

Outline of the lab 3:

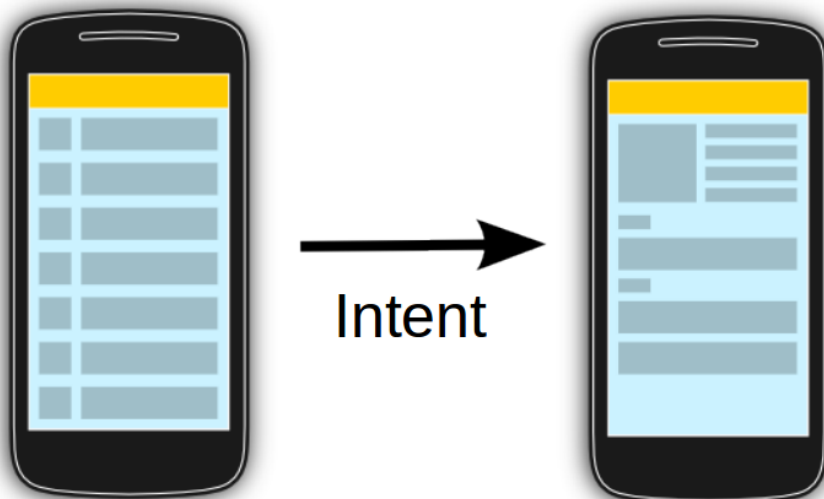
- ✓ Task 1: Simple explicit intent
- ✓ Task 2: understanding activity lifecycle in Android
- ✓ Task 3: using explicit intent for lab 1
- ✓ Task 4: AES encryption of messages

Objective:

This lab will provide you with the knowledge about explicit and implicit intent in Android and how to use them to transfer between activities. The knowledge of cryptography in general and various kinds of encryptions will be introduced. Subsequently, you will program an app that uses AES encrypted algorithm to encrypt the message.

Task 1. Simple explicit intent:

Transferring between activities is very popular in Android. Implicit and explicit intents are two kinds of making intent. The below is modelling the process of making new intent. Simply, this is the way we call new activity.

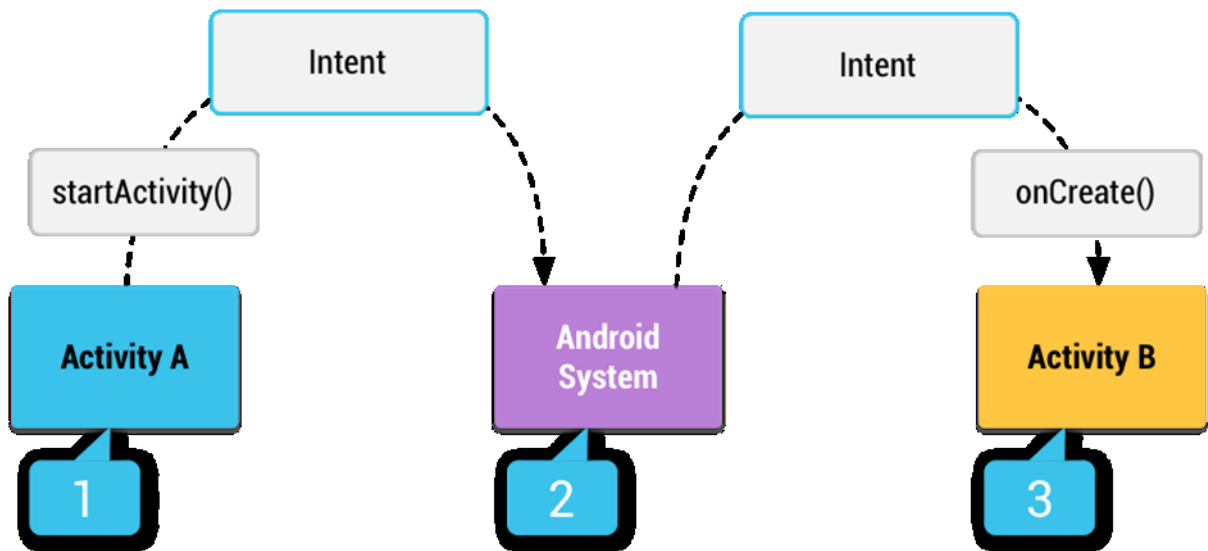


You can refer below links to receive more information about intent.

<https://developer.android.com/guide/components/intents-filters>

<https://www.vogella.com/tutorials/AndroidIntent/article.html>

The model below provides with the process of making the new intent from activity A to Activity B.



