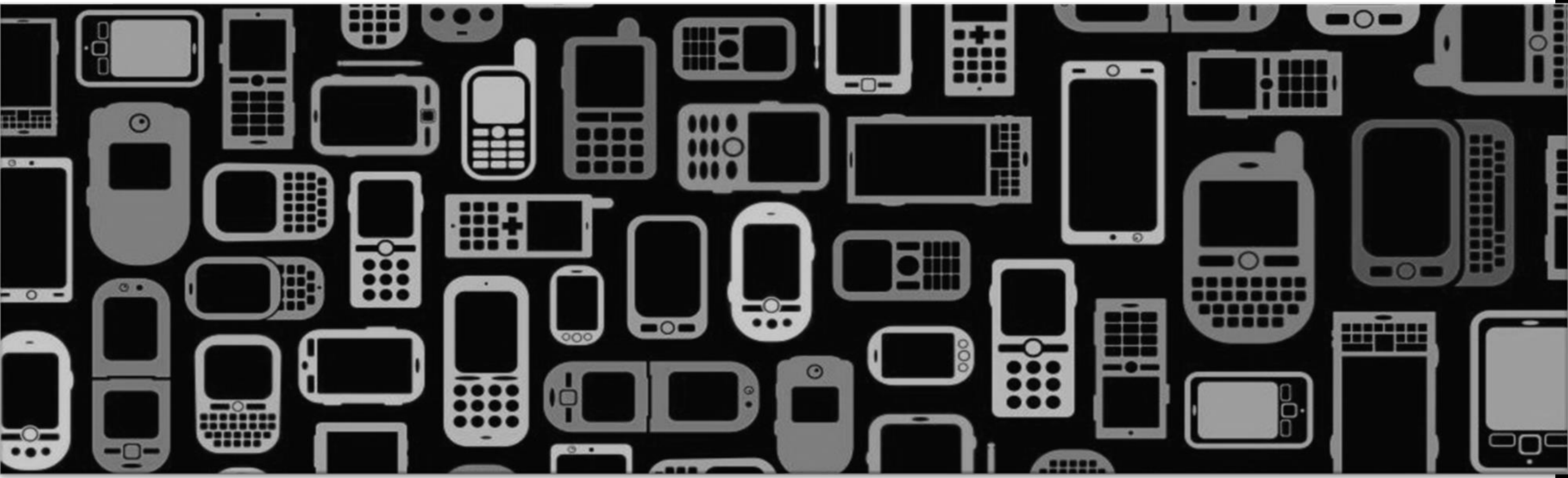
FILKOM | UB



CCE60220

Perangkat Bergerak (TKOM)

Fakultas Ilmu Komputer Universitas Brawijaya

MATAKULIAH	: **Perangkat Bergerak (TKOM)**
KODE/ STATUS	: CCE60220
SKS	: 2
Dosen	: Dahnial Syauqy, S.T, M.T
Email	: dahnial87@ub.ac.id
Ruang	:

Agenda Perkuliahan

1. Intro dan overview perkuliahan
2. Sejarah dan perkembangan teknologi perangkat bergerak
3. Komponen perangkat keras dan perangkat lunak
4. Pengenalan dan instalasi android studio serta aplikasi sederhana
5. Intent dan passing data pada Android Studio
6. Android Studio: Sensor reading
7. Android Studio: Storage & shared preference
8. **=========================UTS**
9. Pengenalan dan aplikasi sederhana dengan MIT AppInventor
10. Appinventor: variable, looping, conditional, tinyDB, file
11. appInventor: sensor reading & persiapan project
12. Appinventor: Akuisisi gambar dan suara
13. Appinventor: komunikasi bluetooth
14. Appinventor: basic animation
15. Presentasi kelompok
16. **=========================UAS**

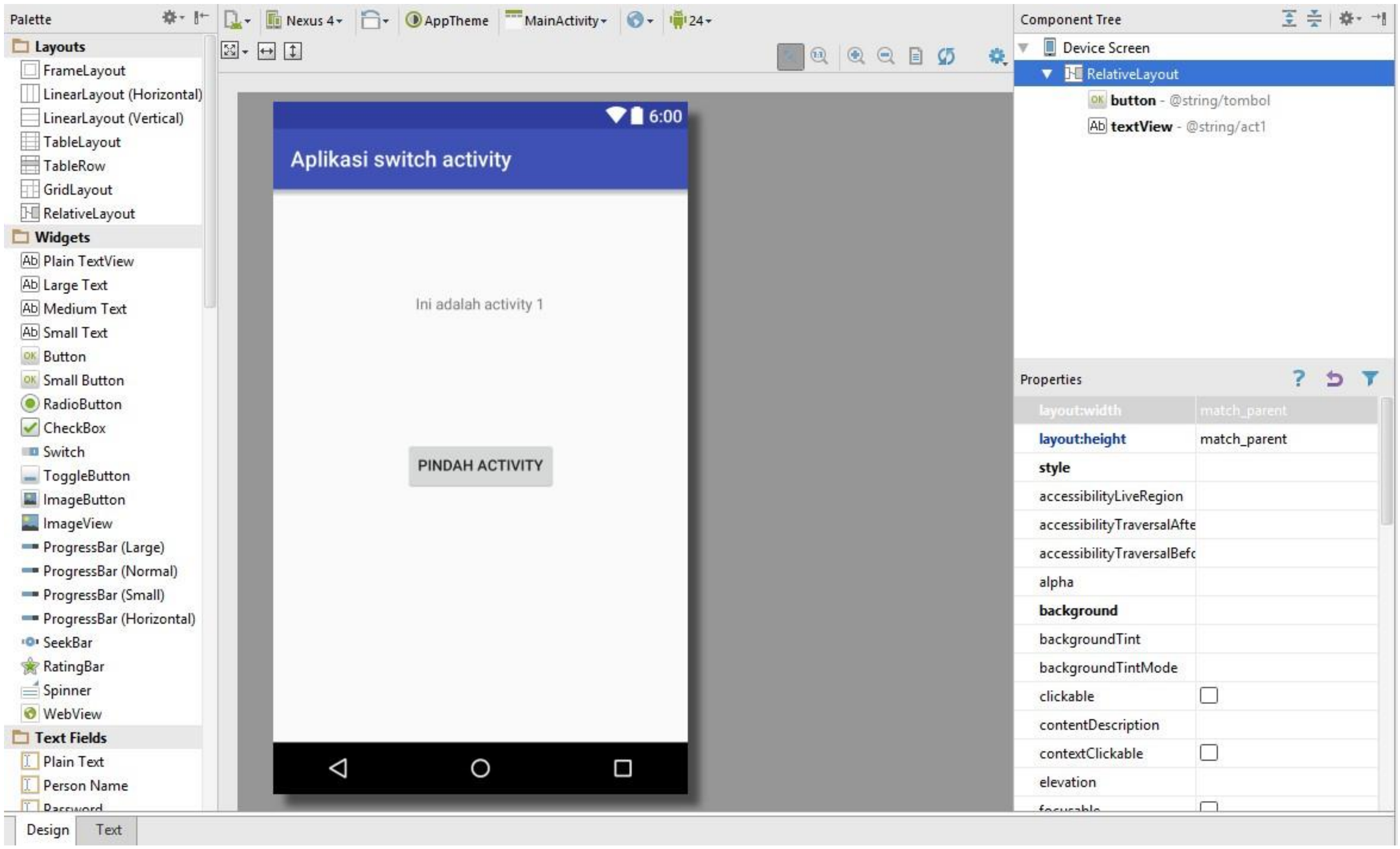
Last week …

- Android Studio
- Struktur dasar
- Listener & Event handler
- Giving your app “function”

Switching Activities
Passing data between activities
Webview
intents

Adding more activities to your current activity

For an example, Current activity is “Empty activity” added with button and textview

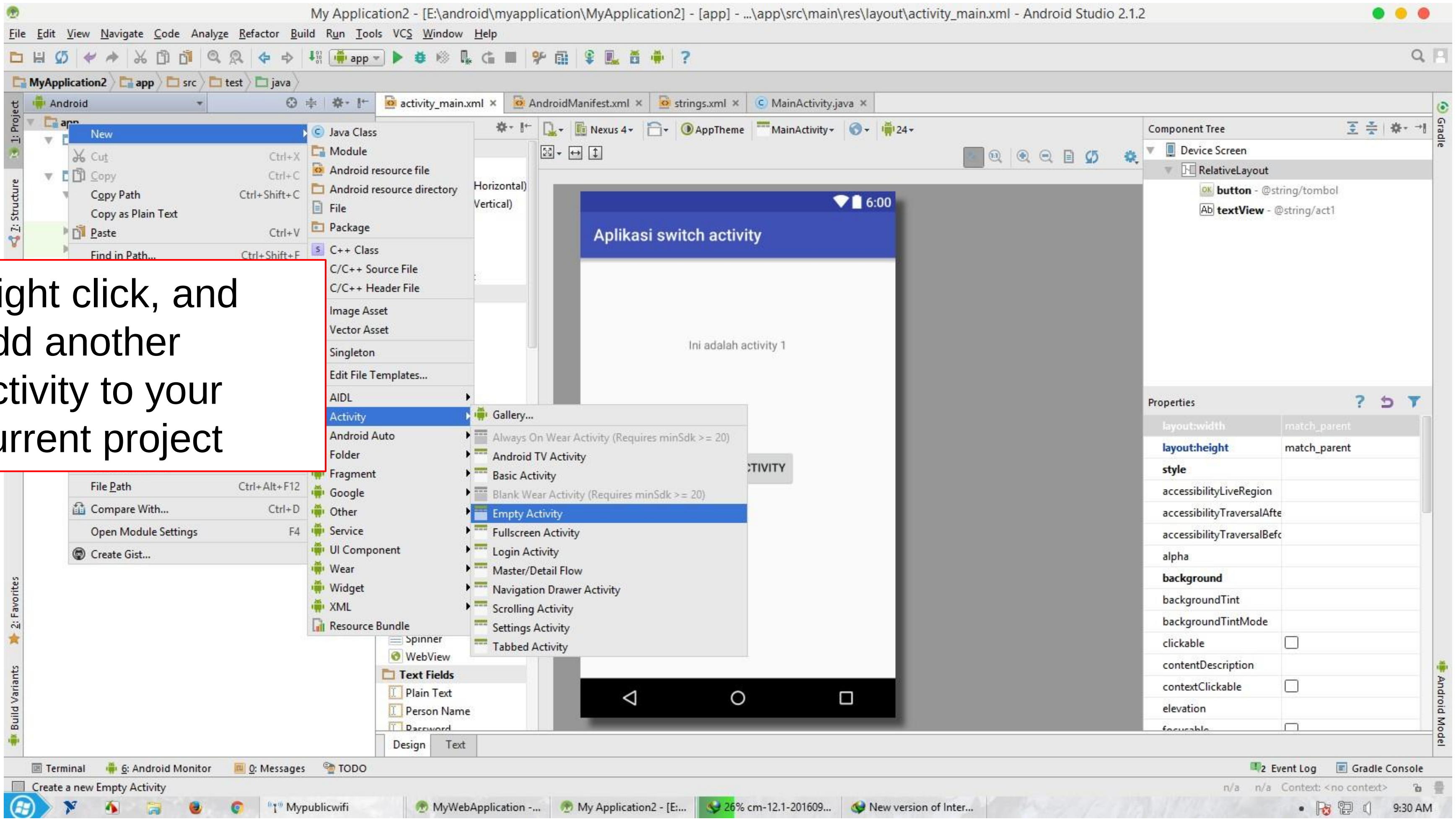


Component Tree
- Device Screen
  - RelativeLayout
    - button - @string/tombol
    - textView - @string/act1

Properties

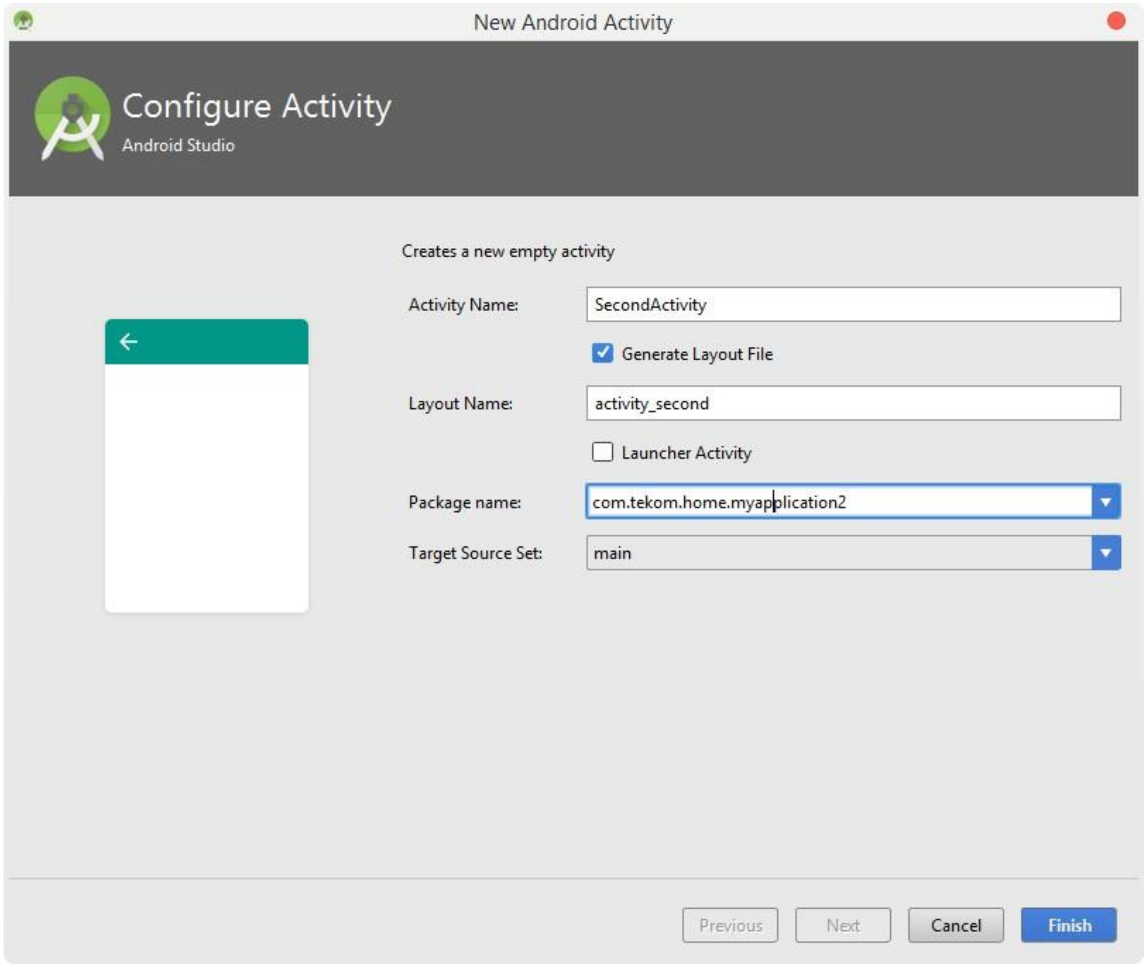
layout:width	match_parent
layout:height	match_parent
style	
accessibilityLiveRegion	
accessibilityTraversalAfte	
accessibilityTraversalBefc	
alpha	
background	
backgroundTint	
backgroundTintMode	
clickable	☐
contentDescription	
contextClickable	☐
elevation	
focusable	☐

Let’s add another activity to our current activity. Here’s how...



