data with help of Bundle.

**Task :**

* A task is a collection of activities that users interact with the application when performing a certain work in app.
* An application can access other Android application components to achieve a task.
* The activities are arranged in a stack called BACK STACK.
* A task should contain atleast one activity.(Each app should contain atleast one Task)
* New activity created are placed on top of stack.
* The back stack will be in order of LIFO Queue order.
* Task can be moved to Background by keeping in Back Stack.

**Uses of Intent :**

1. Start the service
2. Start the activity
3. Broadcast a message
4. Used to transfer the data between activities.
5. Used to retrieve the data from sub activity.

Types of Intent :

In android there are two types of Intent:

1. Implicit Intent
2. Explicit Intent

**Implicit Intent :**

It doesn't specifiy the target component so that android OS provides information of available components provided by the system that is to be invoked.

Example : Making call, Sending Message, Connecting to web page, Sending email etc….

**Explicit Intent :**

* It provides the target component so that intent provides the external class to be invoked when performing the explicit intent.
* In explicit Intent the source and destination classes should be provided.

Example : Intent intent = **new** Intent(Source class, Destination.**class**);

startActivity(intent);

**Passing data from one Activity to another Activity :**

Intent intent = new Intent(Source class, Destination.class);

intent.putExtra(key, value);

startActivity(intent);

**Retriving the Data :**

Bundle extras = getIntent().getExtras();

extras.getString(key);

note : user can get int,double,char….etc data from getIntent().getExtras() method.

**Passing data with help of Bundle :**

Passing data by bundle :

//create the Intent

Intent i = new Intent(Source class, Destination.class);

//Create the bundle

Bundle bundle = new Bundle();

//Add your data to bundle

bundle.putString(key, value);

//Add the bundle to the intent

i.putExtras(bundle);

startActivity(i);

Retriving the data :

Bundle bundle = getIntent().getExtras();

//Extract the data…

String venName = bundle.getString(key);

//when retrieving the data we can retrieve different type of data like inr,double,char etc…..

StartActivityForResult :

* The android startActivityForResult() method,  we can send information from one activity to another activity and we can get the information from previous activity to present activity.
* **startActivityForResult** method, requires a result from the second activity. So that we need to override the **onActivityResult** method that is invoked automatically when second activity returns result to first activity.

Syntax :

public void startActivityForResult (Intent intent, int requestCode)

* When we get data back from second activity to first activity then to get the data in first activity we should override onActivityResult() method.

**Public  void onActivityResult(int requestCode, int resultCode, Intent data)**

**{**

**……..**

**……..**

**………**

**}**

* User want to fetch the data based on request code.

# [setResult](http://stackoverflow.com/questions/12233106/really-not-getting-setresult-and-onactivityresult) :

# setResult() is used to send the data back to first activity.

# Data is sent through the intent along with request code.

# finish():

# Is used to finish the current running activity.

# finish() mehod will call onDestroy method of the current activity.

**Intent Filter :**

* Intent Filter are used to send signal to the Android OS saying that a certain event has occurred by some application.
* Intent filter will describe the action which should be performed and will provide data upon which such an action should be completed.

Example : A application can start a browser component for a certain URL via an intent. But how does the Android OS will identify the components which can react to a certain intent.? So that A component should be registered itself via an intent filter for a specific action and specific data.

* An intent filter will specifies the different types of intents to which an activity, service, or broadcast receiver can respond to by declaring the capabilities of a component.
* Android components will register intent filters in the AndroidManifest.xml file.
* An intent filter is defined by its category, action and data filters.

Example to register intent filter :

<activity android:name=".className"

android:label="@string/app\_name">

<intent-filter>

<action android:name="android.intent.action.VIEW" />

<category android:name="android.intent.category.DEFAULT" />

<data android:scheme="http"/>

</intent-filter>

</activity>