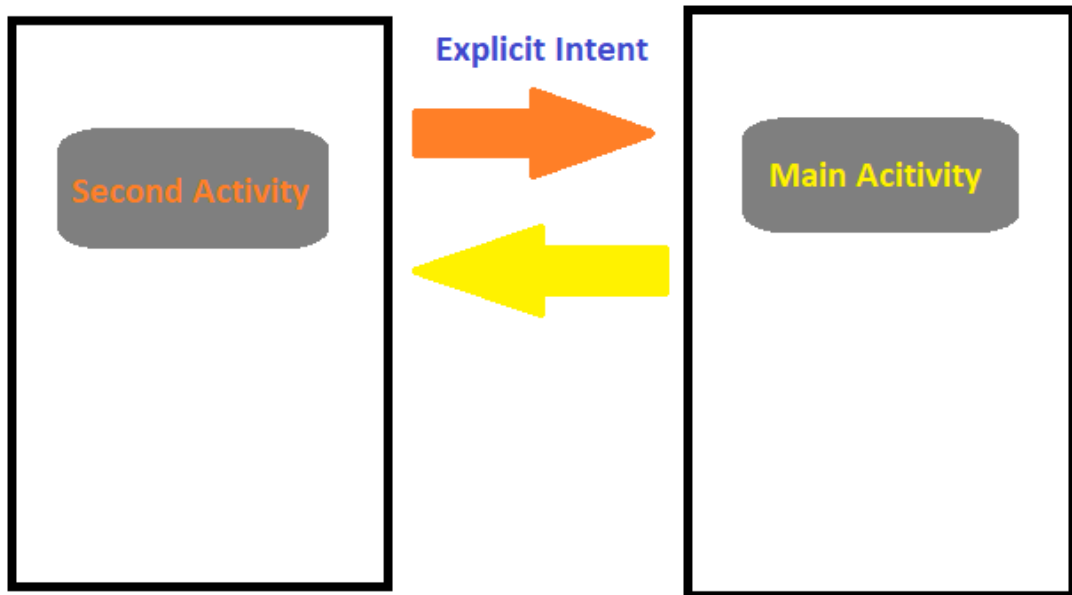
To make the new intent, activity A call the function startActivity(). Then android system will make the new Intent by start the onCreate() the activity B. as a result, activity B will create and start.

The following code demonstrates how you can start another activity via an intent

```
// Start the activity connect to the  
// specified class  
  
Intent i = new Intent(this, ActivityTwo.class);  
startActivity(i);
```

The task is to make 2 activities (two layouts and two java classes) as seen in the design below. Activity 1 (layout 1) has a button (Second Activity). When you click on this button. Then the activity 2 appear. The activity 2 has a button (Main Activity). When you click on this button, activity 1 will appear.

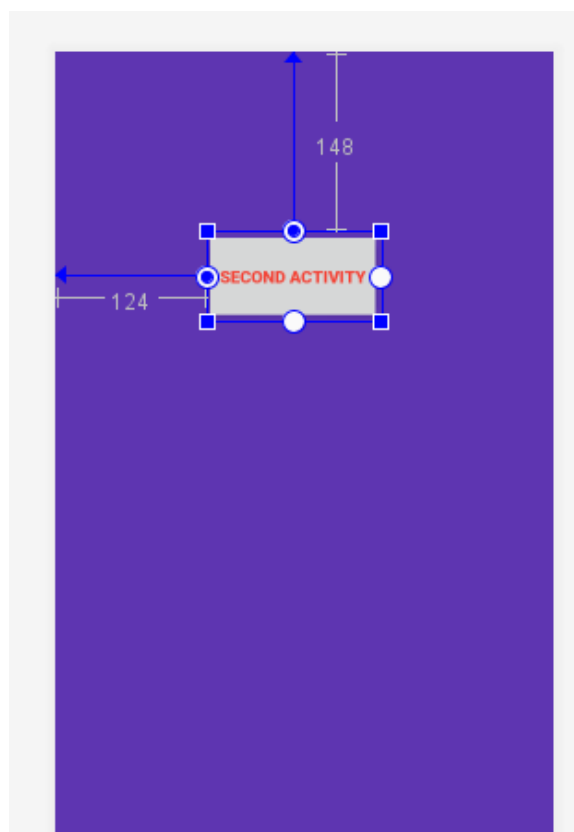
42036 - Cyber Security for Mobile Platforms
Tutorial 3 manual (Week 3): Intent and Encryption



Make a Project name: Explicit Intent.

Design activity_main.xml

Background: Violet for example and Button with size bigger and color of the button.