Right click, and add another activity to your current project

Give your new activity a name, and remember to uncheck "launcher activity"



New Android Activity

Configure Activity
Android Studio

Creates a new empty activity

Activity Name: SecondActivity

[✓] Generate Layout File

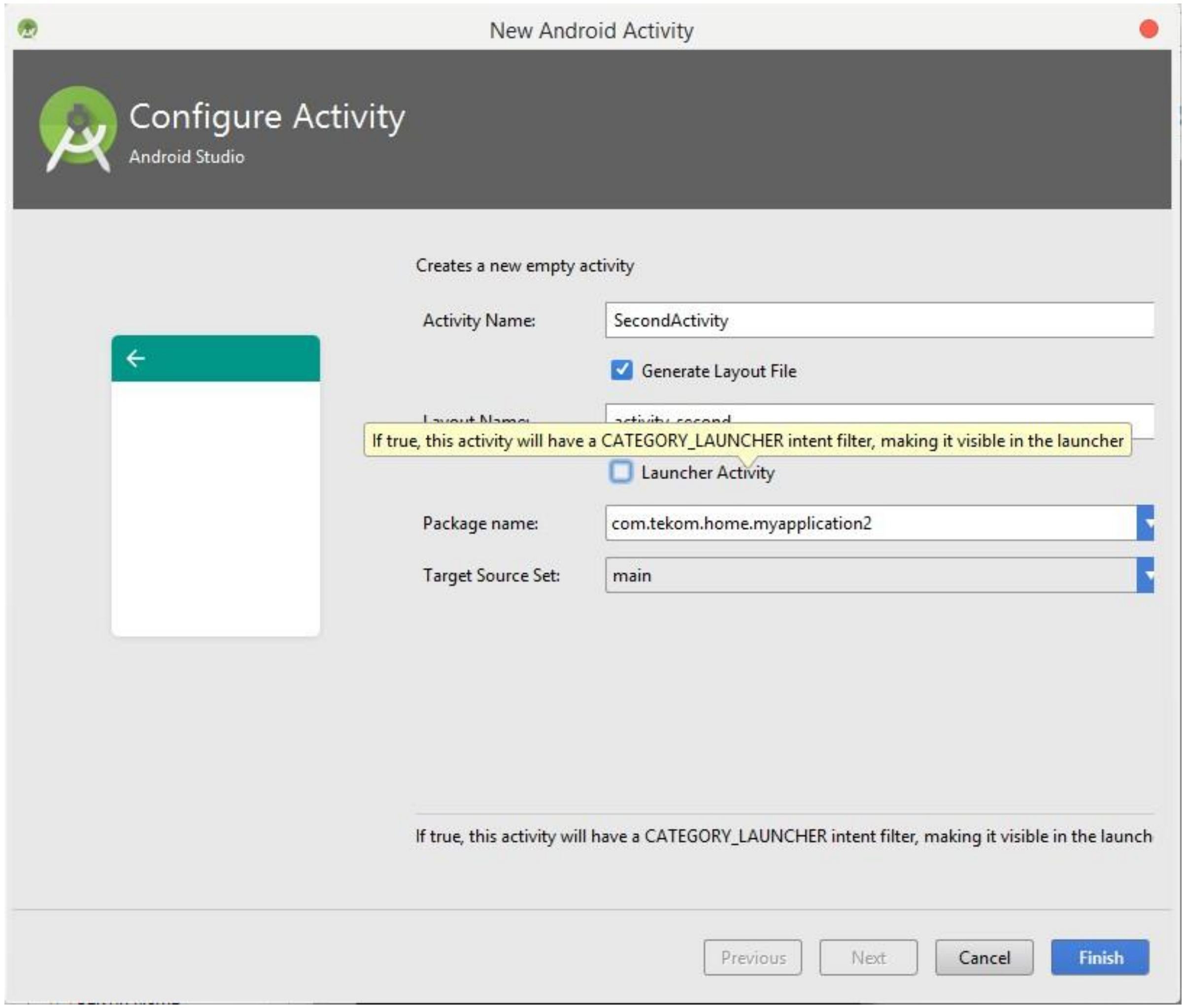
Layout Name: activity_second

[ ] Launcher Activity

Package name: com.tekom.home.myapplication2

Target Source Set: main

Why? It configures whether the activity will appear on the launcher or not


New Android Activity

Configure Activity

Android Studio

Creates a new empty activity

Activity Name: SecondActivity

[✓] Generate Layout File

Layout Name: activity_second

If true, this activity will have a CATEGORY_LAUNCHER intent filter, making it visible in the launcher

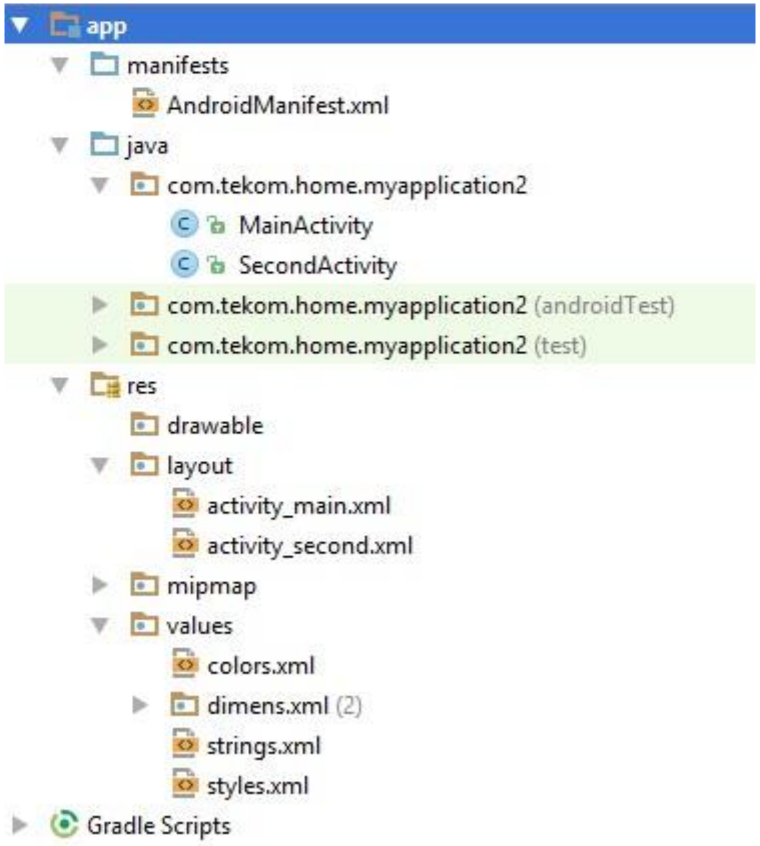
[ ] Launcher Activity

Package name: com.tekom.home.myapplication2

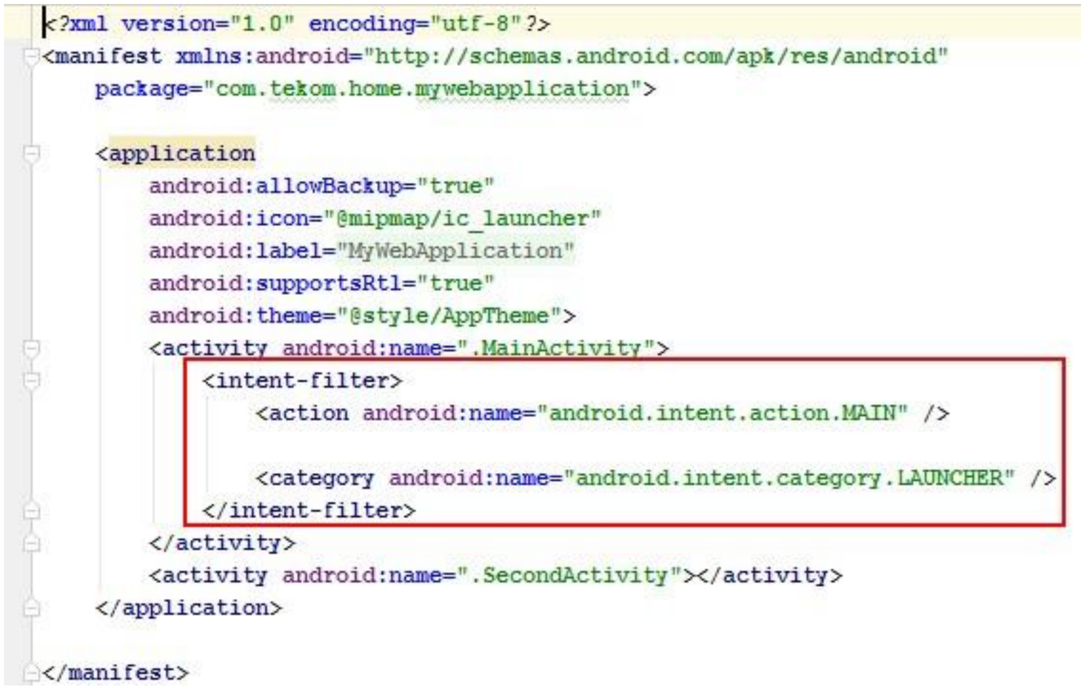
Target Source Set: main

If true, this activity will have a CATEGORY_LAUNCHER intent filter, making it visible in the launch

The wizard will generate both secondactivity java file and xml-layout activity. It will also updates the androidmanifest file

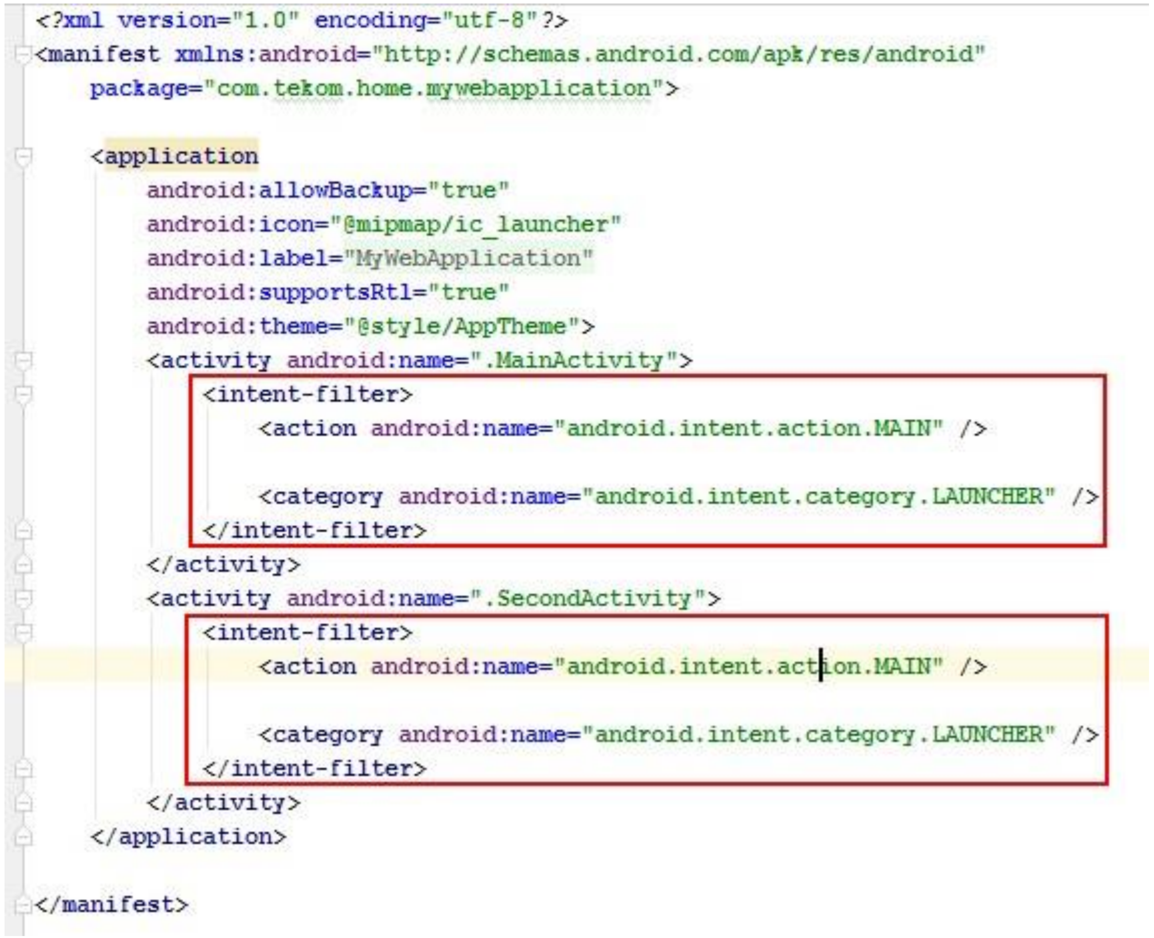


```
app
  manifests
    AndroidManifest.xml
  java
    com.tekom.home.myapplication2
      MainActivity
      SecondActivity
    com.tekom.home.myapplication2 (androidTest)
    com.tekom.home.myapplication2 (test)
  res
    drawable
    layout
      activity_main.xml
      activity_second.xml
    mipmap
    values
      colors.xml
      dimens.xml (2)
      strings.xml
      styles.xml
Gradle Scripts
```

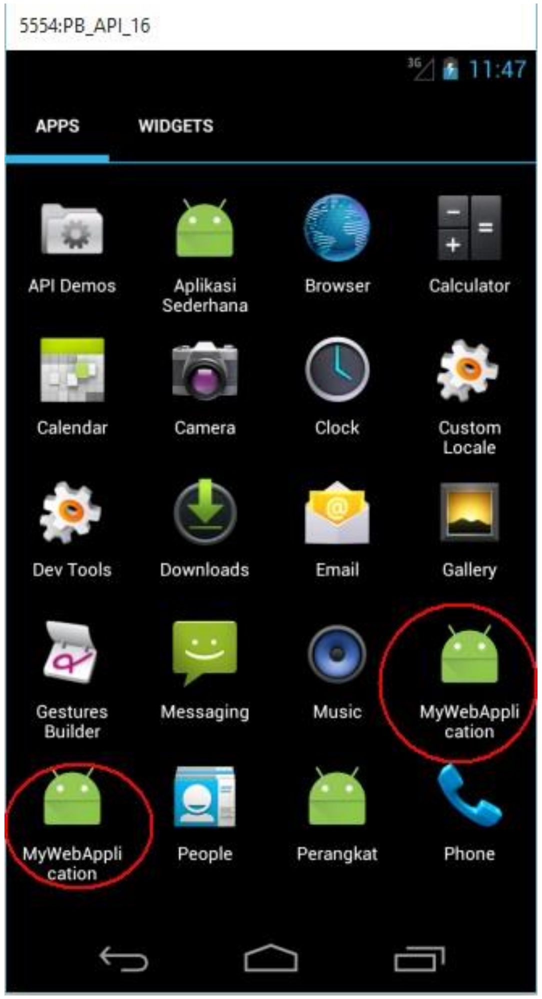


```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.tekom.home.mywebapplication">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="MyWebApplication"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity android:name=".SecondActivity"></activity>
    </application>

</manifest>
```

What happen if you place intent category launcher on both activity?

