**Android 资源(Resources)访问**

有许多东西用来构建一个优秀的 Android 应用程序。除了应用程序的编码，你需要关注各种各样的资源，诸如你用到的各种静态内容，如位图，颜色，布局定义，用户界面字符串，动画等等。这些资源一般放置在项目的 res/ 下独立子目录中。

这节教程将学习如何来组织应用程序资源，指定替代资源，并在应用程序中访问它们。

## 在eclipse中组织资源

你需要将每种资源放置在项目中 res/ 目录的特定子目录下。例如，这是一个简单项目的文件层级：



res/ 目录在各种子目录中包含了所有的资源。这里有一个图片资源，两个布局资源和一个字符串资源文件。下表详细的给出了在项目中 res/ 目录里面支持的资源。

| **目录** | **资源类型** |
| --- | --- |
| anim/ | 定义动画属性的XML文件。它们被保存在res/anim/文件夹下，通过R.anim类访问 |
| color/ | 定义颜色状态列表的XML文件。它们被保存在res/color/文件夹下，通过R.color类访问 |
| drawable/ | 图片文件，如.png,.jpg,.gif或者XML文件，被编译为位图、状态列表、形状、动画图片。它们被保存在res/drawable/文件夹下，通过R.drawable类访问 |
| layout/ | 定义用户界面布局的XML文件。它们被保存在res/layout/文件夹下，通过R.layout类访问 |
| menu/ | 定义应用程序菜单的XML文件，如选项菜单，上下文菜单，子菜单等。它们被保存在res/menu/文件夹下，通过R.menu类访问 |
| raw/ | 任意的文件以它们的原始形式保存。需要根据名为R.raw.filename的资源ID，通过调用Resource.openRawResource()来打开raw文件 |
| values/ | 包含简单值(如字符串，整数，颜色等)的XML文件。这里有一些文件夹下的资源命名规范。arrays.xml代表数组资源，通过R.array类访问；integers.xml代表整数资源，通过R.integer类访问；bools.xml代表布尔值资源，通过R.bool类访问；colors.xml代表颜色资源，通过R.color类访问；dimens.xml代表维度值，通过R.dimen类访问；strings.xml代表字符串资源，通过R.string类访问；styles.xml代表样式资源，通过R.style类访问 |
| xml/ | 可以通过调用Resources.getXML()来在运行时读取任意的XML文件。可以在这里保存运行时使用的各种配置文件 |