42036 - Cyber Security for Mobile Platforms

Tutorial 3 manual (Week 3): Intent and Encryption

Xml here

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#5E35B1"
    tools:context=".MainActivity">
    <Button
        android:id="@+id/bttFirst"
        android:layout_width="wrap_content"
        android:layout_height="75dp"
        android:layout_marginStart="124dp"
        android:layout_marginLeft="124dp"
        android:layout_marginTop="148dp"
        android:text="@string/bt_name"
        android:textColor="#F44336"
        android:textSize="14sp"
        android:textStyle="bold"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

String value for button

```
<resources>
    <string name="app_name">Explicit Intent</string>
    <string name="bt_name">Second Activity</string>
    <string name="btt_name2">Main Activity</string>
</resources>
```

Code for java file MainActivity.java

```
package com.example.explicitintent;
import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
public class MainActivity extends Activity {
    //declare the button for main activity
    Button btnMain;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        //catch from file R
        btnMain=(Button) findViewById(R.id.bttFirst);
    }
}
```

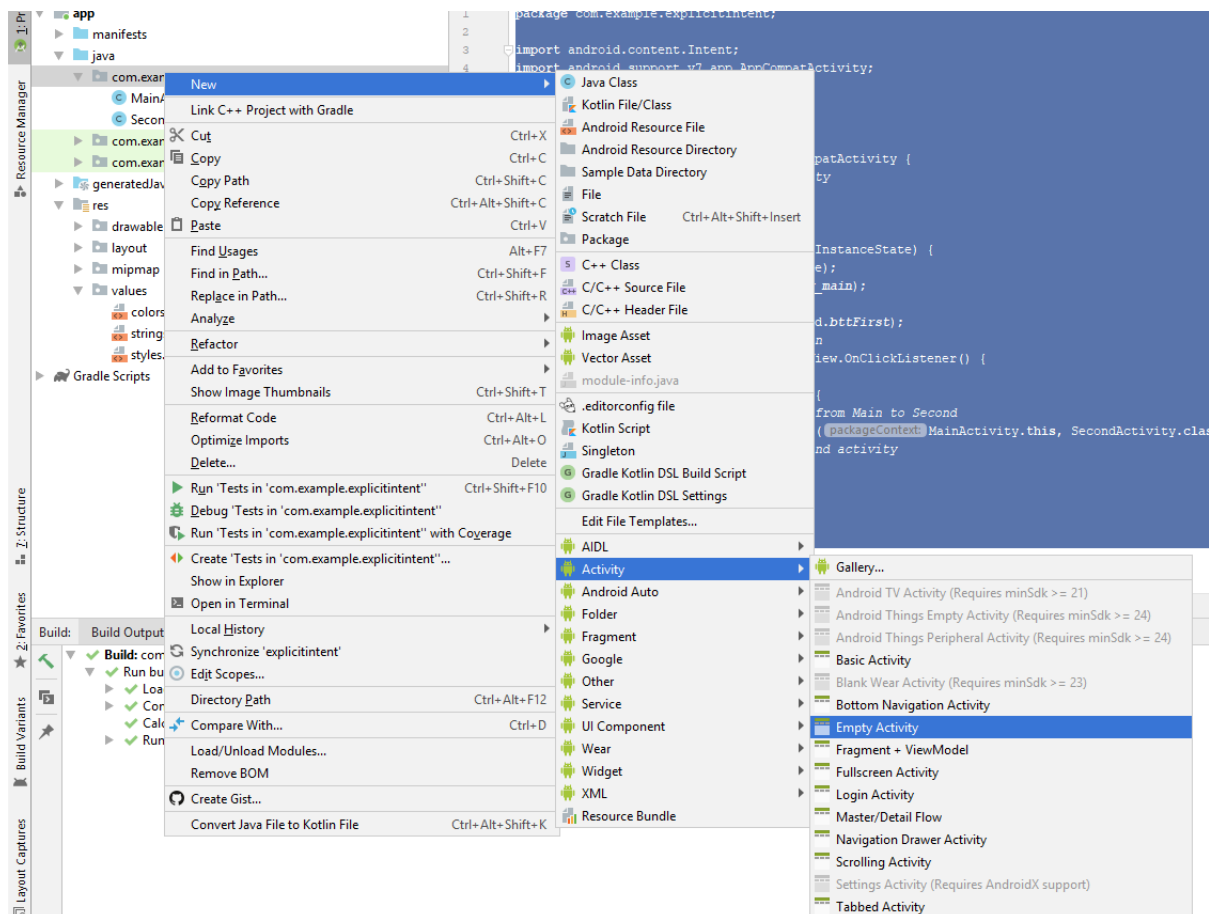
42036 - Cyber Security for Mobile Platforms

Tutorial 3 manual (Week 3): Intent and Encryption

```
// Catch when we click the button
btnMain.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        // Declare a new intent from Main to Second
        Intent intent=new Intent(MainActivity.this,
        SecondActivity.class);
        // Start to run the second activity
        startActivity(intent);
    }
});
}
```

For second activity do follows:

Right click on the package name >> New >> Activity >> Empty Activity



Then set the name for SecondActivity, automatically generating activity_second.xml

42036 - Cyber Security for Mobile Platforms

Tutorial 3 manual (Week 3): Intent and Encryption

Background: Yellow for example and Button with size bigger and color of the button.

