University of Anbar
College of Computer Science and Information Technology
Computer Science Department

Mobile Application Programming

Lab Six

Third Stage

First Course 2023 - 2024

Rihab Hazim Qasim

MSc Computer Science

rehz1991@uoanbar.edu.iq

Application Components – Activity

- Implementing your activity lifecycle methods properly ensures your app behaves well in several ways, including that it:
  Does not crash if user receives a phone call or switches to another app while using your app.
  Does not consume valuable system resources when user is not actively using it.
  Does not lose user's progress if they leave your app and return to it at a later time.
  Does not crash or lose user's progress when screen rotates between landscape and portrait orientation.

Starting an Activity

```
Intent intent = new Intent(this, SignInActivity.class);
startActivity(intent);
```

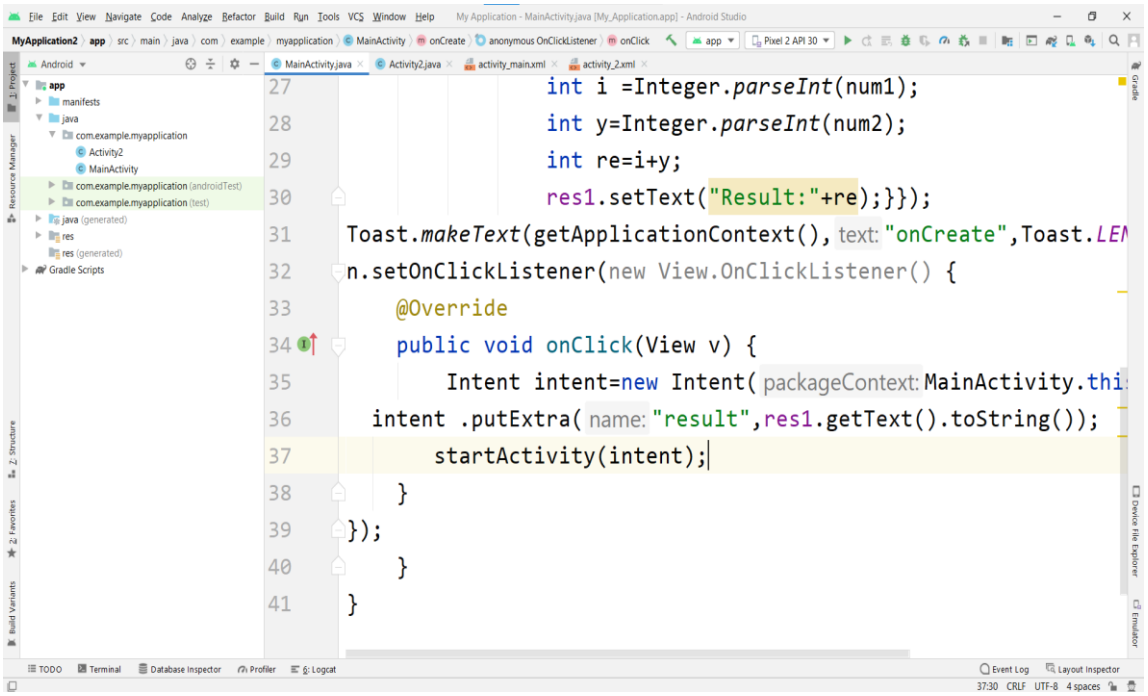
Specify Your App's Launcher Activity

- From AndroidManifest.xml
- If either MAIN action or LAUNCHER category are not declared for one of your activities, then your app icon will not appear in Home screen's list of apps.

Shutting Down an Activity

- Manually shut down an activity by calling its finish() method. You can also shut down a separate activity that you previously started by calling finishActivity().

Application Components – Activity Example (zoom it in)



Application Components – Intents

- Intent is an abstract concept or data structure for some operation that should be performed in the Android OS.
- Often used to launch external applications with intent to do something, such as make a phone call, display a web page, or map an address.
- Can be used with startActivity() to launch an Activity, broadcastIntent to send it to any interested BroadcastReceiver components, and startService(Intent) or bindService(Intent, ServiceConnection, int) to communicate with a background Service.
- Provides a facility for performing late runtime binding between code in different applications.
- Most significant use is in launching of activities, where it can be thought of as glue between activities.

Intent Object

- An Intent generally has two pieces of information associated with it;
    1. What the intent is (as in make a phone call), **(Action)**
    2. What data does the intent need (such as a phone number) to perform the intention. **(Data)**

Intents – Types of Intents

1. Explicit Intent
   Used to launch a specific app component, such as a particular activity or

service in your app.

```
// Explicit Intent by specifying its class name

Intent i = new Intent(FirstActivity.this, SecondAcitivity.class);

// Starts TargetActivity

startActivity(i);
```

2. Implicit Intent

Specifies an action that can invoke any app on the device able to perform the action.

```
Intent read1=new Intent();

read1.setAction(android.content.Intent.ACTION_VIEW);

read1.setData(ContactsContract.Contacts.CONTENT_URI);

startActivity(read1);
```

Intent Fields – Action

- Action is a String representing desired operation

Some Standard Activity Actions

android.intent.action.ACTION_DIAL – Dial a number
android.intent.action.ACTION_EDIT – Display data to edit
android.intent.action.ACTION_SYNC – Synchronize device data with server
android.intent.action.ACTION_MAIN – Start as initial activity of app
android.intent.action.CALL – Perform a call to someone specified by the data
android.intent.action.WEB_SEARCH – Perform a web search
android.intent.action.SENDTO – Send a message to someone specified by the data

- Action in an Intent object can be set by the setAction() method and read by getAction()

Setting the Intent Action

```
Intent newInt = newIntent(Intent.ACTION_DIAL);   //Or
Intent newInt = new Intent();
newInt.setAction(Intent.ACTION_DIAL);
```

Intent Fields – Data

- Data associated with the Intent
- Formatted as a Uniform Resource Identifier (URI)

Examples

i. Data to view on a map

Uri.parse("geo:0,0? q=1600+Pennsylvania+Ave+Washington+DC");

i. Number to dial in phone dialer

Uri.parse("tel:+3007001001");

- The setData() method specifies data only as a URI and The URI is read by getData().

Setting Intent Data

```
Intent newInt = new Intent (Intent.ACTION_DIAL,
Uri.parse("tel:+3007001001"));
Or
Intent newInt = new Intent(Intent.ACTION_DIAL);
newInt.setData(Uri.parse("tel:+3007001001"));
```