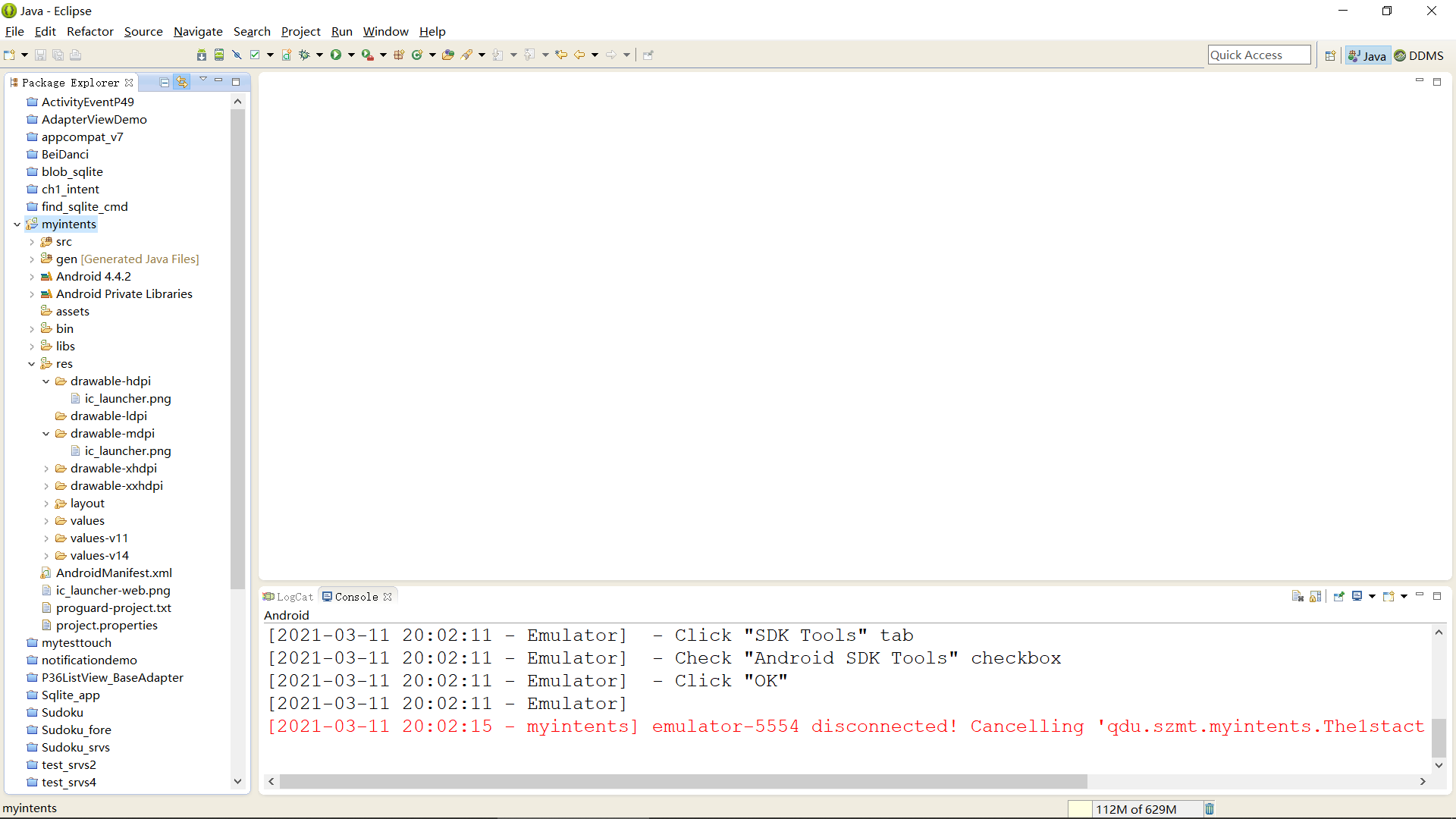
## 替代资源

你的应用程序需要为特定的设备配置提供替代的资源支持。比如说，你需要为不同的屏幕分辨率提供替代的图片资源，为不同的语言提供替代的字符串资源。在运行时，Android 检测当前设备配置，并为应用程序加载合适的资源。

要为特定的配置的确定一系列替代资源，遵循如下的步骤：

* 在res/ 下创建一个新的目录，以 <resource\_name>\_<config\_qualifier> 的方式命名。这里的 resources\_name 是上表中提到的任意资源，如布局、图片等。 qualifier 将确定个性的配置使用哪些资源。你可以查看官方文档来了解不同类型资源的一个完整 qualifier 列表。
* 在这个目录中保存响应的替代资源。这些资源文件必须与下面例子中展示的默认资源文件名一致，然而这些文件将确定的内容进行替代。例如：虽然图片的文件名一样，但是高分辨率的屏幕，图片的分辨率也会高。

下图中drawable文件夹包含图片不同分辨率版本。



## 访问资源

在应用程序开发中，需要访问定义好的资源，不论是通过代码还是通过 XML 文件。下面的章节介绍如何分别在这两种场景中访问资源。

### 在代码访问资源

当 Android 应用程序被编译，生成一个 R 类，其中包含了所有 res/ 目录下资源的 ID。你可以使用 R 类，通过子类+资源名或者直接使用资源 ID 来访问资源。

### 实例

访问 res/drawable/myimage.png，并将其设置到 ImageView 上，你将使用以下代码：

ImageView imageView = (ImageView) findViewById(R.id.myimageview);

imageView.setImageResource(R.drawable.myimage);

这里第一行代码用 R.id.myimageview 来在布局文件中获取定义为 myimageview 的 ImageView。第二行用 R.drawable.myimage 来获取在 res/ 的 drawable 子目录下名为 myimage 的图片。

### 实例

考虑下一个例子，其中 res/values/strings.xml 有如下定义：

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

<resources>

<string name="hello">Hello, World!</string>

</resources>

现在你可以在 ID 为 msg 的 TextView 对象上使用资源 ID 来设置文本，具体如下：

TextView msgTextView = (TextView) findViewById(R.id.msg);

msgTextView.setText(R.string.hello);

### 实例

考虑如下定义的布局 res/layout/activity\_main.xml

<?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:id="@+id/text"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Hello, I am a TextView" />

<Button android:id="@+id/button"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Hello, I am a Button" />

</LinearLayout>

这个应用程序代码将为活动加载这个布局，onCreate() 方法中如下：

public void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.main\_activity);

}

### 在XML中访问

考虑下面的 XML 资源文件 res/values/strings.xml，其中包含一个颜色资源和一个字符串资源 -

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

<resources>

<color name="opaque\_red">#f00</color>

<string name="hello">Hello!</string>

</resources>

现在，你可以在下面的布局文件中使用这些资源来设置文本颜色和文本内容：

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

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

android:layout\_width="fill\_parent"

android:layout\_height="fill\_parent"

android:textColor="@color/opaque\_red"

android:text="@string/hello" />