Code for activity_second

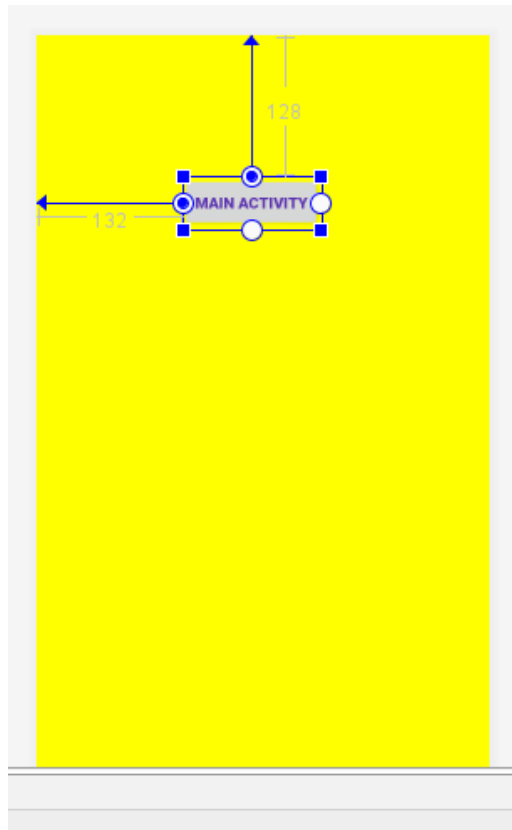
```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"

    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#fff0"
    tools:context=".SecondActivity">

    <Button
        android:id="@+id/bttSecond"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="132dp"
        android:layout_marginLeft="132dp"
        android:layout_marginTop="128dp"
        android:text="@string/btt_name2"
        android:textColor="#5E35B1"
        android:textColorHint="#C0CA33"
        android:textStyle="bold"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

42036 - Cyber Security for Mobile Platforms

Tutorial 3 manual (Week 3): Intent and Encryption



Code for java file

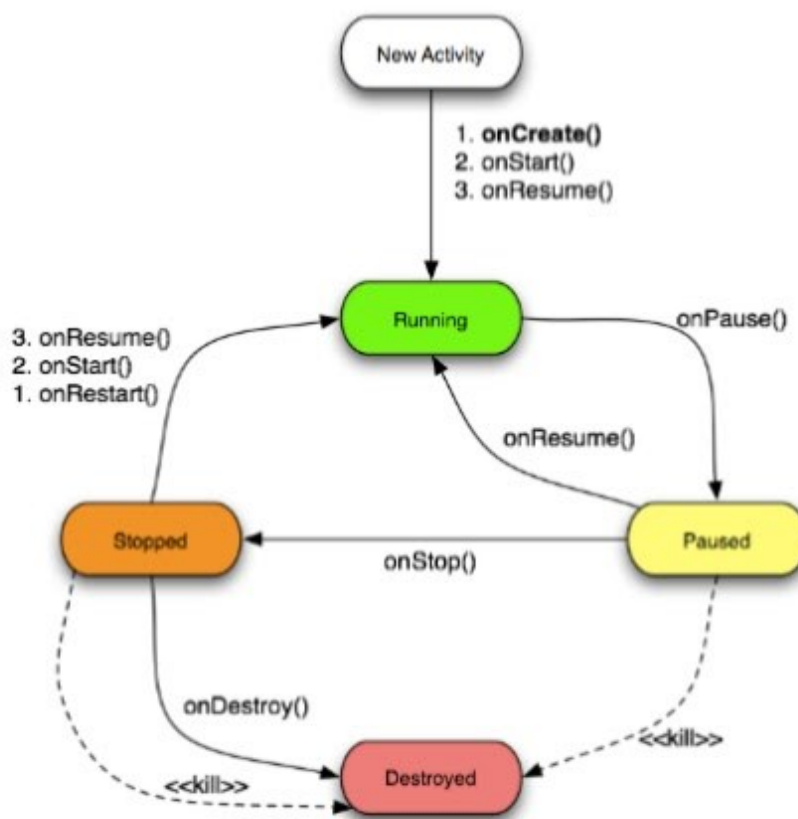
```
package com.example.explicitintent;
import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
public class SecondActivity extends Activity {
    Button btnSecond;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_second);
        btnSecond = (Button) findViewById(R.id.bttSecond);
        btnSecond.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent=new Intent(SecondActivity.this,
MainActivity.class);
                startActivity(intent);
            }
        });
    }
}
```

