

SESSION 01

- onSaveInstanceState() and onRestoreInstanceState()
 - Bundle stores NVP - Name-Value Pair
 - onSaveInstanceState() is called between onPause() and onStop()
 - onRestoreInstanceState() is called between onStart() and onPostCreate(Bundle)
- startActivityForResult() - [[https://developer.android.com/reference/android/app/Activity#startActivityForResult\(android.content.Intent,%20int\).](https://developer.android.com/reference/android/app/Activity#startActivityForResult(android.content.Intent,%20int).)]
- onActivityResult() - [[https://developer.android.com/reference/android/app/Activity#onActivityResult\(int,%20int,%20android.content.Intent\).](https://developer.android.com/reference/android/app/Activity#onActivityResult(int,%20int,%20android.content.Intent).)]
- registerForActivityResult()
 - ActivityResultCallback
 - ActivityResultLauncher
- Intent
 - implicit intent vs explicit intent
 - ◆ **Implicit intents** do not name a specific component, but instead declare a general action to perform, which allows a component from another app to handle it. For example, if you want to show the user a location on a map, you can use an implicit intent to request that another capable app show a specified location on a map.
 - ◆ **Explicit intents** specify which application will satisfy the intent, by supplying either the target app's package name or a fully-qualified

component class name. You'll typically use an explicit intent to start a component in your own app, because you know the class name of the activity or service you want to start. For example, you might start a new activity within your app in response to a user action, or start a service to download a file in the background.

- ◆ Without a component name, the intent is *implicit* and the system decides which component should receive the intent based on the other intent information.
 - ◆ Action
 - ◆ `setAction()`
 - ◆ `ACTION_VIEW`
 - ◆ `ACTION_SEND`
 - ◆ Data
 - ◆ `setData()`
 - ◆ Category
 - ◆ `CATEGORY_LAUNCHER`
 - ◆ `CATEGORY_BROWSABLE`
- `putExtra` name parameter needs package prefix - [<https://stackoverflow.com/questions/12269019/why-do-the-android-docs-say-intent-extras-need-package-prefix>]
- Add Upward Navigation to activities, in `AndroidManifest.xml`
- `<intent-filter>` - [<https://stackoverflow.com/questions/3321514/what-are-intent-filters-in-android>], [<https://developer.android.com/guide/topics/manifest/intent-filter-element>]
- Intent Resolution

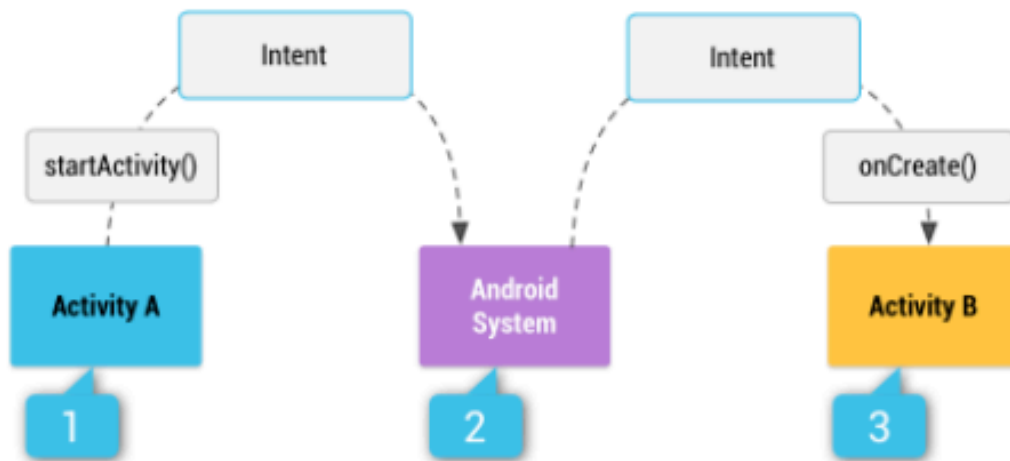


Figure 1. How an implicit intent is delivered through the system to start another activity: **[1]** Activity A creates an `Intent` with an action description and passes it to `startActivity()`. **[2]** The Android System searches all apps for an intent filter that matches the intent. When a match is found, **[3]** the system starts the matching activity (Activity B) by invoking its `onCreate()` method and passing it the `Intent`.

```
intent.setAction(Intent.ACTION_VIEW);
intent.setData(ContactsContract.Contacts.CONTENT_URI);
startActivity(intent);
```

```
intent.setAction(Settings.ACTION_SETTINGS);
intent.addFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
startActivity(intent);
```

```
intent.setAction(Intent.ACTION_SEND);
intent.putExtra(Intent.EXTRA_TEXT, value: "Hello, World!");
intent.setType("text/plain");

if(intent.resolveActivity(getPackageManager()) != null)
    startActivity(intent);
```

```
public void resultActivity(View view) {
    Intent intent = new Intent( packageContext: this, ActivityTwo.class);
    startActivityForResult(intent, RESULT_FROM_ACTIVITY_TWO);
}

@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data) {

    super.onActivityResult(requestCode, resultCode, data);

    if(requestCode == RESULT_FROM_ACTIVITY_TWO && resultCode == Activity.RESULT_OK) {
        TextView textView = findViewById(R.id.tv_two);
        textView.setText(data.getStringExtra(Keystore.EXTRA_MESSAGE));
    }
}
```

```
Intent intent = new Intent();

intent.putExtra(Keystore.EXTRA_MESSAGE, value: "result back from activity");
setResult(Activity.RESULT_OK, intent);
finish();
```