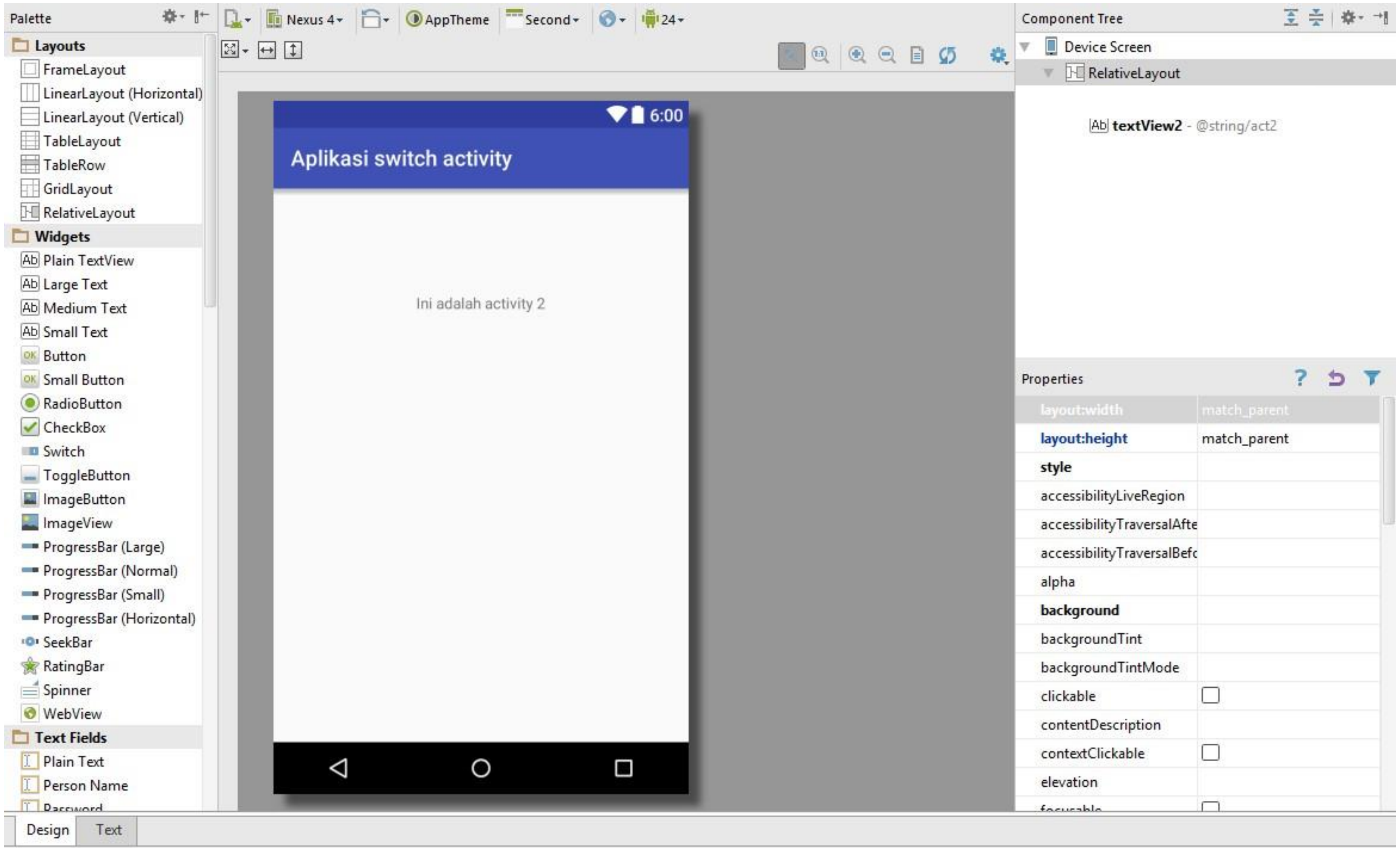


```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.tekom.home.mywebapplication">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="MyWebApplication"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity android:name=".SecondActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```



17 March 2023

In the second activity,
Add content as the first activity: textview



Aplikasi switch activity

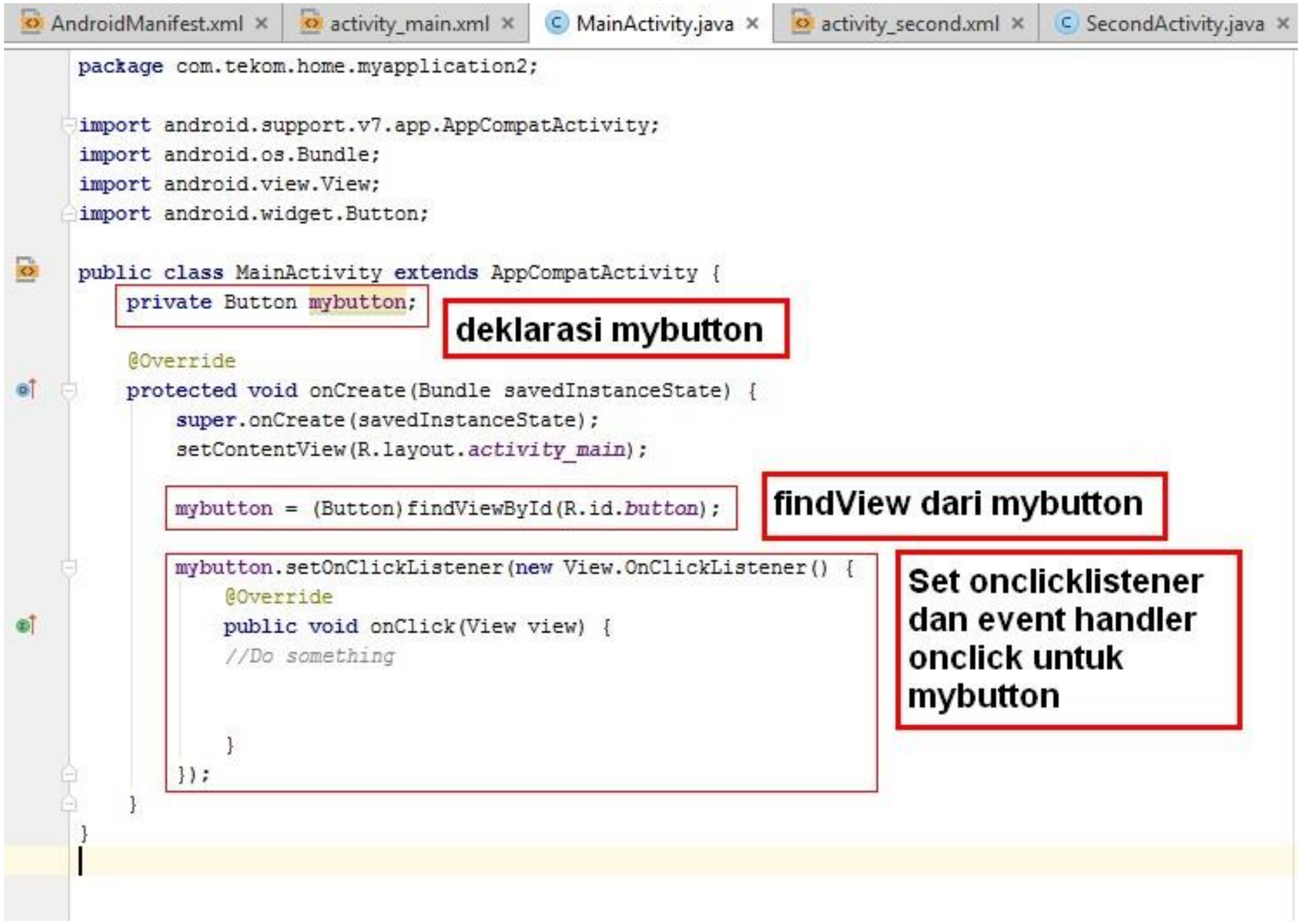
Ini adalah activity 2

Component Tree
- Device Screen
  - RelativeLayout
    - textView2 - @string/act2

Properties

layout:width	match_parent
layout:height	match_parent
style	
accessibilityLiveRegion	
accessibilityTraversalAfte	
accessibilityTraversalBefc	
alpha	
background	
backgroundTint	
backgroundTintMode	
clickable	☐
contentDescription	
contextClickable	☐
elevation	
focusable	☐

Now open the mainactivity.java file and modify the file



```java
package com.tekom.home.myapplication2;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;

public class MainActivity extends AppCompatActivity {
    private Button mybutton;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        mybutton = (Button)findViewById(R.id.button);

        mybutton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
            //Do something


            }
        });
    }
}
```

What will happen when the buttton is clicked? It needs to start the second activity

So, we need to implement Intent in the onclick event handler to start the second activity

What is an intent?

Intents

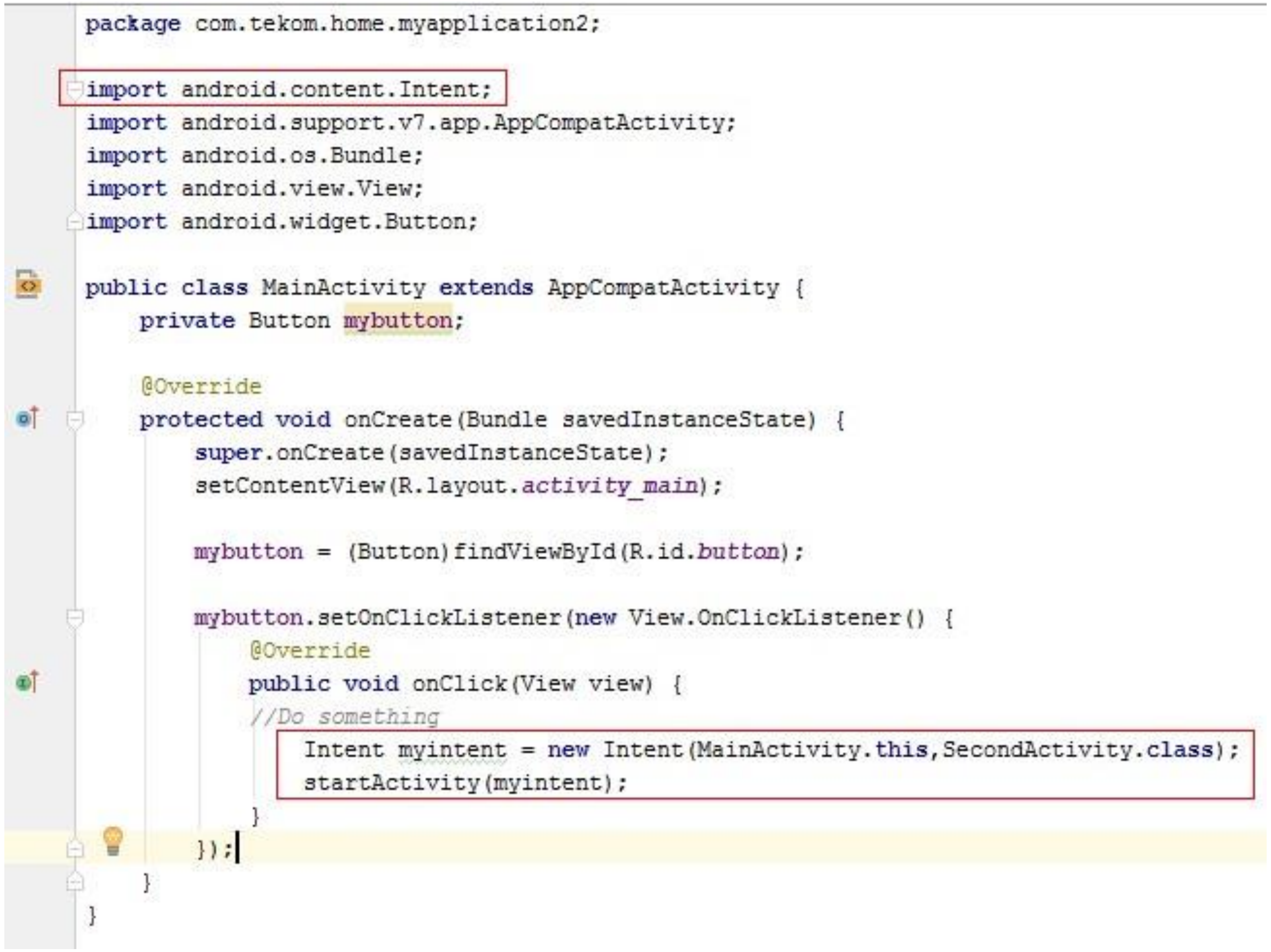
Intents are asynchronous messages which allow application components to request functionality from other Android components.

Intents allow you to interact with components from the same applications as well as with components contributed by other applications. For example, an activity can start an external activity for taking a picture.

- The simplest intent type is the Explicit Intent
- For when you know what component to use and don’t want free access to the user to choose
- We can explicitly launch an activity, service or broadcast receiver

```
Intent intent_name = new Intent(ActivityA.this, ActivityB.class)
startActivity(intent_name)
```

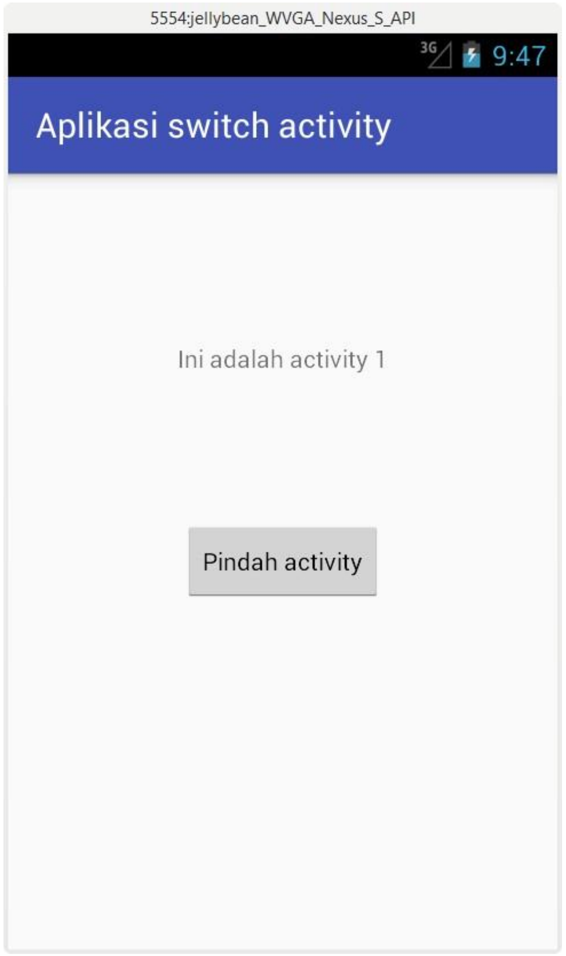
In the mainactivity we will invoke secondactivity in onclick event handler

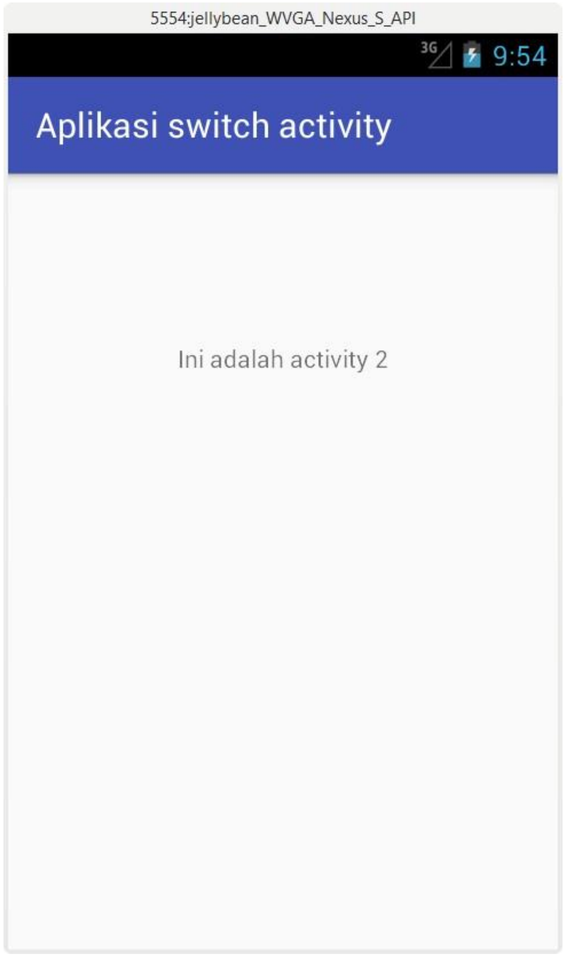


```
package com.tekom.home.myapplication2;

import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;

public class MainActivity extends AppCompatActivity {
    private Button mybutton;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        mybutton = (Button)findViewById(R.id.button);

