Android UI Part 2

Toast / AlertDialog / Style

Toast

- Note : Local Nested Class
- Default
- Custom Position
- Custom Layout With XML
- Custom Layout With HardCoding
- Chap 4-3

Default

```
Toast.makeText(MainActivity.this,
MainActivity.this.getString(R.string.btn_default),
Toast.LENGTH_SHORT).show();
```

Custom Position

```
        Toast toast = Toast.makeText(this,
getString(R.string.btn_custom_layout),
Toast.LENGTH_SHORT);

        toast.setGravity(Gravity.CENTER, 0, 0);

        toast.show();
```

Custom Layout with XML

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

    android:id=“@+id/rootLayout"  android:layout_width=“match_parent"  android:layout_height="match_parent"

    android:background=“#000"  android:orientation=“horizontal" android:padding="10dp" >


    <ImageView

        android:id=“@+id/imageView1"  android:layout_width=“wrap_content"  android:layout_height="wrap_content"

        android:src="@drawable/ic_launcher" />


    <TextView

        android:id=“@+id/textView1"  android:layout_width=“wrap_content"  android:layout_height="wrap_content"

        android:text=“@string/btn_custom_layout_xml"  android:textColor=“#FFF"  android:textSize="30sp" />


</LinearLayout>
```

Custom Layout with XML

```
        Toast toast = new Toast(this);

        toast.setGravity(Gravity.CENTER|Gravity.LEFT, 0, 0);

        LayoutInflater inflater = getLayoutInflater();

        View view =  inflater.inflate(R.layout.toast_demo,
(ViewGroup) findViewById(R.id.rootLayout));

        toast.setView(view);

        toast.setDuration(200);

        toast.show();
```

Custom Position With Hardcoding

```
        Toast toast = Toast.makeText(this,
getString(R.string.btn_custom_position), Toast.LENGTH_SHORT);

        toast.setGravity(Gravity.CENTER|Gravity.RIGHT, 0, 0);

        LinearLayout layout = (LinearLayout)toast.getView();

        ImageView imgView = new ImageView(this);

        imgView.setImageResource(R.drawable.toast_icon);

        layout.addView(imgView,0);

        toast.show();
```

AlertDialog

- Default
- Single Selection
- String Array Values
- Chap 6-1

Default

```
AlertDialog.Builder dialog = new AlertDialog.Builder(this);
dialog.setTitle("Information");
dialog.setPositiveButton("OK", new DialogInterface.OnClickListener() {
    @Override
    public void onClick(DialogInterface dialog, int which) {

    }
});
dialog.show();
```

Single Selection - 1

```
    AlertDialog.Builder dialog = new AlertDialog.Builder(this);

    dialog.setTitle("Information");

    dialog.setSingleChoiceItems(products, -1, new DialogInterface.OnClickListener(){

        @Override

        public void onClick(DialogInterface dialog, int which) {

            // TODO Auto-generated method stub

            Toast.makeText(MainActivity.this, "You selected : "+products[which], Toast.LENGTH_SHORT).show();

            selectedIndex = which;

        }});

    dialog.setPositiveButton("OK", new DialogInterface.OnClickListener() {

        @Override

        public void onClick(DialogInterface dialog, int which) {

            Toast.makeText(MainActivity.this, "You selected : "+products[selectedIndex],
Toast.LENGTH_SHORT).show();
```

Single Selection - 2

```
        dialog.setPositiveButton("OK", new DialogInterface.OnClickListener() {


            @Override

            public void onClick(DialogInterface dialog, int which) {

                Toast.makeText(MainActivity.this, "You selected : "+products[selectedIndex],
Toast.LENGTH_SHORT).show();

            }

        });

        dialog.setNegativeButton("Cancel", new DialogInterface.OnClickListener() {

            @Override

            public void onClick(DialogInterface dialog, int which) {


            }

        });

        dialog.show();
```

strings.xml & code

strings.xml :

```
<resources>

    ……

    <string-array name="products">

        <item>iPod</item>

        <item>iPad</item>

        <item>iPhone</item>

        <item>iMac</item>

    </string-array>

</resources>
```

Code :

```
final String[] products = getResources().getStringArray(R.array.products);
```

Style

- For reuse
- Type1 : Style
    - Create <Project>/res/**values**/styles.xml
- Type2 : Selector
    - Create <Project>/res/**drawable**
    - Create <Project>/res/drawable/**xxxx.xml**(selector)
- Modify layout_xxx.xml

Style : values/**styles**.xml

```
<resources xmlns:android="http://schemas.android.com/apk/res/android">

…….

    <style name="rootLayout">

        <item name="android:layout_width">match_parent</item>

        <item name="android:layout_height">match_parent</item>

        <item name="android:orientation">vertical</item>

    </style>

…….
```

Style : layout_main.xml

```
<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:android="http://schemas.android.com/apk/
res/android"

    android:id="@+id/rootLayout"

      style=“@style/rootLayout">

........
```

Selector : drawable/**button_voice**.xml

```
<selector xmlns:android="http://schemas.android.com/apk/res/
android">

    <item android:drawable="@drawable/
search_voiceseach_btn_press" android:state_focused="true"></item>

    <item android:drawable="@drawable/
search_voiceseach_btn_press" android:state_pressed="true"></
item>

    <item android:drawable="@drawable/
search_voiceseach_btn_on"></item>

</selector>
```

Selector : layout_main.xml

```
<RelativeLayout xmlns:android=“http://schemas.android.com/apk/res/android" ….

    tools:context=".MainActivity" >

    …………………….

    <Button

        android:id="@+id/imageButton1"

        android:layout_width="wrap_content"

        android:layout_height="wrap_content"

        android:layout_alignBottom="@+id/editText1"

        android:layout_marginLeft="21dp"

        android:layout_toRightOf="@+id/editText1"

        android:background="@drawable/button_voice" />


</RelativeLayout>
```