```
}  
}
```

Then run and debug

Task 2: understanding activity lifecycle in Android

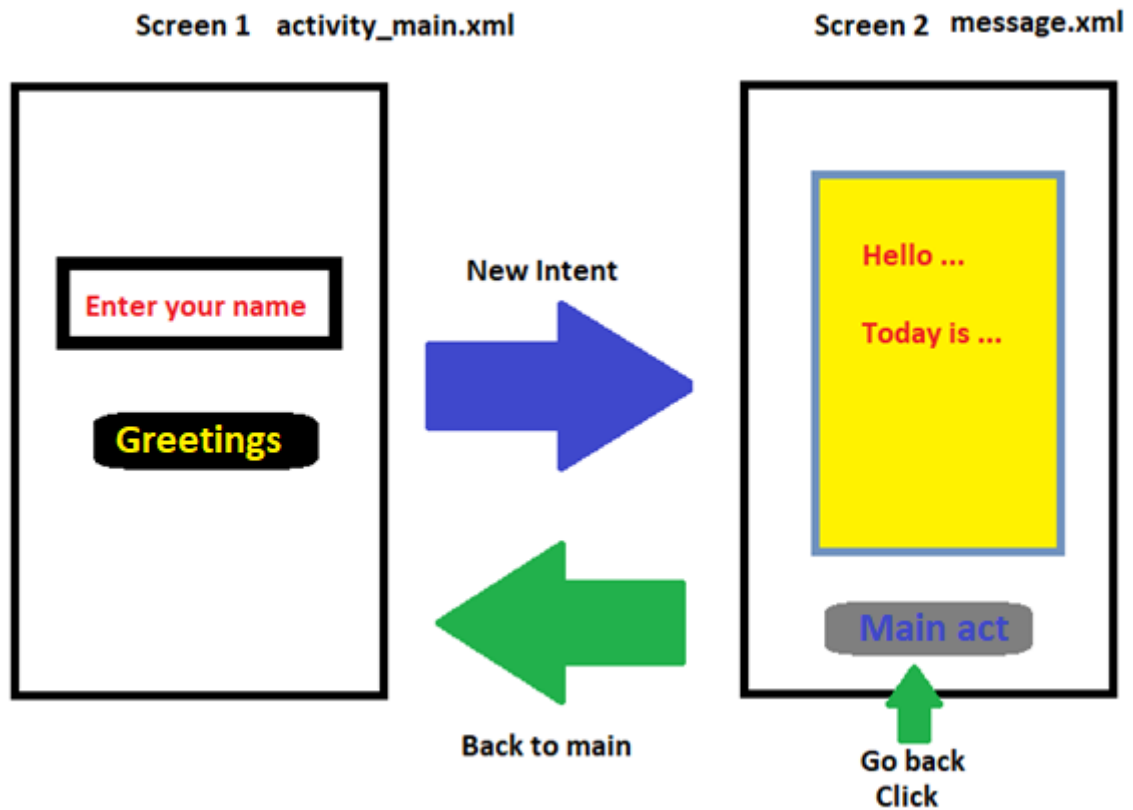
Activity Lifecycle



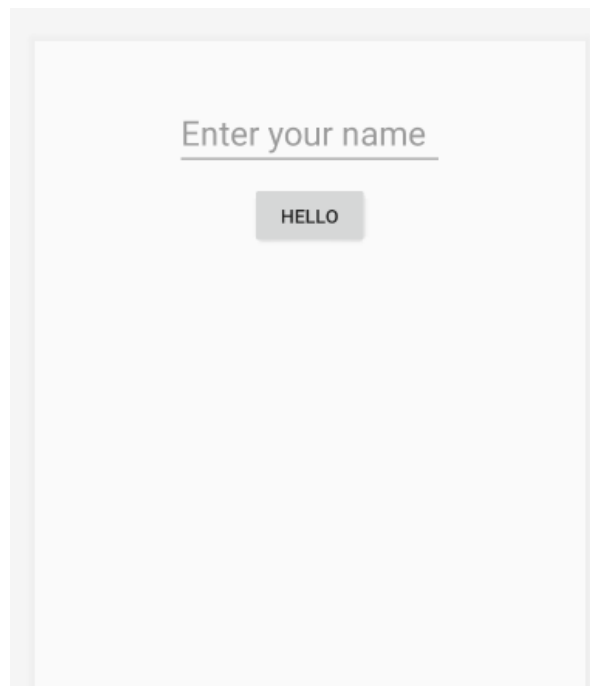
Task 3: using explicit intent for lab 1

The task is demonstrated in the screen below. The screen 1 (`activity_main.xml`) has a `edit_text` to get your name and has a button (Greetings). When you click on greeting button. The screen 2 (`message.xml`) will appear and display on the text view with the message that we did from the lab 1. The screen 2 has a button (Main Act). When you click on this button. The main activity screen will appear.

42036 - Cyber Security for Mobile Platforms
Tutorial 3 manual (Week 3): Intent and Encryption



Make the new project name is First App Intent
Main activity like this



42036 - Cyber Security for Mobile Platforms

Tutorial 3 manual (Week 3): Intent and Encryption

Code for activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="16dp"
    android:paddingLeft="16dp"
    android:paddingRight="16dp"
    android:paddingTop="16dp"
    tools:context=".MainActivity">
    <EditText
        android:id="@+id/editText1"
        android:layout_width="200dp"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="25dp"
        android:hint="@string/name"
        android:inputType="text"
        android:textSize="25sp"
        android:textColor="#000" />
    <Button
        android:id="@+id/button1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/editText1"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="10dp"
        android:text="@string/hello_name" />
</RelativeLayout>
```