        mybutton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
            //Do something
                Intent myintent = new Intent(MainActivity.this,SecondActivity.class);
                startActivity(myintent);
            }
        });
    }
}
```

Result

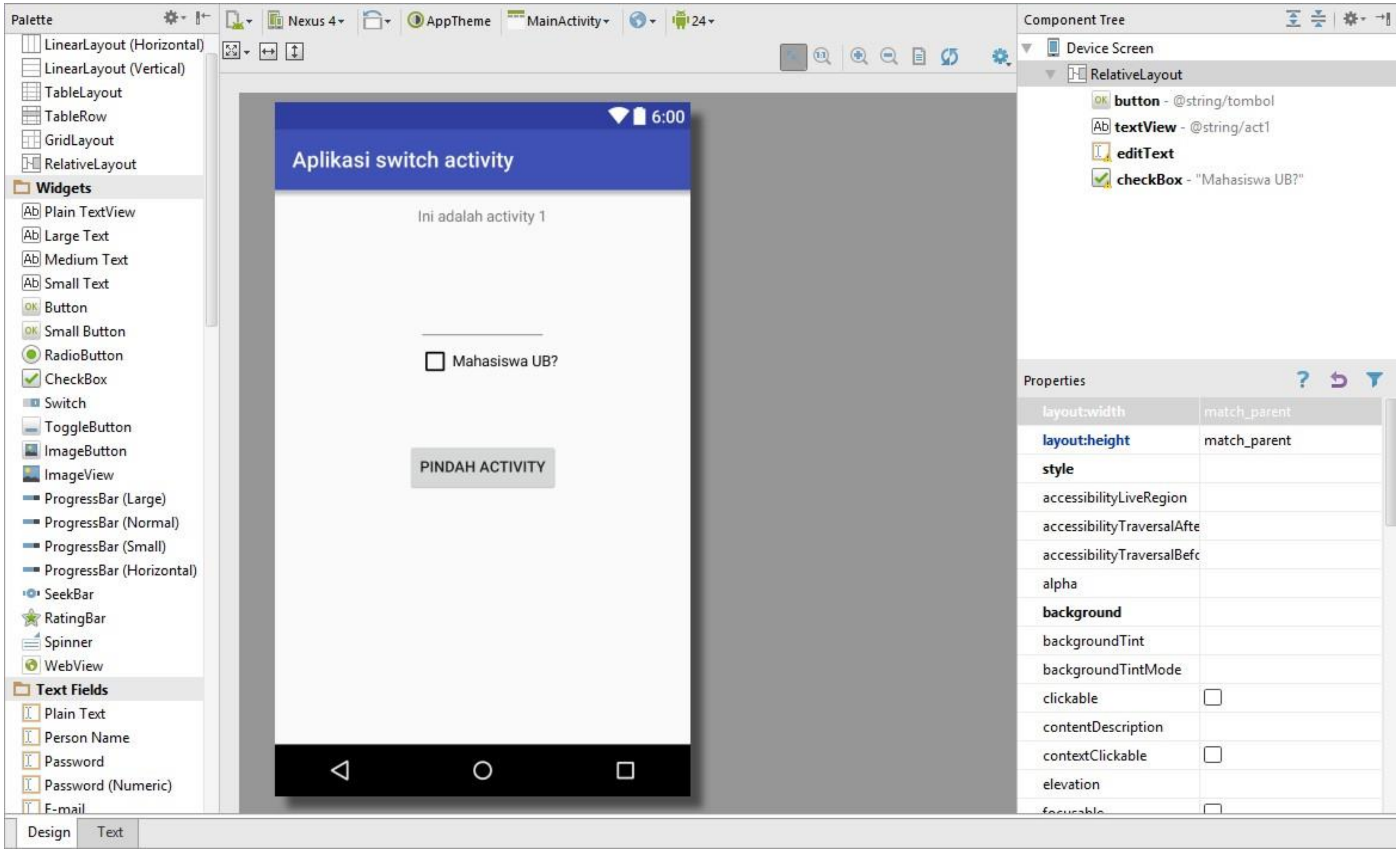
Remember this is different screen





Now, how if we want to pass data/send data from one activity to another?

Let’s try to pass “string” and “boolean” data type.
In the first activity add edittext and checkbox



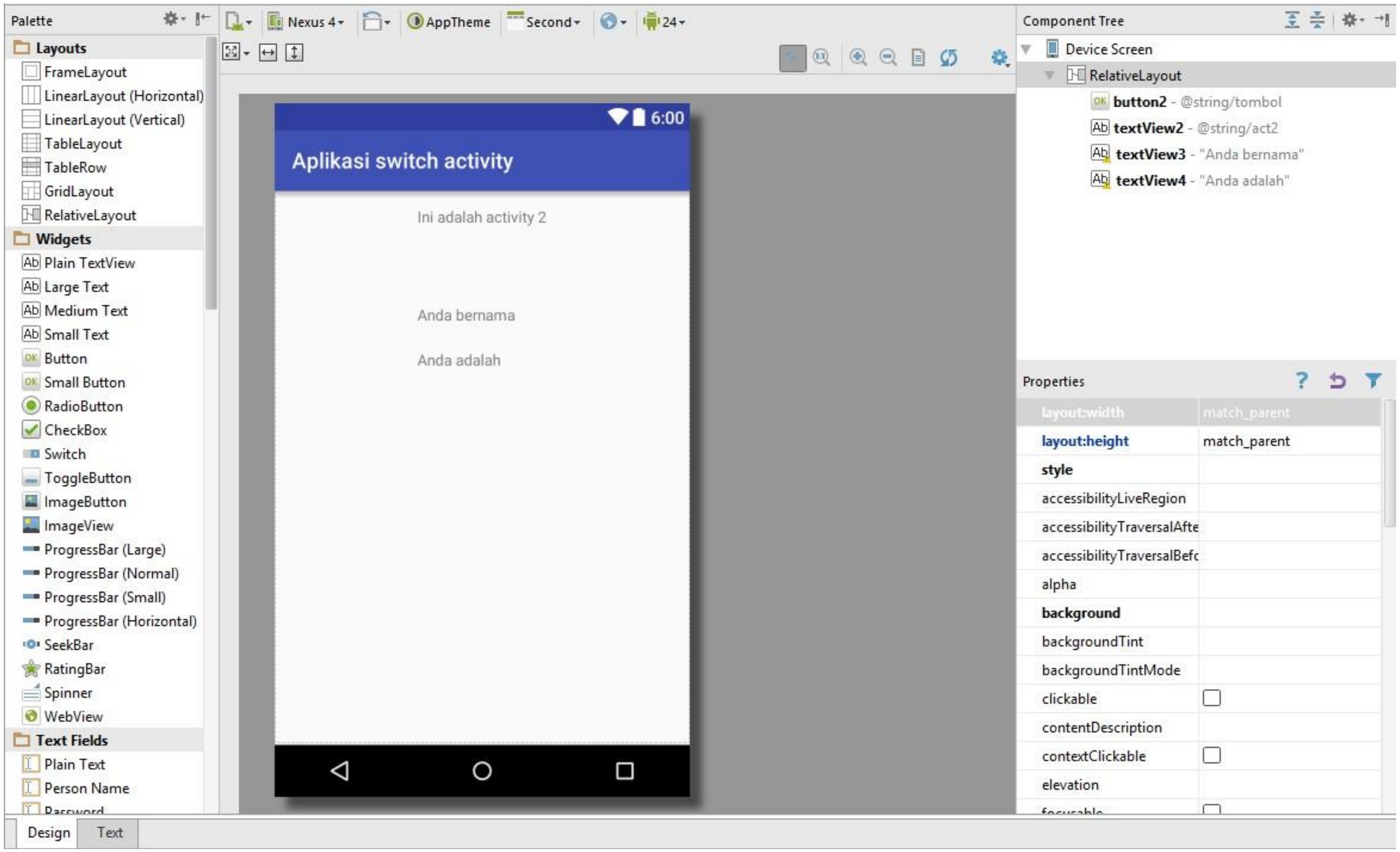
Component Tree

- Device Screen
  - RelativeLayout
    - **button** - @string/tombol
    - **textView** - @string/act1
    - **editText**
    - **checkBox** - "Mahasiswa UB?"

Properties

layout:width	match_parent
layout:height	match_parent
style	
accessibilityLiveRegion	
accessibilityTraversalAfte	
accessibilityTraversalBefo	
alpha	
background	
backgroundTint	
backgroundTintMode	
clickable	☐
contentDescription	
contextClickable	☐
elevation	
focusable	☐

And this is the second activity that will get/receive the data


Palette

Aplikasi switch activity

Ini adalah activity 2

Anda bernama

Anda adalah

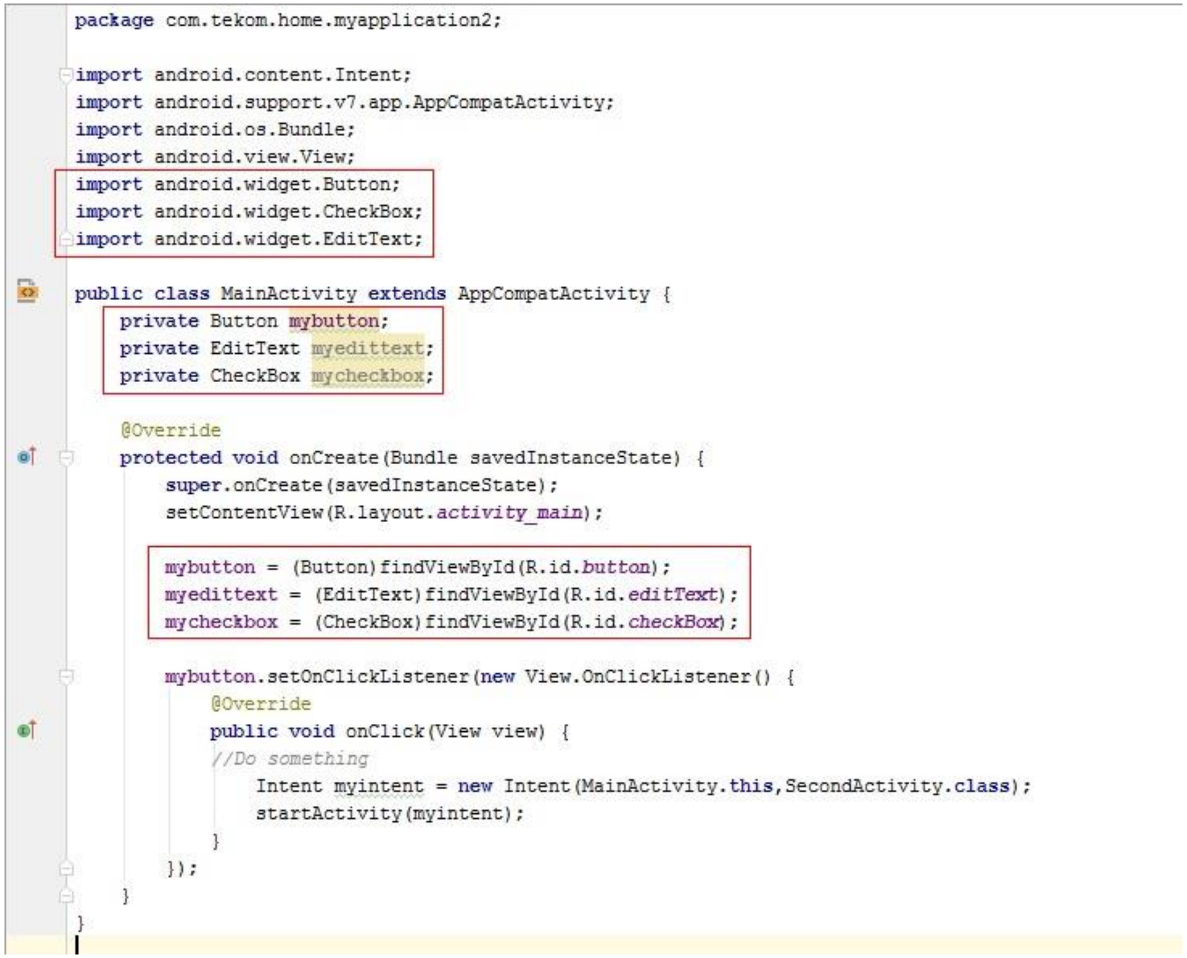
Component Tree

- Device Screen
  - RelativeLayout
    - button2 - @string/tombol
    - textView2 - @string/act2
    - textView3 - "Anda bernama"
    - textView4 - "Anda adalah"

Properties

layout:width	match_parent
layout:height	match_parent
style	
accessibilityLiveRegion	
accessibilityTraversalAfte	
accessibilityTraversalBefo	
alpha	
background	
backgroundTint	
backgroundTintMode	
clickable	☐
contentDescription	
contextClickable	☐
elevation	
focusable	☐

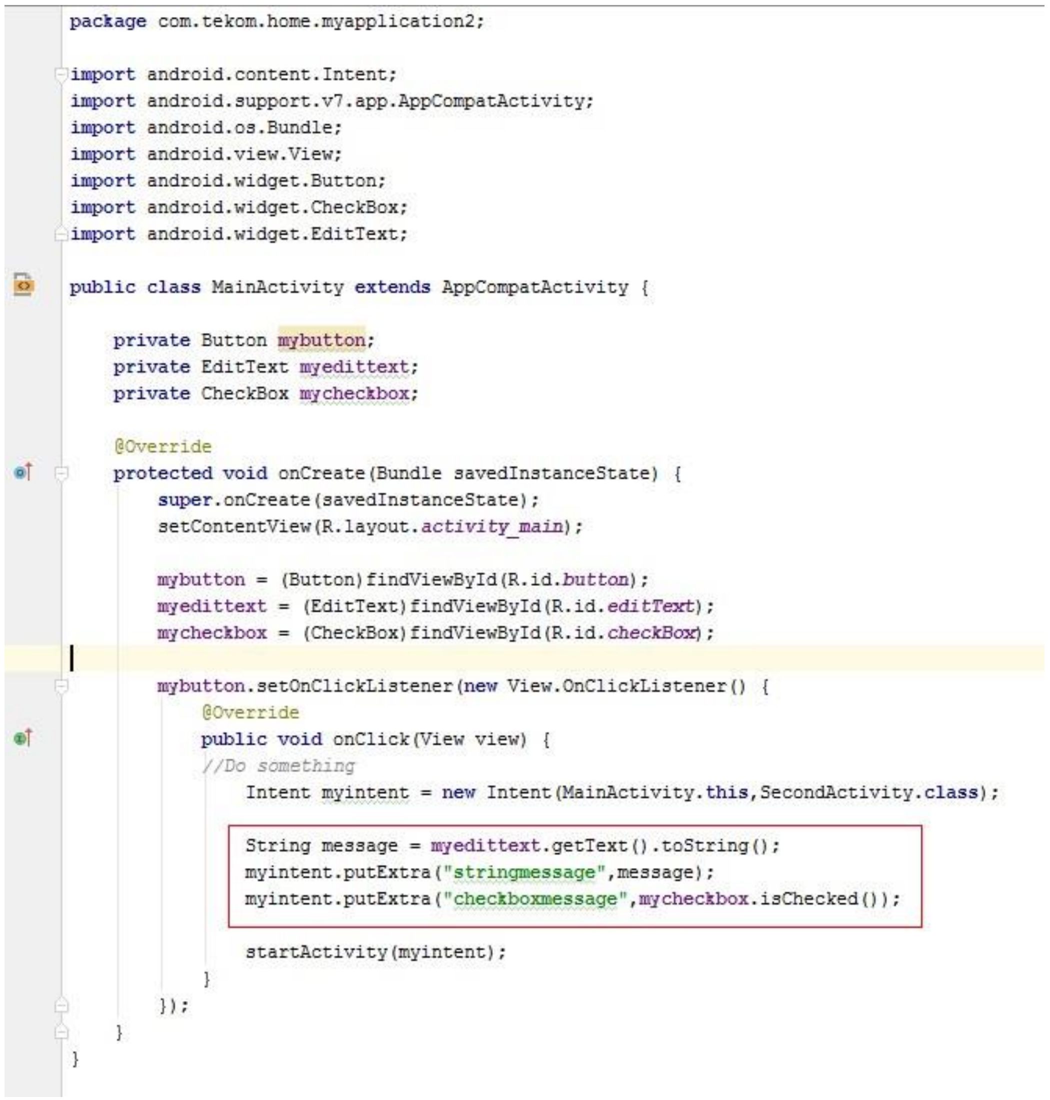
In the first activity java file:



```
package com.tekom.home.myapplication2;

import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.CheckBox;
import android.widget.EditText;

public class MainActivity extends AppCompatActivity {
    private Button mybutton;
    private EditText myedittext;
    private CheckBox mycheckbox;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        mybutton = (Button)findViewById(R.id.button);
        myedittext = (EditText)findViewById(R.id.editText);
        mycheckbox = (CheckBox)findViewById(R.id.checkBox);

        mybutton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
            //Do something
                Intent myintent = new Intent(MainActivity.this,SecondActivity.class);
                startActivity(myintent);
            }
        });
    }
}
```

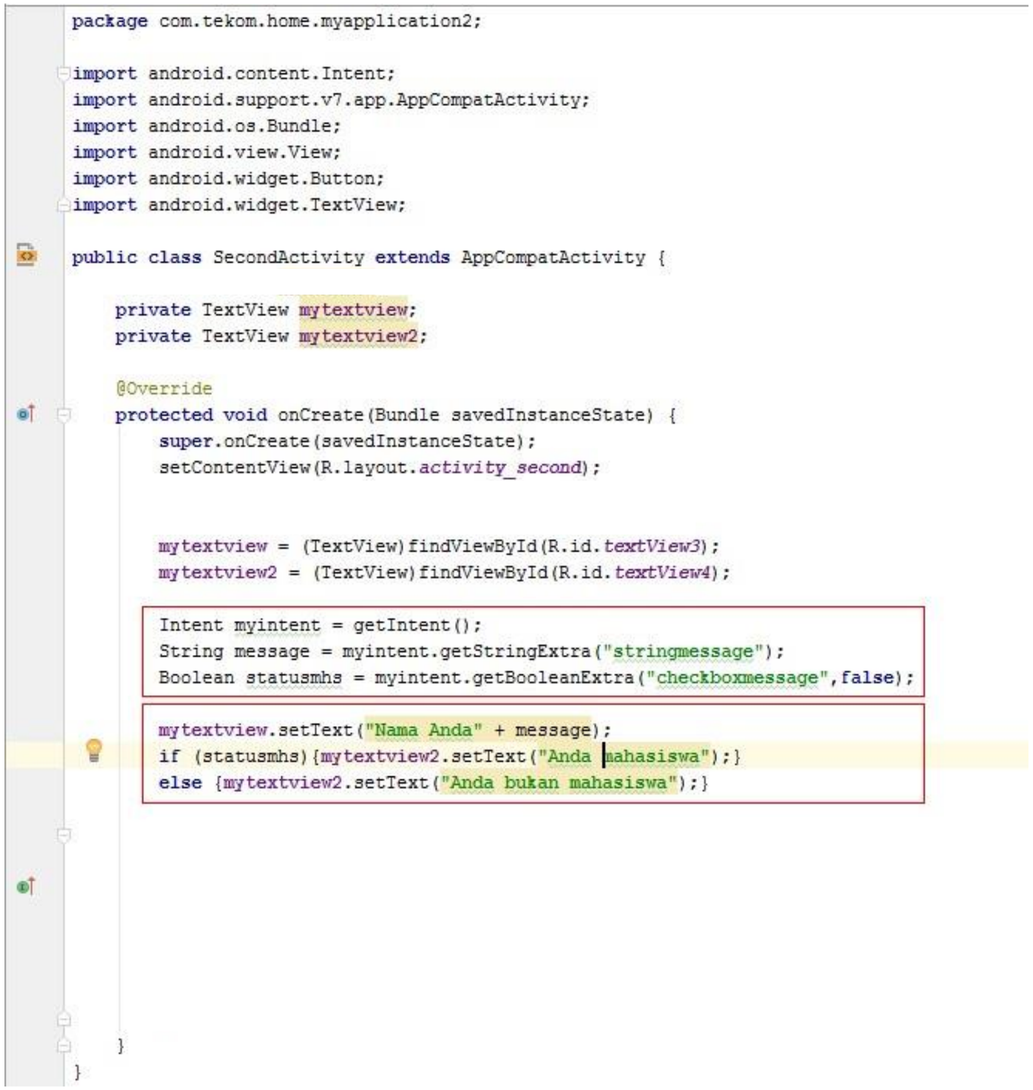
Pass the data using putExtra

- We will use the 'intent' to store important information
  Intents can have 'extras', arbitrary data included in the intent kind of like Bundles
- Extras are key-value pairs:
  public Intent putExtra(String name, Boolean value)



```
package com.tekom.home.myapplication2;

import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.CheckBox;
import android.widget.EditText;

public class MainActivity extends AppCompatActivity {

    private Button mybutton;
    private EditText myedittext;
    private CheckBox mycheckbox;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        mybutton = (Button)findViewById(R.id.button);
        myedittext = (EditText)findViewById(R.id.editText);
        mycheckbox = (CheckBox)findViewById(R.id.checkBox);

        mybutton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
            //Do something
                Intent myintent = new Intent(MainActivity.this,SecondActivity.class);

