

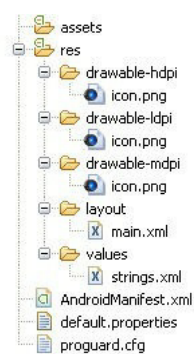
Risorse Android

Programmazione di dispositivi mobili - v1.0

1

Struttura delle risorse

- Android fornisce una architettura per esternalizzare alcune informazioni
 - es. un parametro può essere inserito in un file di configurazione
- Le risorse in Android sono contenute in file organizzate in cartelle/file precise
- Le due cartelle principali sono **res** e **asset**
 - res conterrà le risorse da “compilare”
 - asset conterrà le altre risorse



Programmazione di dispositivi mobili - v1.0

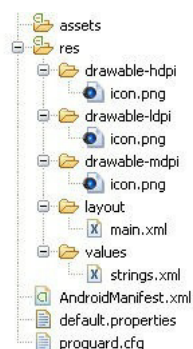
2

Risorse nella cartella res

- le risorse sono dichiarate nei file

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
  <string name="hello">Hello World, HelloWorld2Activity!</string>
  <string name="app_name">HelloWorld2</string>
</resources>
```

- le risorse possono essere dei file
 - eg. l'icona per lanciare la activity principale



Programmazione di dispositivi mobili - v1.0

3

La classe R

- In compilazione ad ogni risorsa viene associata una costante di tipo int

```
package ppl.pdm.helloworld2;

public final class R {
    public static final class attr {
    }
    public static final class drawable {
        public static final int ic_launcher=0x7f020000;
    }
    public static final class layout {
        public static final int main=0x7f030000;
    }
    public static final class string {
        public static final int app_name=0x7f040001;
        public static final int hello=0x7f040000;
    }
}
```

- la classe R è contenuta nella cartella gen
- getResources().getString(R.string.hello);

Programmazione di dispositivi mobili - v1.0

4

Risorse che descrivono valori

- Sono contenute in file xml nella cartella res/values
- Sono dichiarate in fra tag specifici

- <string>
- <string-array>
- <integer-array>
- <color>
- <drawable>
- <style>
- <dimension>

```
<?xml version="1.0" encoding="UTF-8"?>
<resources>
  <string name="app_title">Button Counter</string>
  <string name="button_label">Press</string>
  <string name="output_label">Numero Click:</string>
  <string-array name="myArray">
    <item>Valore 1</item>
    <item>Valore 2</item>
    <item>Valore 3</item>
  </string-array>
  <integer-array name="primeArray">
    <item>2</item>
    <item>3</item>
    <item>5</item>
    <item>7</item>
    <item>11</item>
  </integer-array>
</resources>
```

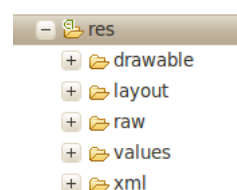
- Per accedere
 - R.<tipo>.<nome>
 - eg. R.string.app_title

Programmazione di dispositivi mobili - v1.0

5

File come risorse

- Alcuni file sono risorse
 - Non i valori contenuti
 - es. le immagini, i file xml
- I file vanno inseriti in cartelle specifiche
 - Le immagini/icone in res/drawable
 - I file xml generici in res/xml
 - raw contiene file qualunque
- per accedere alla risorsa uso la costante
 - R.<cartella>.<file name>
 - eg. R.xml.libri



Programmazione di dispositivi mobili - v1.0

6

Risorse di tipo Layout

- Ogni activity ha almeno un layout
 - sono contenuti nella cartella layout
- Serve a posizionare degli “elementi grafici” sullo schermo
 - le view

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical" >
    <TextView
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:text="@string/hello" />
</LinearLayout>
```

- Esempi - ResourceLayout // ResourceColor

Programmazione di dispositivi mobili - v1.0

7

Identificatori per le risorse

- Ad ogni risorsa viene associato un “nome”
- Ad ogni componente viene associato un id che si usa nel codice per farvi riferimento
 - android:id
- Sintassi per identificatori di risorse
 - @[package:][+]type/name
- Esempi
 - @string/app_name
 - @+id/myButton
 - @id/myButton

Programmazione di dispositivi mobili - v1.0

8