Add code into the strings.xml

```
<resources>
    <string name="app_name">First App Intent</string>
    <string name="hello_name">Hello</string>
    <string name="name">Enter your name</string>
    <string name="btn_name2">Main Act</string>
</resources>
```

Add code for Java file for MainActivity.java

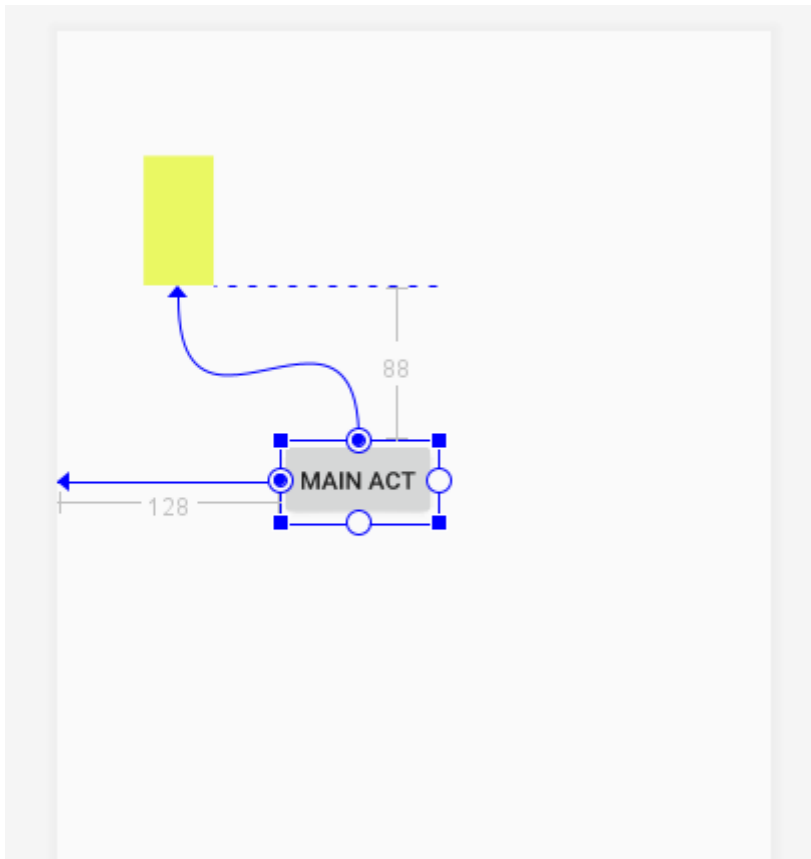
42036 - Cyber Security for Mobile Platforms

Tutorial 3 manual (Week 3): Intent and Encryption

```
package com.example.firstappintent;
import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import static android.provider.AlarmClock.EXTRA_MESSAGE;
public class MainActivity extends Activity {
    EditText editText;
    Button btnSend;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        editText=(EditText) findViewById(R.id.editText1);
        btnSend=(Button) findViewById(R.id.button1);
        btnSend.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                // Make new intent to Hello Activity
                Intent intent=new Intent(MainActivity.this, HelloActivity.class);
                // Declare message and get the string from editText get the string
that we type
                String message=editText.getText().toString();
                // Keep the message in the EXTRA_MESSAGE in intent to prepare to
send to the intent
                intent.putExtra("data", message);
                // Start the Hello Activity
                startActivity(intent);
            }
        });
    }
}
```

Now right click on the package name then create a new activity (empty)
Name for the activity is HelloActivity

42036 - Cyber Security for Mobile Platforms
Tutorial 3 manual (Week 3): Intent and Encryption



Add this code for xml file, activity_hello.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"

    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".HelloActivity">
    <TextView
        android:id="@+id/textView1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:layout_marginStart="50dp"
        android:layout_marginLeft="50dp"
        android:layout_marginTop="72dp"
        android:background="#EAF863"
        android:padding="20dp"
        android:textColor="#F30723"
        android:textSize="25sp"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
```

42036 - Cyber Security for Mobile Platforms

Tutorial 3 manual (Week 3): Intent and Encryption

```
<Button
    android:id="@+id/btMain"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="128dp"
    android:layout_marginLeft="128dp"
    android:layout_marginTop="88dp"
    android:text="@string/btn_name2"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/textView1" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