                String message = myedittext.getText().toString();
                myintent.putExtra("stringmessage",message);
                myintent.putExtra("checkboxmessage",mycheckbox.isChecked());

                startActivity(myintent);
            }
        });
    }
}
```

Still in the first activity, We will pass both data using putExtra

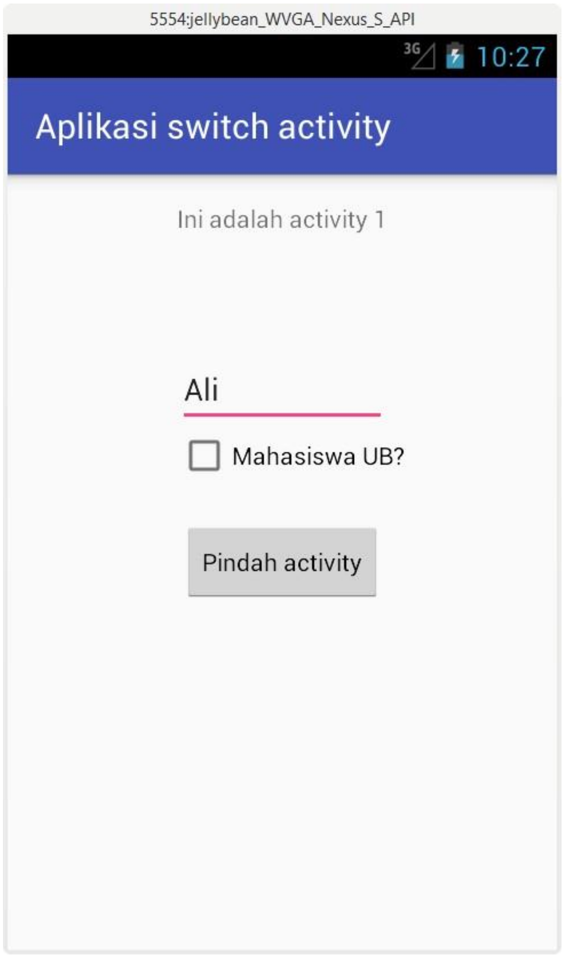


```
package com.tekom.home.myapplication2;

import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;

public class SecondActivity extends AppCompatActivity {

    private TextView mytextview;
    private TextView mytextview2;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_second);


        mytextview = (TextView)findViewById(R.id.textView3);
        mytextview2 = (TextView)findViewById(R.id.textView4);

        Intent myintent = getIntent();
        String message = myintent.getStringExtra("stringmessage");
        Boolean statusmhs = myintent.getBooleanExtra("checkboxmessage",false);

        mytextview.setText("Nama Anda" + message);
        if (statusmhs){mytextview2.setText("Anda mahasiswa");}
        else {mytextview2.setText("Anda bukan mahasiswa");}






    }
}
```

Then, in the second activity,
We will get the Intent using

- getStringExtra and
- getBooleanExtra

Result


5554:jellybean_WVGA_Nexus_S_API

10:27

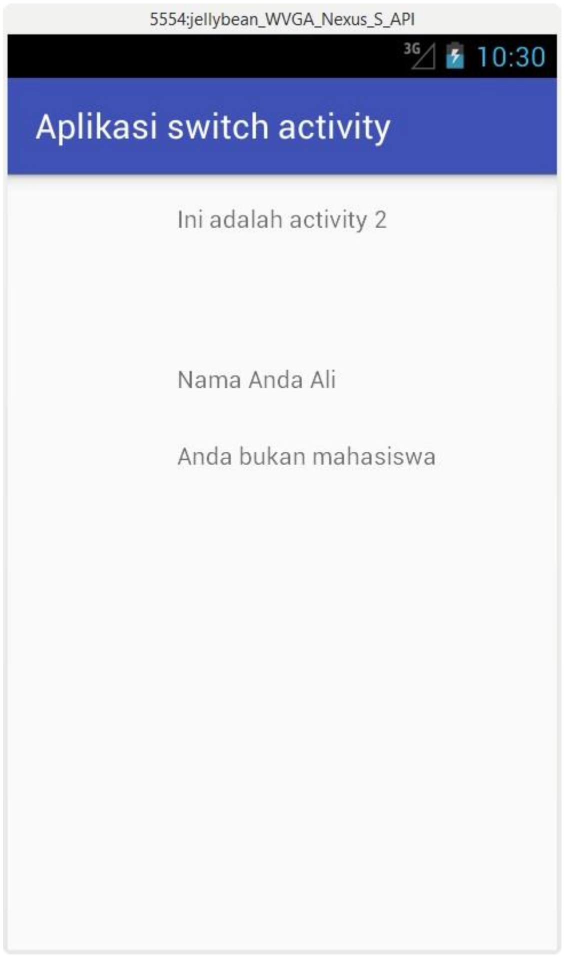
Aplikasi switch activity

Ini adalah activity 1

Ali

Mahasiswa UB?

Pindah activity


5554:jellybean_WVGA_Nexus_S_API

10:30

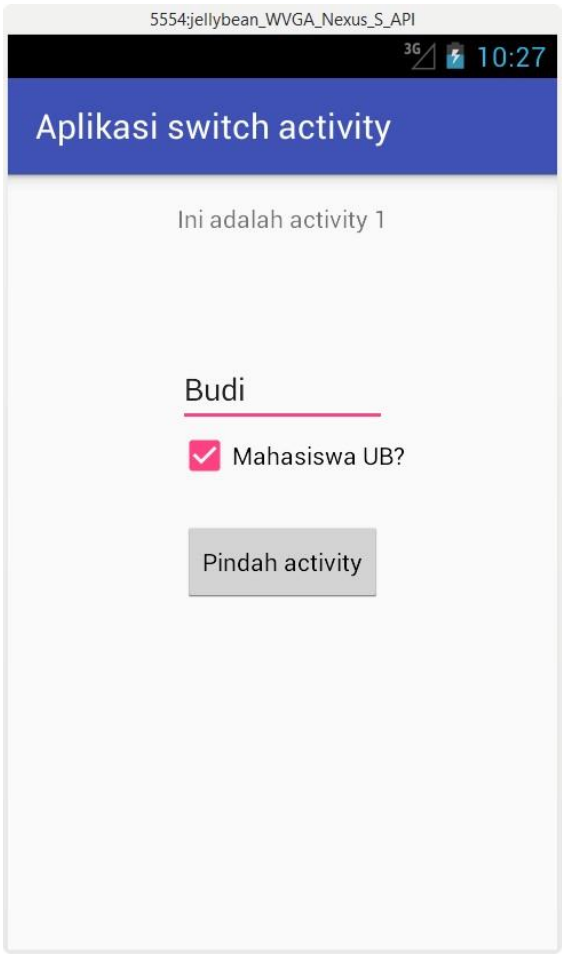
Aplikasi switch activity

Ini adalah activity 2

Nama Anda Ali

Anda bukan mahasiswa

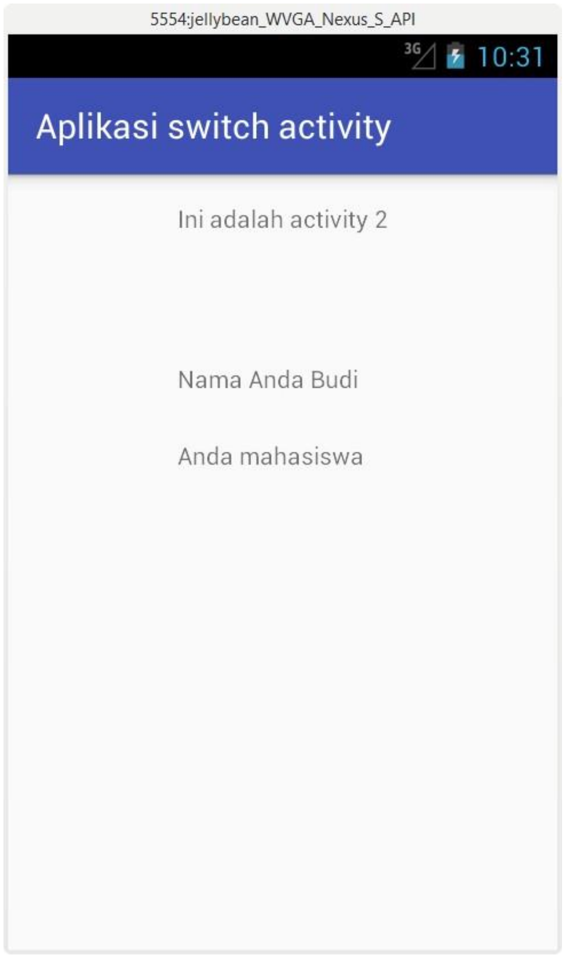
Result (cont.)


5554:jellybean_WVGA_Nexus_S_API

Ini adalah activity 1

Budi

☑ Mahasiswa UB?


5554:jellybean_WVGA_Nexus_S_API

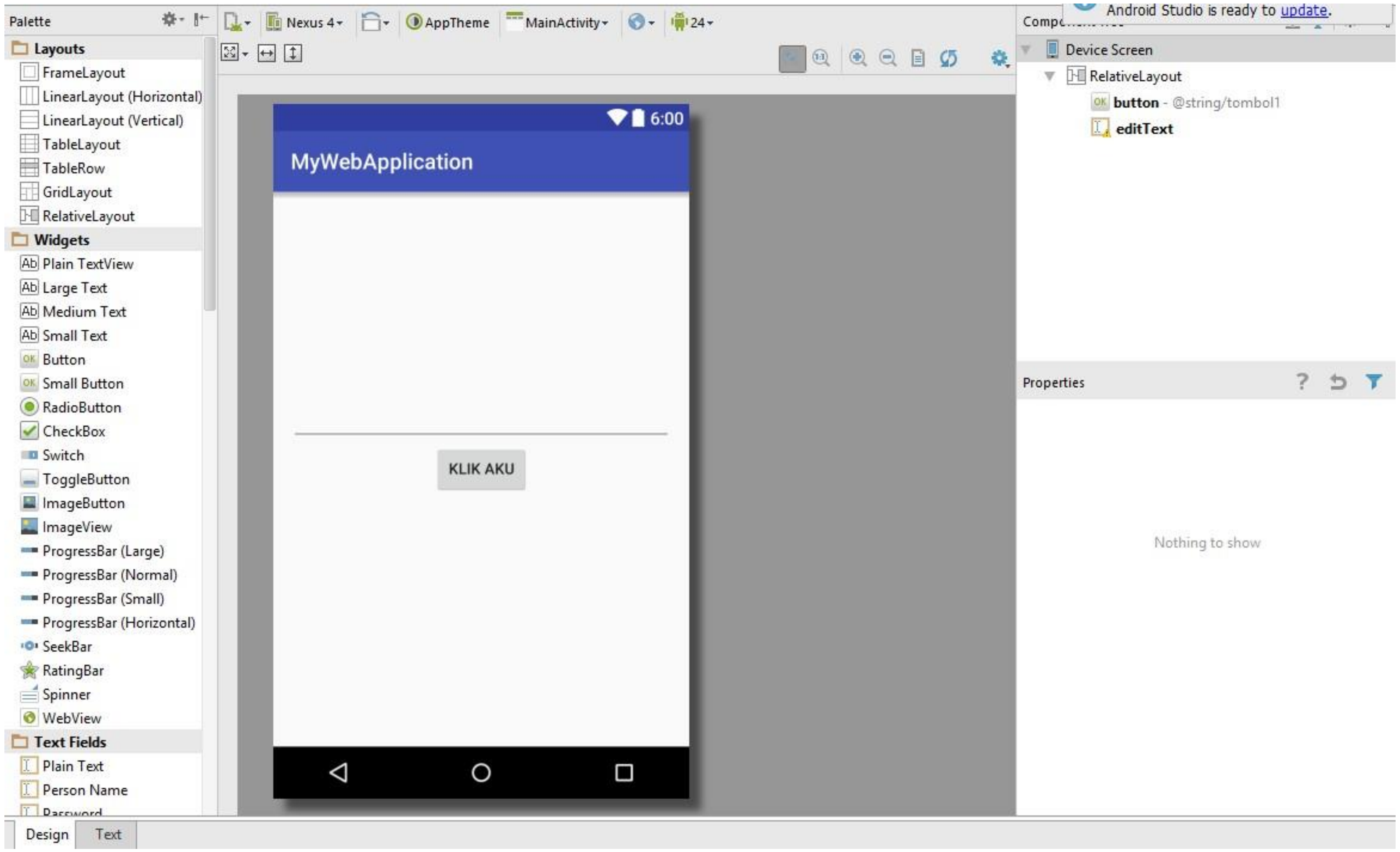
Ini adalah activity 2

Nama Anda Budi

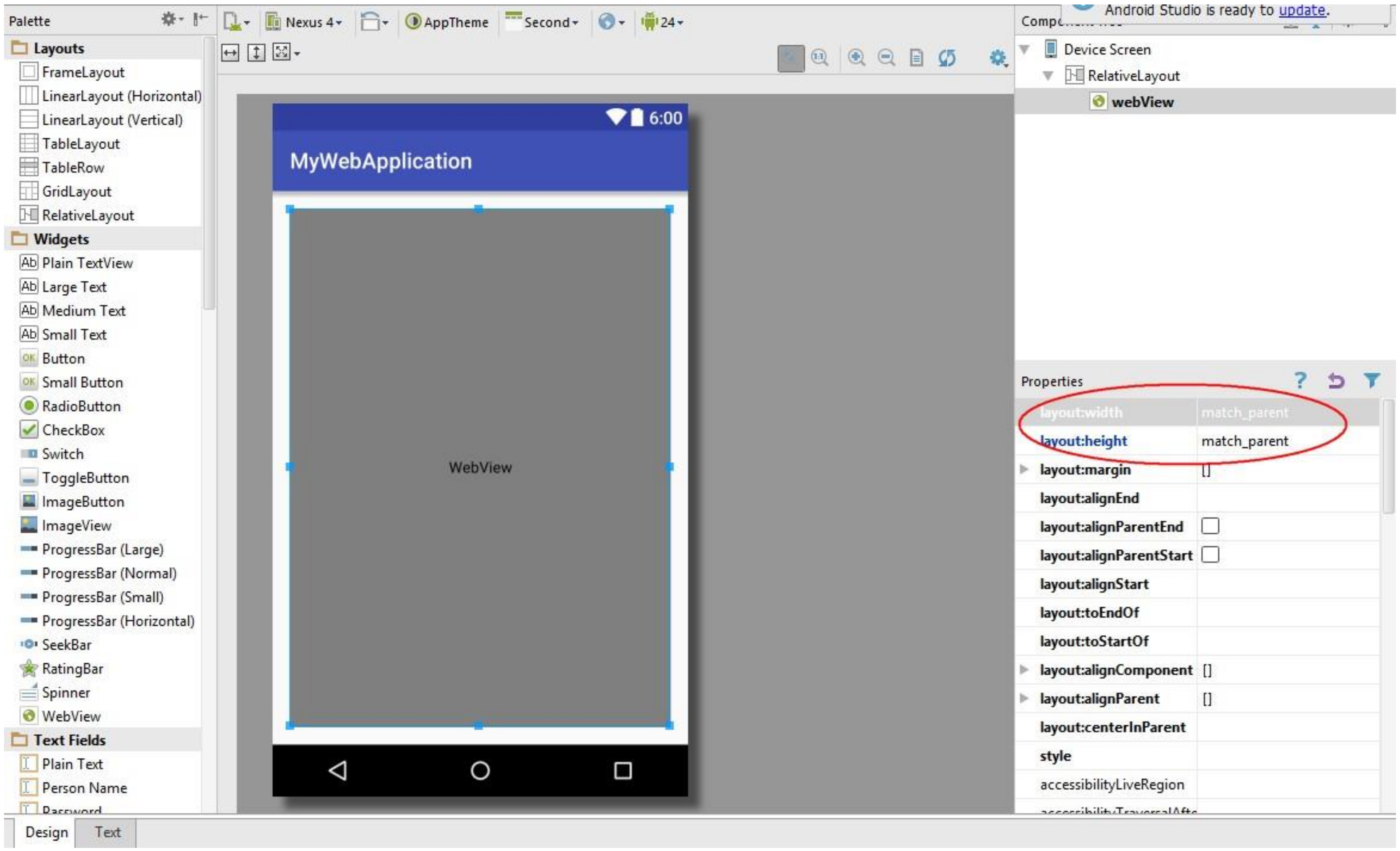
Anda mahasiswa

WEBVIEW

Let's try to make 2 activity. The first one will send the string containing URL to be opened in the second activity using WebView

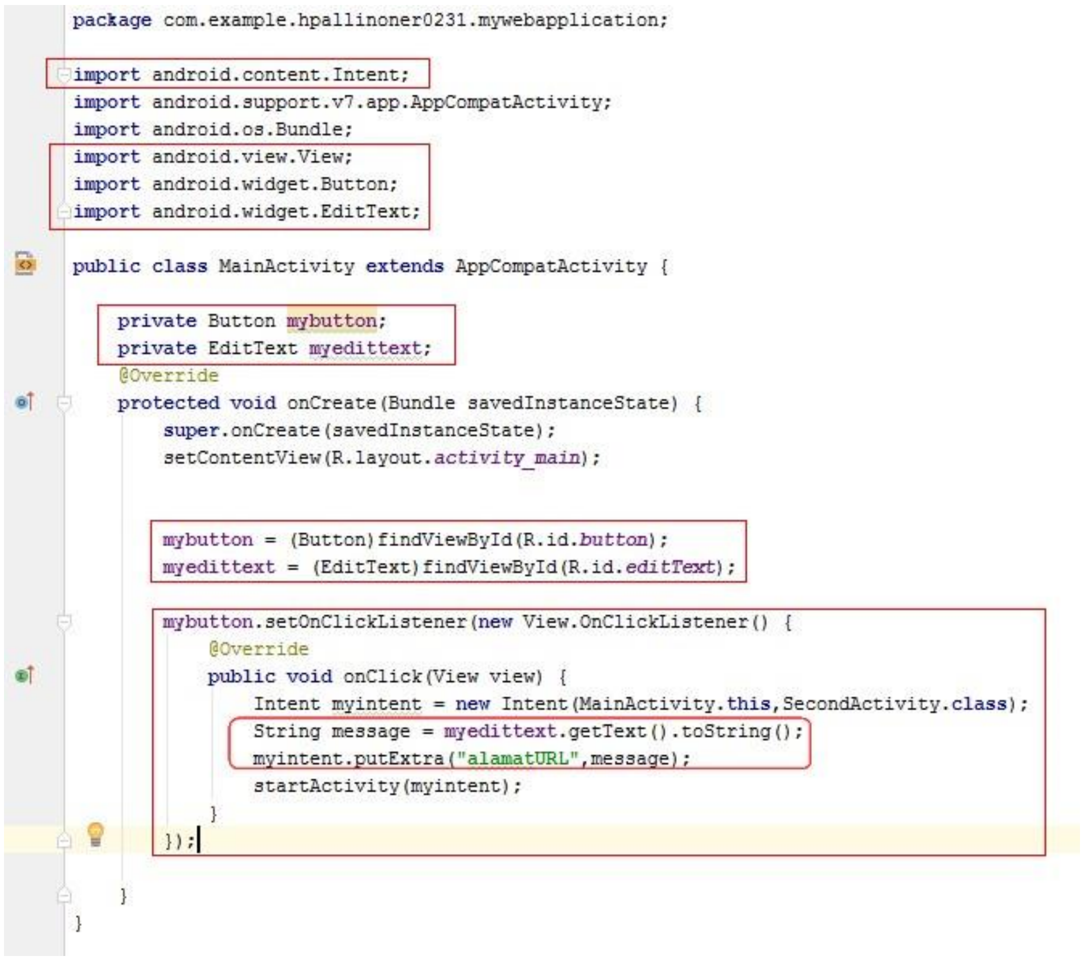


Add WebView widget in the second activity. Set the layout width and height to match_parent



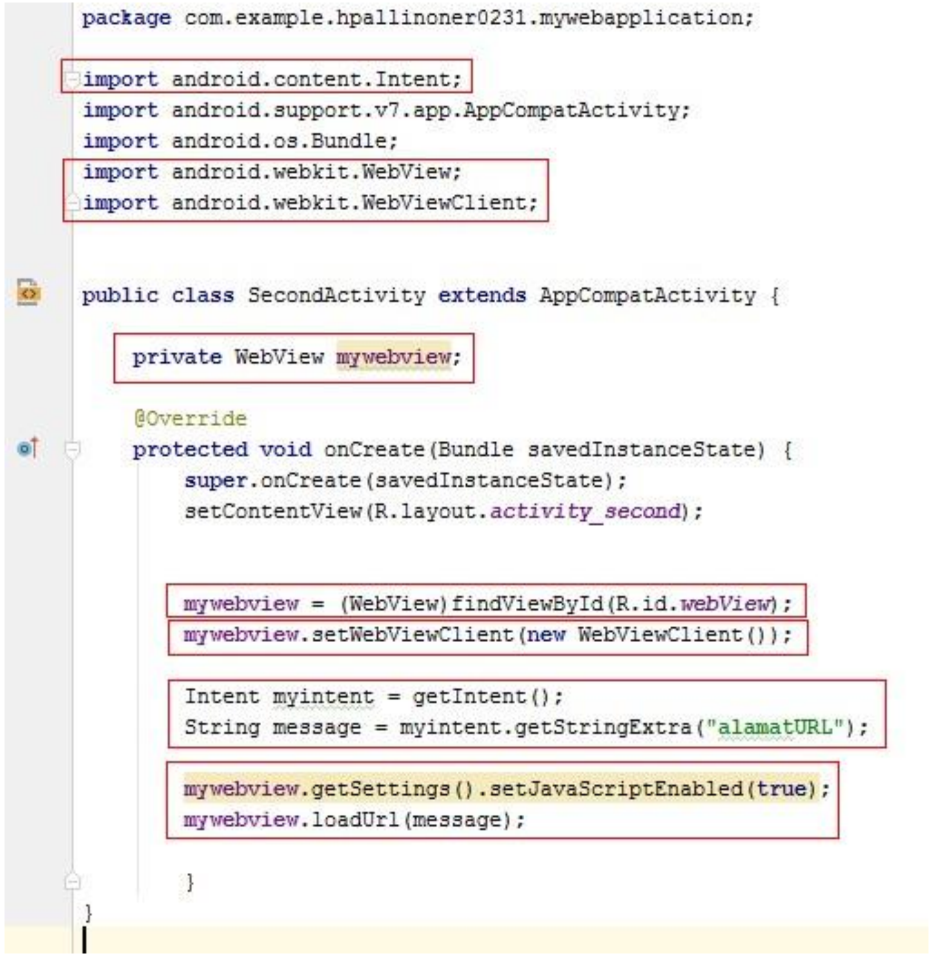
layout:width	match_parent
layout:height	match_parent
layout:margin	[]
layout:alignEnd	
layout:alignParentEnd	☐
layout:alignParentStart	☐
layout:alignStart	
layout:toEndOf	
layout:toStartOf	
layout:alignComponent	[]
layout:alignParent	[]
layout:centerInParent	
style	
accessibilityLiveRegion	

In the first activity, pass the URL string to second activity using intent putExtra



```
package com.example.hpallinoner0231.mywebapplication;

import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;

public class MainActivity extends AppCompatActivity {

    private Button mybutton;
    private EditText myedittext;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);


        mybutton = (Button)findViewById(R.id.button);
        myedittext = (EditText)findViewById(R.id.editText);

        mybutton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent myintent = new Intent(MainActivity.this,SecondActivity.class);
                String message = myedittext.getText().toString();
                myintent.putExtra("alamatURL",message);
                startActivity(myintent);
            }
        });

    }
}
```

Then in the second activity, get the intent using getStringExtra and load the url in the webView



```java
package com.example.hpallinoner0231.mywebapplication;

import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.webkit.WebView;
import android.webkit.WebViewClient;


public class SecondActivity extends AppCompatActivity {

    private WebView mywebview;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_second);


        mywebview = (WebView)findViewById(R.id.webView);
        mywebview.setWebViewClient(new WebViewClient());

        Intent myintent = getIntent();
        String message = myintent.getStringExtra("alamatURL");

        mywebview.getSettings().setJavaScriptEnabled(true);
        mywebview.loadUrl(message);

    }
}
```

Android Manifest

- **Every application must have a Manifest named AndroidManifest.xml**
- **Provides Android info about your app including:**
  - Unique app name (package)
  - Describes components in app
  - Permissions we need
  - Minimum API level

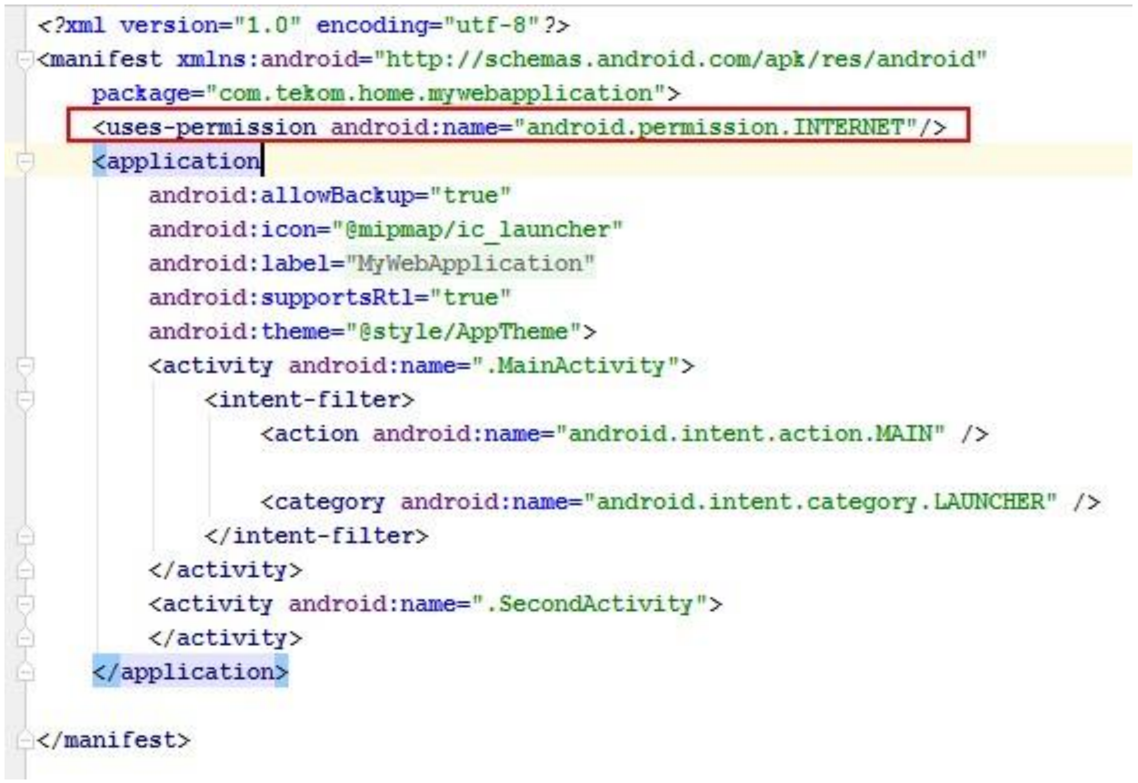
Permissions restricts access to code / data on a device

Android has its own host of permissions such as:

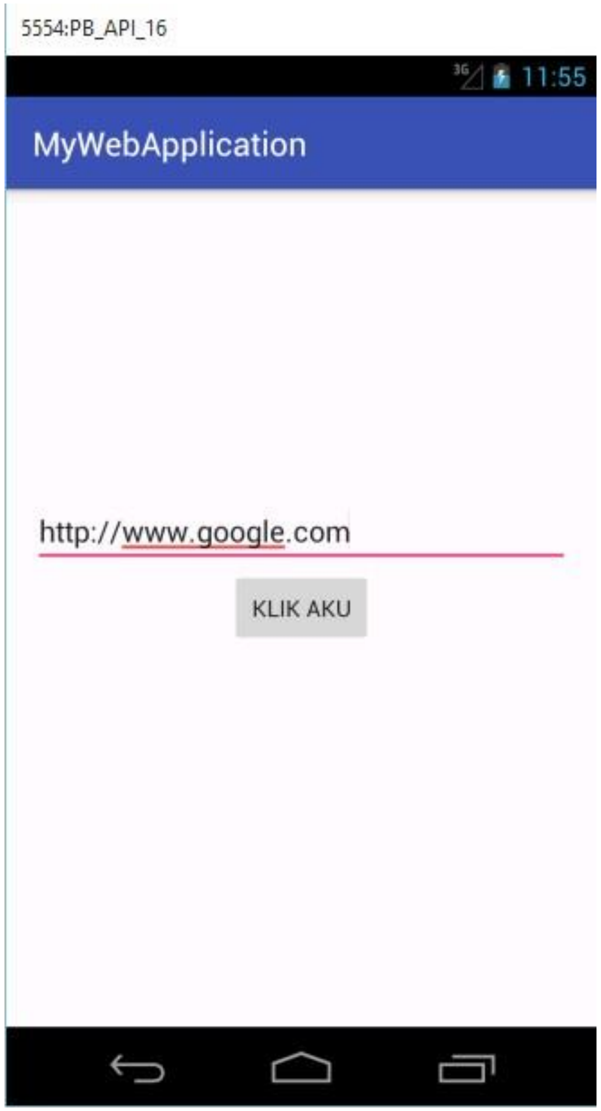
- Android.permission.CALL_EMERGENCY_NUMBERS
- Android.permission.READ_OWNER_DATA
- Android.permission.SET_WALLPAPER
- Android:permission.RECORD_AUDIO
- Android:permission.INTERNET
- android.permission.WRITE_EXTERNAL_STORAGE
- Android.permission.ACCESS_COARSE_LOCATION
- Android.permission.NFC
- …

We must declare a <uses-permission> element in the androidmanifest.xml if we want to use a protected feature

Add internet permission in Androidmanifest.xml



```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.tekom.home.mywebapplication">
    <uses-permission android:name="android.permission.INTERNET"/>
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="MyWebApplication"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity android:name=".SecondActivity">
        </activity>
    </application>