Java code for HelloActivity.java

```
package com.example.firstappintent;
import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;
import java.util.Date;
import static java.text.DateFormat.getDateInstance;

public class HelloActivity extends Activity {
    TextView txtV;
    Button btnMain;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_hello);
        txtV=(TextView) findViewById(R.id.textView1);
        btnMain=(Button) findViewById(R.id.btMain);
        // Get the Intent from Main and get the string from message name "data"
        final Intent intent=getIntent();
        String message=intent.getStringExtra("data");

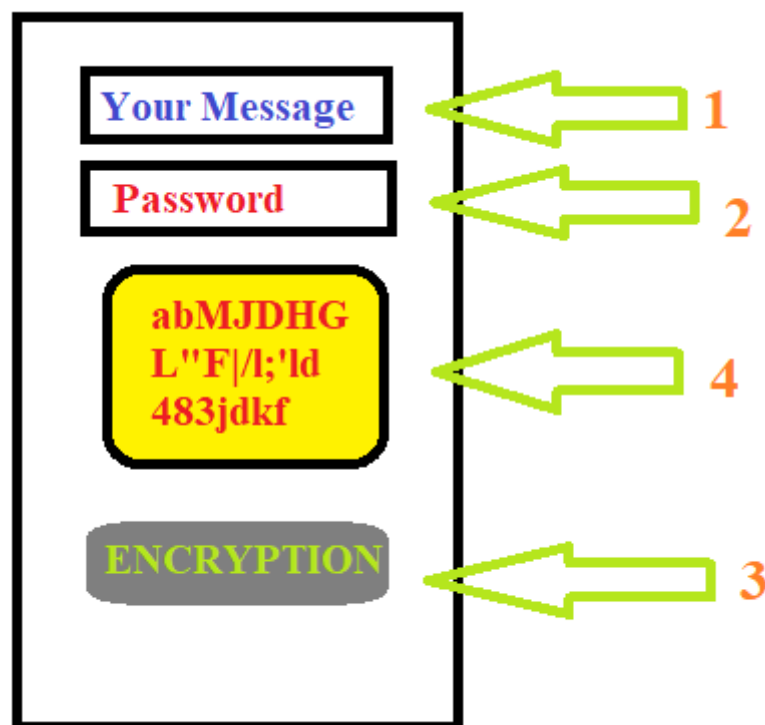
        // Display the text view by hello user and today is...
        String currentDateTimeString = getDateInstance().format(new Date());
        txtV.setText("Hello! " + message + "\n Today is " +
currentDateTimeString);

        // Go back to the main activity when get the click on button main
        btnMain.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent1=new Intent(HelloActivity.this, MainActivity.class);
```

```
startActivity(intent1);  
    }  
    });  
}  
}
```

Task 4: AES encryption of messages

The task is illustrated in the design picture below. We have two edit-texts including the message and the password to get the text input. We have the button (Encryption). And we have a text-view that displays the encrypted message (Cipher text). The task is to program that when we click the button. The AES encrypted text will displayed in the text-view as in the screen.



This the code for this app:

First, we can make a new project with the name is “Enc Dec”

This is the code for xml file activity_main.xml

42036 - Cyber Security for Mobile Platforms

Tutorial 3 manual (Week 3): Intent and Encryption

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="16dp"
    android:paddingLeft="16dp"
    android:paddingRight="16dp"
    android:paddingTop="16dp"
    tools:context=".MainActivity">

    <EditText
        android:id="@+id/inputText"
        android:textColor="#000"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter Text to encrypt"
        android:inputType="text" />

    <EditText
        android:id="@+id/inputPass"
        android:textColor="#000"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter password to encrypt"
        android:layout_below="@id/inputText"
        android:inputType="text" />

    <TextView
        android:id="@+id/outputText"
        android:textSize="25sp"
        android:textColor="#F30723"
        android:background="#EAF863"
        android:layout_centerHorizontal="true"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/inputPass" />

    <Button
        android:id="@+id/encBtn"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/outputText"
        android:layout_centerHorizontal="true"
        android:text="@string/encButton"/>

</RelativeLayout>
```

Add code for strings.xml

42036 - Cyber Security for Mobile Platforms

Tutorial 3 manual (Week 3): Intent and Encryption

```
<resources>
    <string name="app_name">Enc Dec</string>
    <string name="encButton">Encryption</string>
    <string name="decButton">Encryption</string>
</resources>
```

This is code for Java file MainActivity.java

```
package com.example.encdec;

import android.app.Activity;
import android.os.Bundle;
import android.util.Base64;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;
import java.security.MessageDigest;
import javax.crypto.Cipher;
import javax.crypto.spec.SecretKeySpec;

public class MainActivity extends Activity {
    EditText inputTxt, inputPw;
    TextView outputTxt;
    Button encBt, decBt;
    String outputString, encryptedValue, decryptedValue;
    String AES="AES";