</manifest>
```

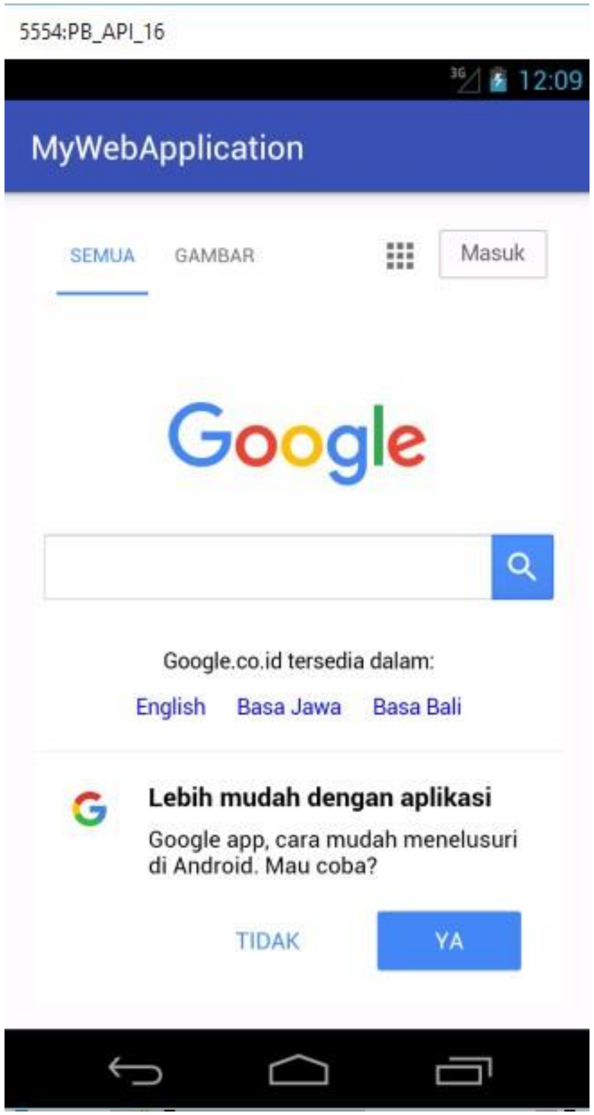

5554:PB_API_16

11:55

MyWebApplication

http://www.google.com

KLIK AKU


5554:PB_API_16

12:09

MyWebApplication

SEMUA GAMBAR Masuk

Google

Google.co.id tersedia dalam:

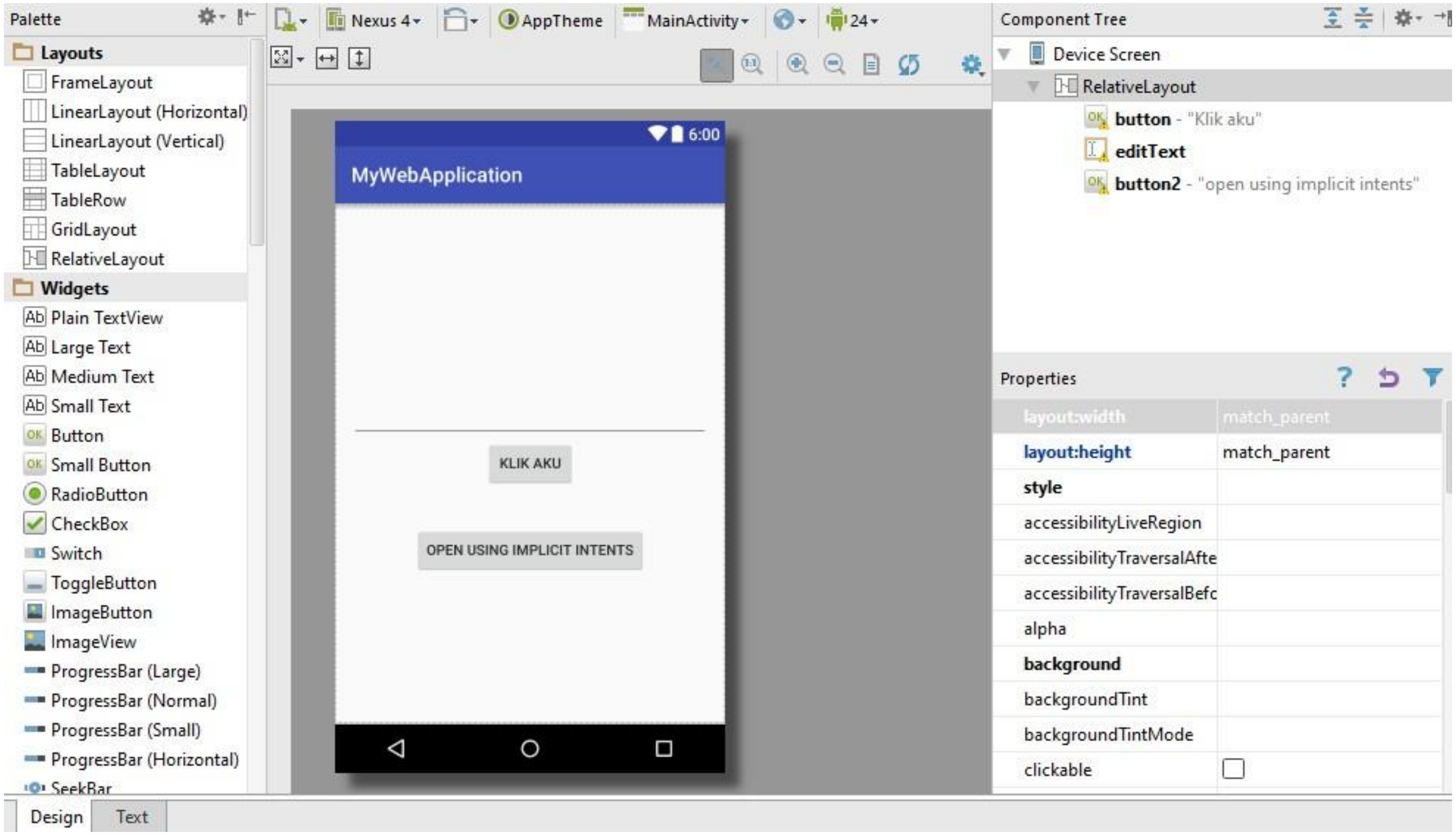
English Basa Jawa Basa Bali

Lebih mudah dengan aplikasi

Google app, cara mudah menelusuri di Android. Mau coba?

TIDAK YA

Open the link using another apps?



Component Tree
- Device Screen
  - RelativeLayout
    - button - "Klik aku"
    - editText
    - button2 - "open using implicit intents"

Properties

layout:width	match_parent
layout:height	match_parent
style	
accessibilityLiveRegion	
accessibilityTraversalAfte	
accessibilityTraversalBefc	
alpha	
background	
backgroundTint	
backgroundTintMode	
clickable	☐

Implicit intents

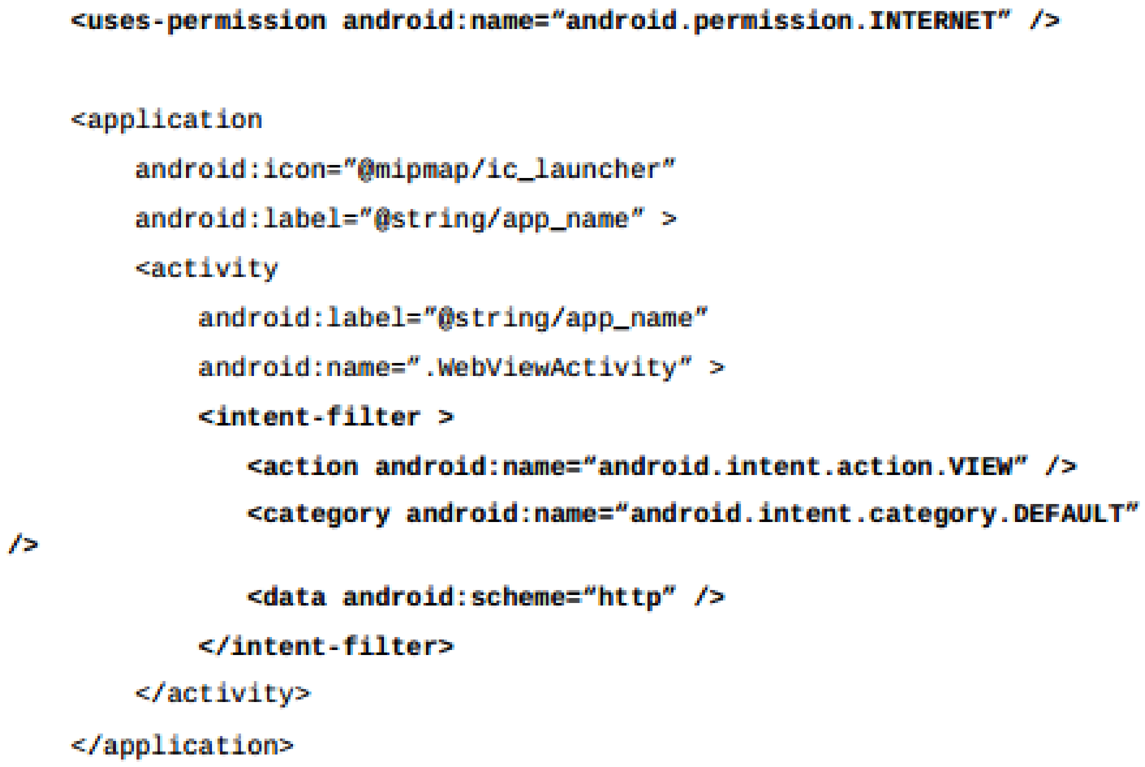
- Implicit means we do not directly specify the Android component to be activated
- We only specify the action to be performed and the type of data to be handled
- This is done through an different intent constructor:

Intent intent_name = new Intent(Intent.ACTION_SEND);

```
Intent i = new Intent(Intent.ACTION_VIEW,
                      Uri.parse("http://www.ebookfrenzy.com"));
```

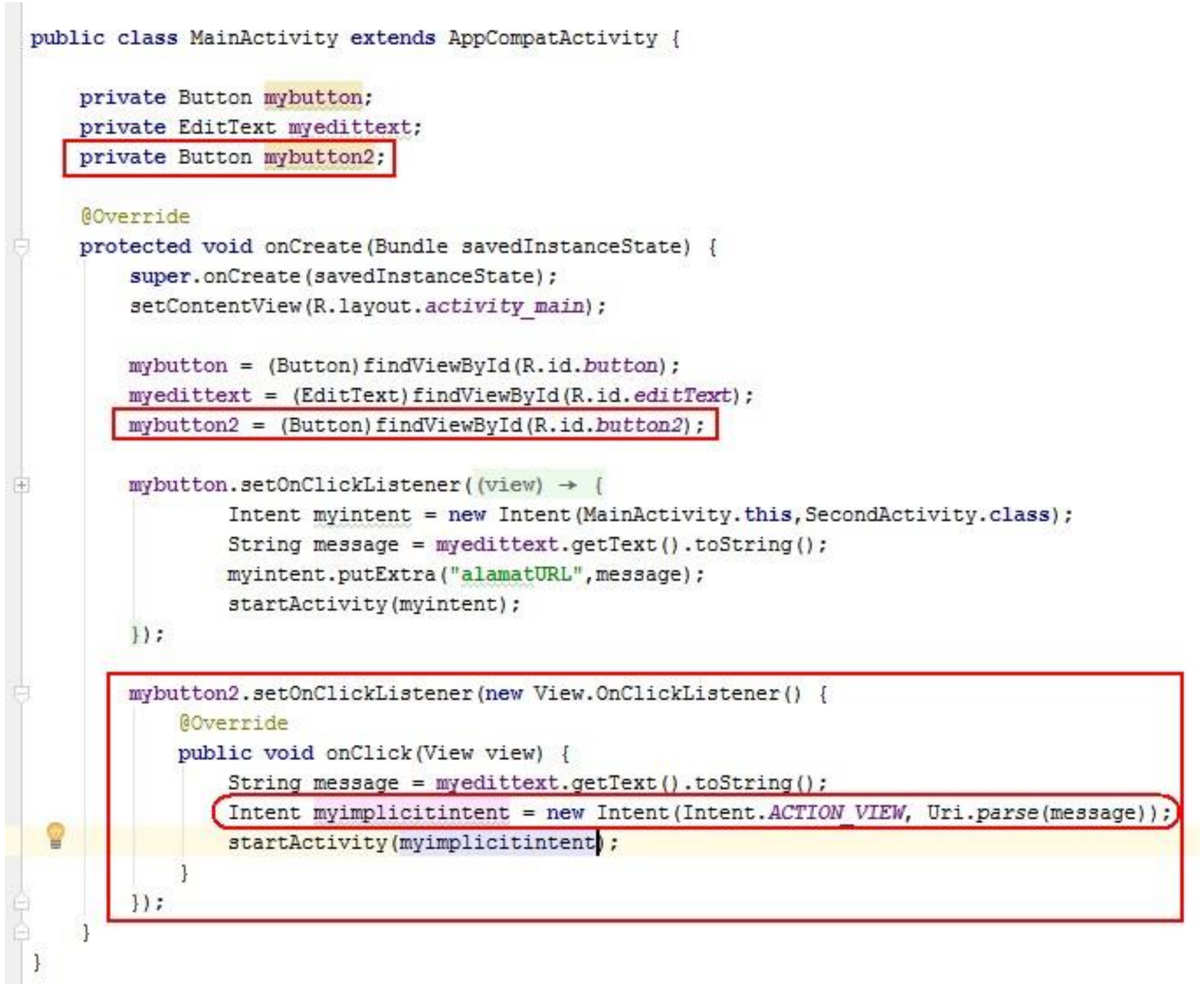
- ❖ When the above implicit intent is issued by an activity, the Android system will search for activities on the device that have registered the ability to handle ACTION_VIEW requests on *http* scheme data
- ❖ In the event that a single match is found, that activity will be launched.
- ❖ If more than one match is found, the user will be prompted to choose from the available activity options.

Intent filters are the mechanism by which activities “advertise” supported actions and data handling capabilities to the Android intent resolution process.

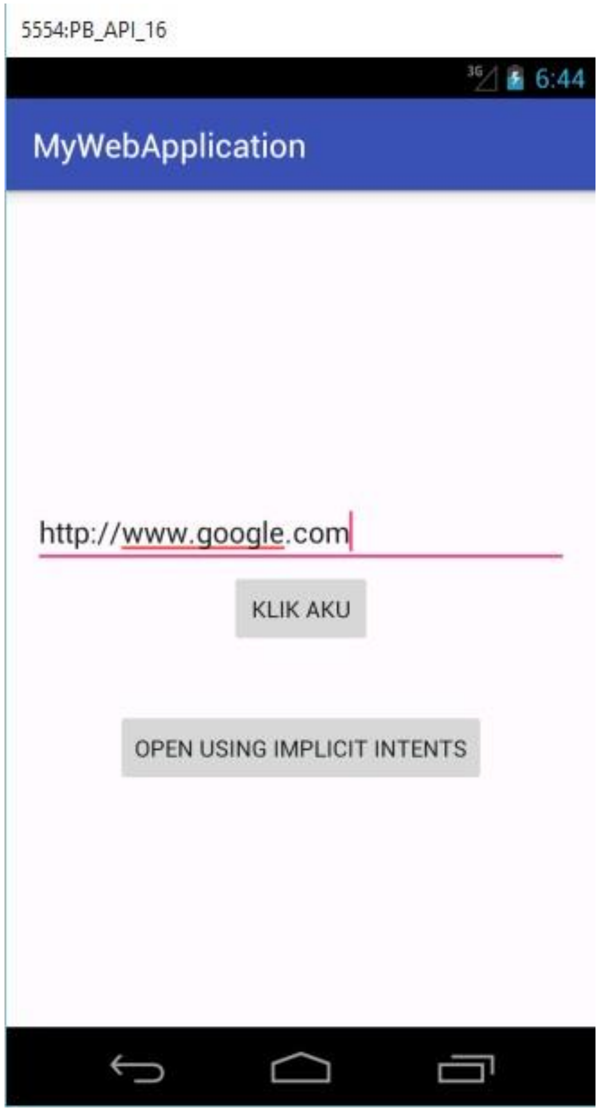


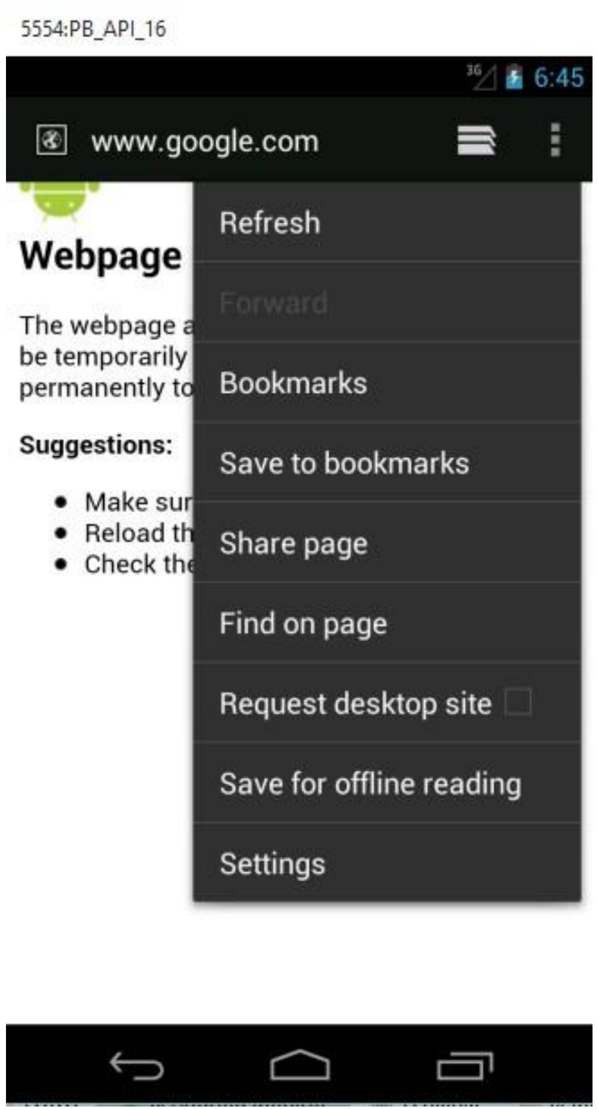
```
        <uses-permission android:name="android.permission.INTERNET" />

        <application
            android:icon="@mipmap/ic_launcher"
            android:label="@string/app_name" >
            <activity
                android:label="@string/app_name"
                android:name=".WebViewActivity" >
                <intent-filter >
                   <action android:name="android.intent.action.VIEW" />
                   <category android:name="android.intent.category.DEFAULT"
    />
                   <data android:scheme="http" />
                </intent-filter>
            </activity>
        </application>
```

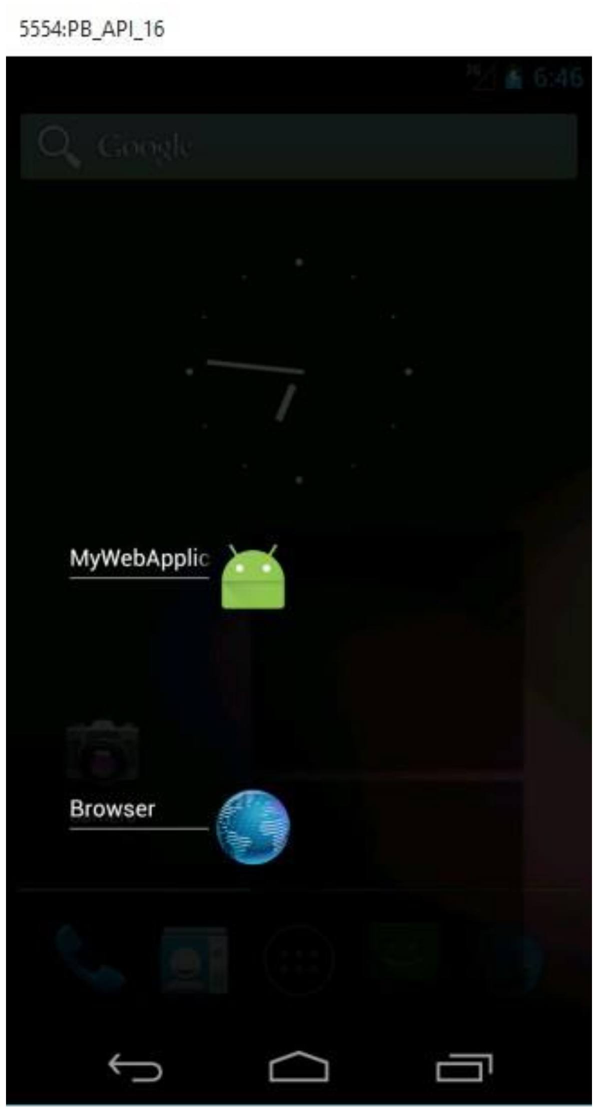
Androidmanifest.xml



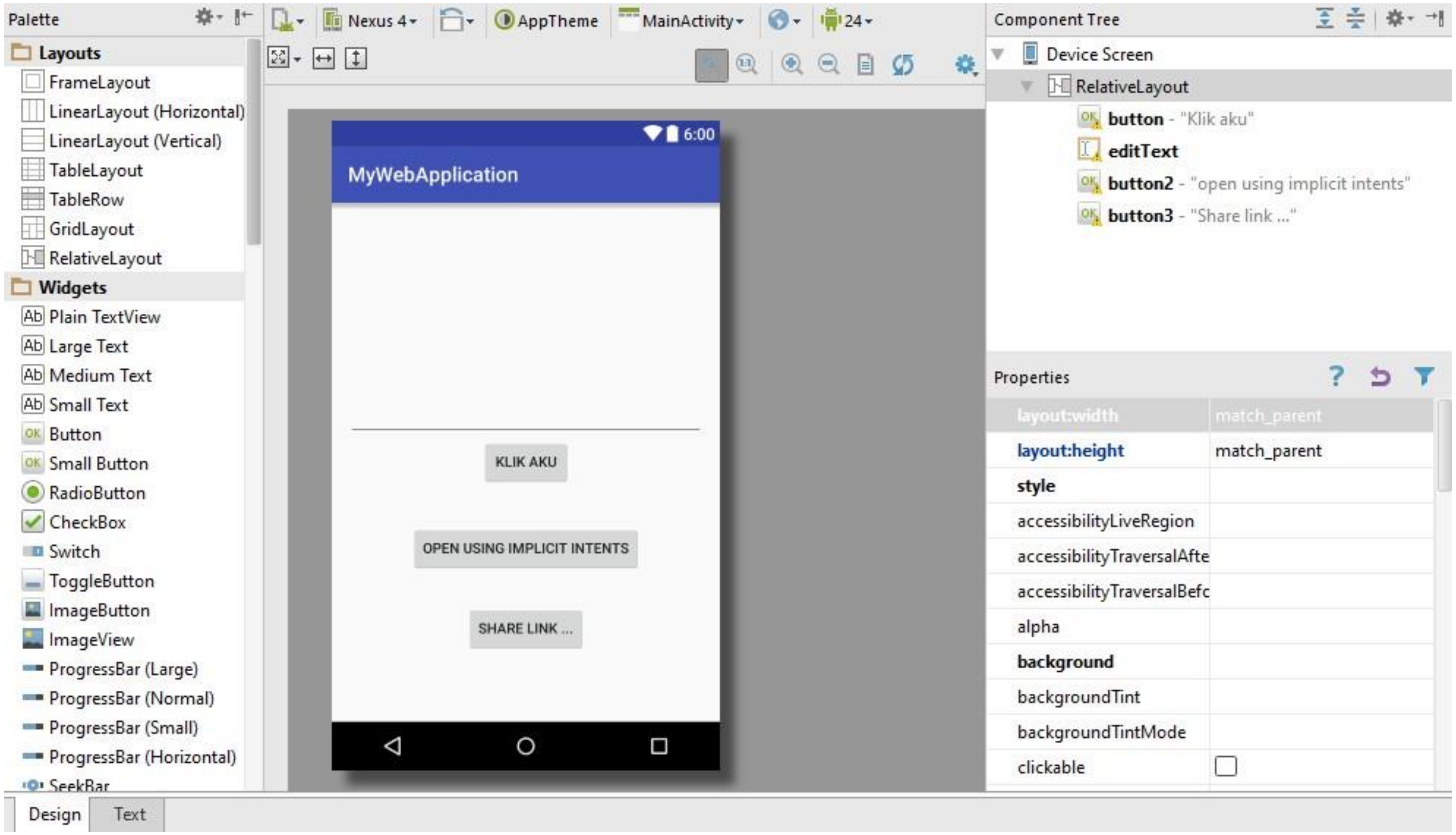
```
public class MainActivity extends AppCompatActivity {

    private Button mybutton;
    private EditText myedittext;
    private Button mybutton2;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        mybutton = (Button)findViewById(R.id.button);
        myedittext = (EditText)findViewById(R.id.editText);
        mybutton2 = (Button)findViewById(R.id.button2);

        mybutton.setOnClickListener((view) → {
                Intent myintent = new Intent(MainActivity.this,SecondActivity.class);
                String message = myedittext.getText().toString();
                myintent.putExtra("alamatURL",message);
                startActivity(myintent);
        });

        mybutton2.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                String message = myedittext.getText().toString();
                Intent myimplicitintent = new Intent(Intent.ACTION_VIEW, Uri.parse(message));
                startActivity(myimplicitintent);
            }
        });
    }
}
```


5554:PB_API_16
6:44
MyWebApplication
http://www.google.com
KLIK AKU
OPEN USING IMPLICIT INTENTS


5554:PB_API_16
6:45
www.google.com
Webpage
The webpage a
be temporarily
permanently to
Suggestions:
Make sur
Reload th
Check the
Refresh
Forward
Bookmarks
Save to bookmarks
Share page
Find on page
Request desktop site
Save for offline reading
Settings

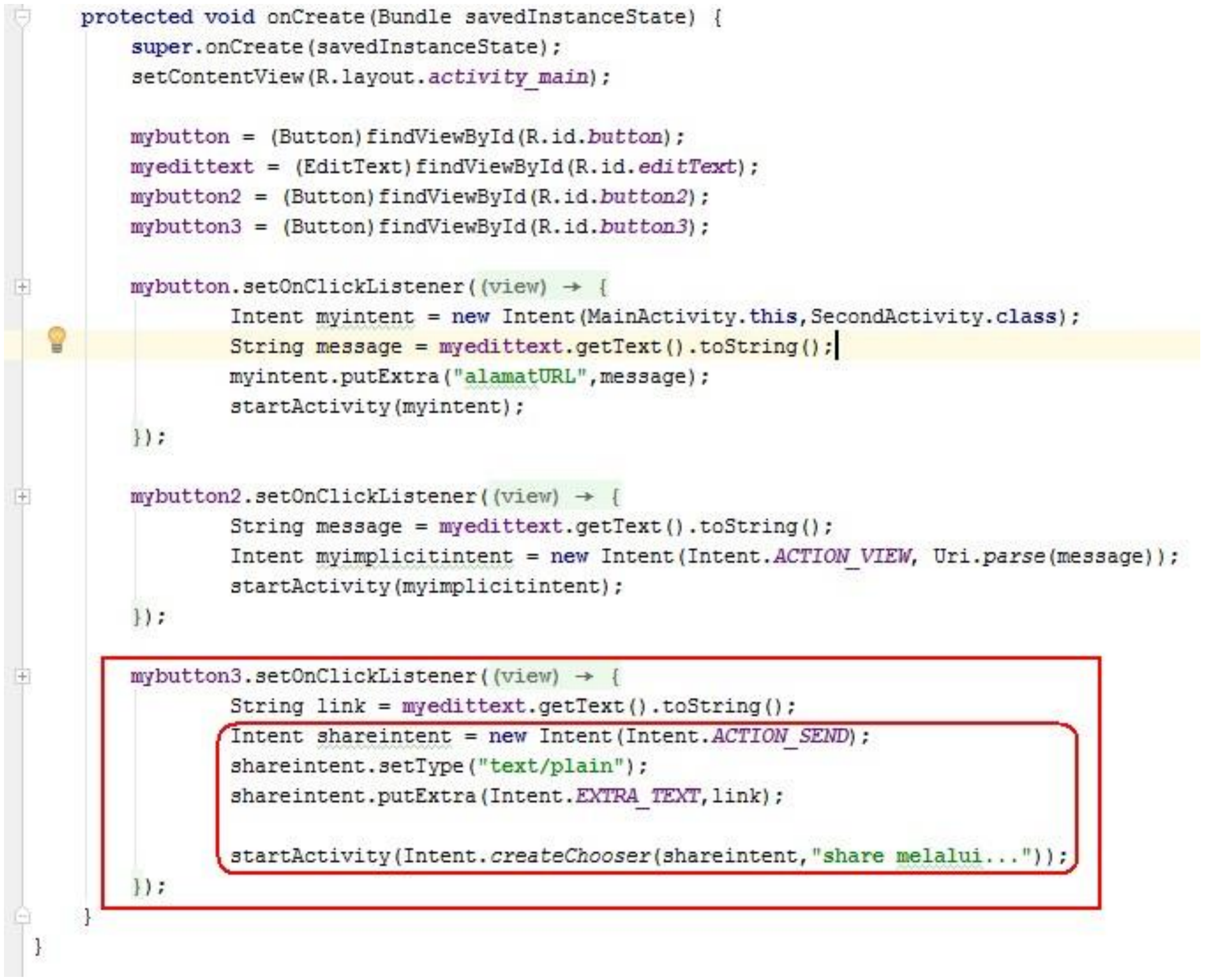

5554:PB_API_16
6:46
Google
MyWebApplic
Browser

Implicit intents-sharing

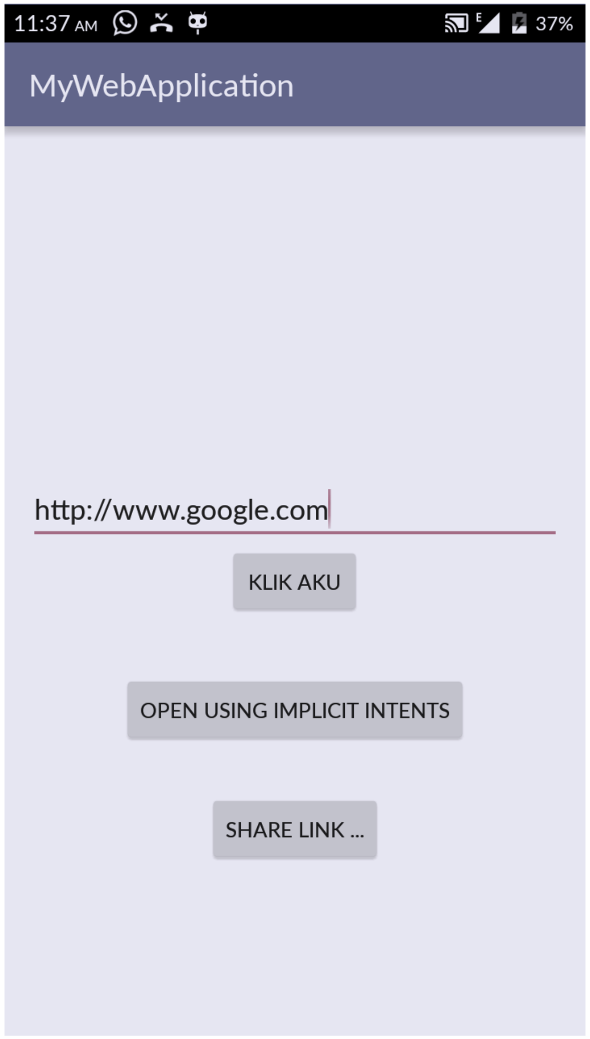


Device Screen
RelativeLayout
button - "Klik aku"
editText
button2 - "open using implicit intents"
button3 - "Share link ..."

layout:width	match_parent
layout:height	match_parent
style	
accessibilityLiveRegion	
accessibilityTraversalAfte	
accessibilityTraversalBefc	
alpha	
background	
backgroundTint	
backgroundTintMode	
clickable	☐



```
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        mybutton = (Button)findViewById(R.id.button);
        myedittext = (EditText)findViewById(R.id.editText);
        mybutton2 = (Button)findViewById(R.id.button2);
        mybutton3 = (Button)findViewById(R.id.button3);

        mybutton.setOnClickListener((view) → {
                Intent myintent = new Intent(MainActivity.this,SecondActivity.class);
                String message = myedittext.getText().toString();
                myintent.putExtra("alamatURL",message);
                startActivity(myintent);
        });

        mybutton2.setOnClickListener((view) → {
                String message = myedittext.getText().toString();
                Intent myimplicitintent = new Intent(Intent.ACTION_VIEW, Uri.parse(message));
                startActivity(myimplicitintent);
        });

        mybutton3.setOnClickListener((view) → {
                String link = myedittext.getText().toString();
                Intent shareintent = new Intent(Intent.ACTION_SEND);
                shareintent.setType("text/plain");
                shareintent.putExtra(Intent.EXTRA_TEXT,link);

                startActivity(Intent.createChooser(shareintent,"share melalui..."));
        });
    }
}
```


11:37 AM 37%

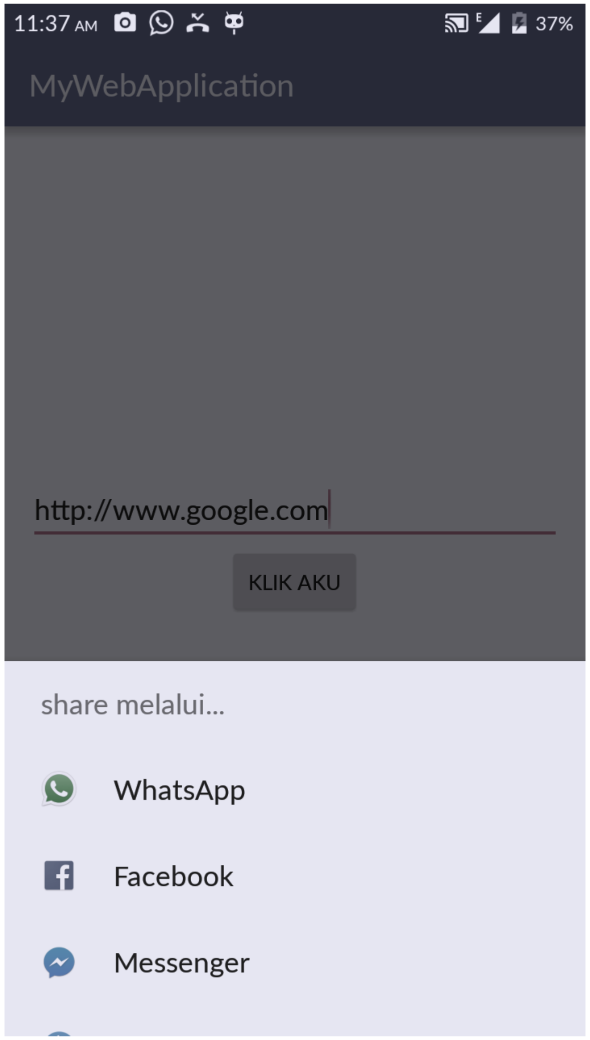
MyWebApplication

http://www.google.com

KLIK AKU

OPEN USING IMPLICIT INTENTS

SHARE LINK ...


11:37 AM 37%

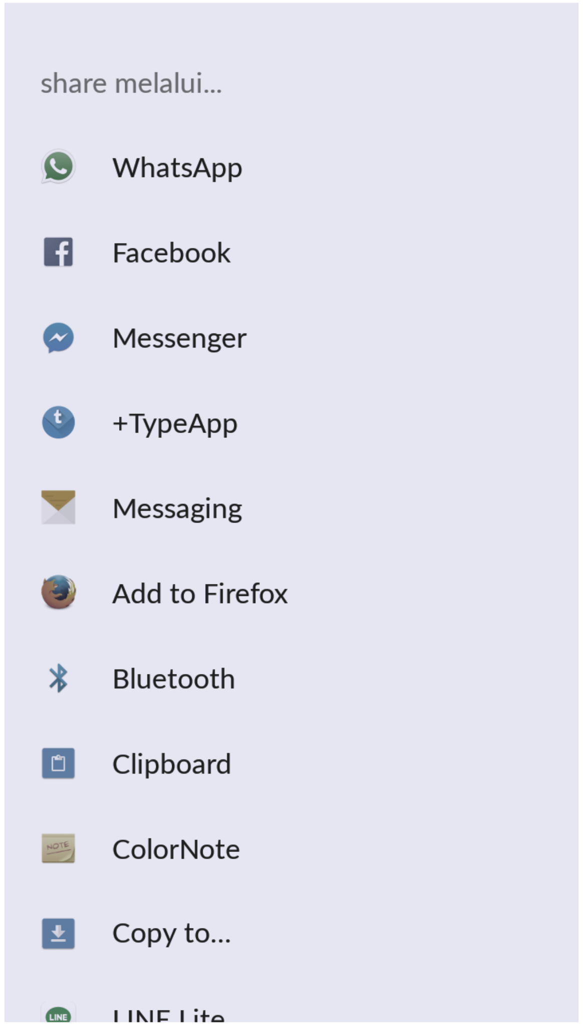
MyWebApplication

http://www.google.com

KLIK AKU

share melalui...

- WhatsApp
- Facebook
- Messenger


share melalui...

- WhatsApp
- Facebook
- Messenger
- +TypeApp
- Messaging
- Add to Firefox
- Bluetooth
- Clipboard
- ColorNote
- Copy to...
- LINE Lite

More on share contents using intents:

Guides.codepath.com/android/Sharing-Content-with-Intents

- Sharing HTML
- Sharing images
- Sharing links
- Share in facebook
- Sharing multiple types

TERIMA KASIH