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        inputTxt = (EditText) findViewById(R.id.inputText);
        inputPw = (EditText) findViewById(R.id.inputPass);
        outputTxt = (TextView) findViewById(R.id.outputText);
        encBt = (Button) findViewById(R.id.encBtn);
        encBt.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                try {
                    outputString = encrypt(inputTxt.getText().toString(),
inputPw.getText().toString());
                    outputTxt.setText(outputString);
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }
}
```

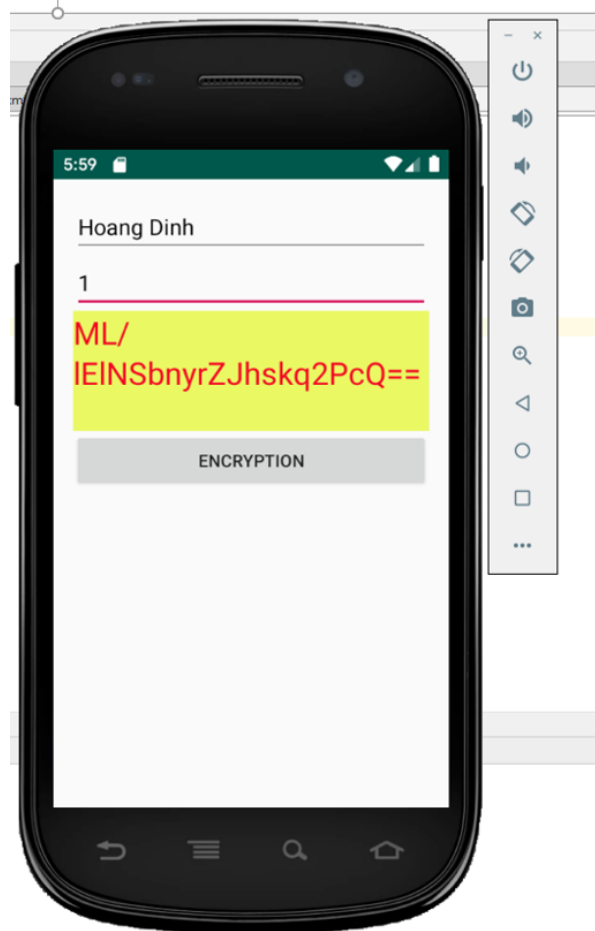

42036 - Cyber Security for Mobile Platforms

Tutorial 3 manual (Week 3): Intent and Encryption

```
// Encrypt function that encrypts the message with key is the input password
private String encrypt (String Data, String pass) throws Exception {
    SecretKeySpec key = generateKey(pass);
    Cipher c = Cipher.getInstance(AES);
    c.init(Cipher.ENCRYPT_MODE, key);
    byte[] encVal = c.doFinal(Data.getBytes());
    encryptedValue = Base64.encodeToString(encVal, Base64.DEFAULT);
    return encryptedValue;
}

// generateKey will create key via AES encryption
private SecretKeySpec generateKey (String pass) throws Exception {
    final MessageDigest digest = MessageDigest.getInstance("SHA-256");
    byte[] bytes = pass.getBytes("UTF-8");
    digest.update(bytes, 0, bytes.length);
    byte[] key = digest.digest();
    SecretKeySpec secretKeySpec = new SecretKeySpec(key, "AES");
    return secretKeySpec;
}
```

Run and debug like this.



42036 - Cyber Security for Mobile Platforms

Tutorial 3 manual (Week 3): Intent and Encryption

That is done for the lab 3 today.

If you have time you can think about the code for the decryption button.

Here is the suggestion

First, you add the decryption button into your xml file

```
<Button
    android:id="@+id/decBtn"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@id/encBtn"
    android:layout_centerHorizontal="true"
    android:text="@string/decButton"/>
```

Then add this code for the code for decryption button

Reflect the decBt in java file

```
//reflect for the decrypt button
decBt=(Button) findViewById(R.id.decBtn);
```

Code for the button decryption catching onclick function

```
//function to do the onclick function for decrypt button
decBt.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        try {
            outputString = decrypt(outputString, inputPw.getText().toString());
        } catch (Exception e) {
            Toast.makeText(MainActivity.this, "Wrong Password",
                Toast.LENGTH_LONG).show();
            e.printStackTrace();
        }
        outputTxt.setText(outputString);
    }
});
```

Function for decrypting the cipher text

```
private String decrypt (String outputString, String pass) throws Exception {
    SecretKeySpec key = generateKey(pass);
    Cipher c = Cipher.getInstance(AES);
    c.init(Cipher.DECRYPT_MODE, key);
    byte[] decodedValue = Base64.decode(outputString, Base64.DEFAULT);
    byte[] decValue = c.doFinal(decodedValue);
    decryptedValue = new String(decValue);
    return decryptedValue;
}
```