

# Hello, Android

日本Androidの会 勉強会WG

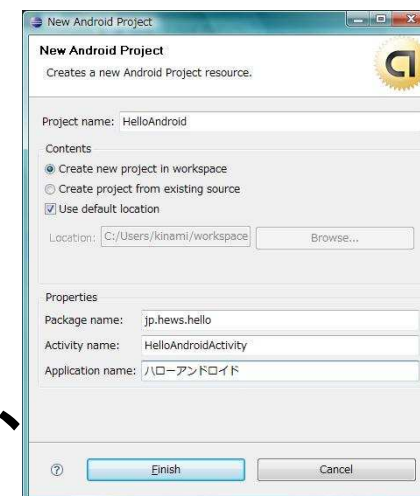
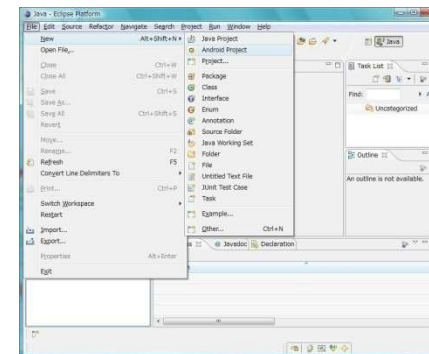
木南英夫

# Hello, Android

- Hello, Worldの中身を理解する
  - 開発ディレクトリの構成
  - プログラムの実行方法
  - Activityクラス
  - XMLによる画面の構築方法
  - XMLによる文字列の指定方法を

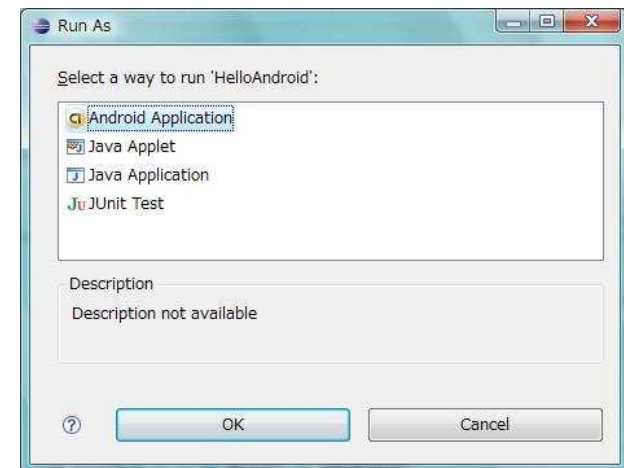
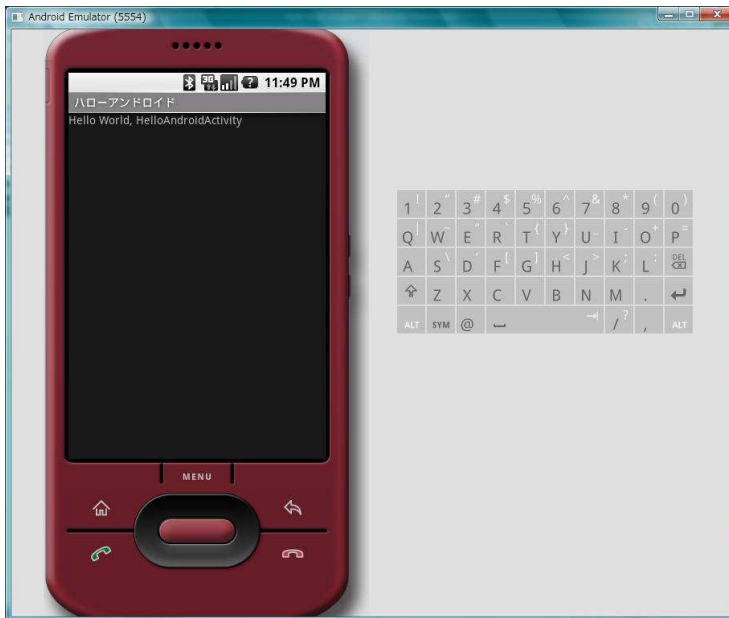
# Eclipseプロジェクトの作成

- Eclipseを立ち上げて、File > New > Android Projectメニューを選ぶ
- 表示された「New Android Project」に以下の内容を記入します
  - Project Name: HelloAndroid
  - Package Name: jp.hews.hello
  - Activity Name: HelloAndroidActivity
  - Application Name: ハローアンドロイド
- 設定後、「Finish」ボタンを選択すると、プロジェクトが作成される



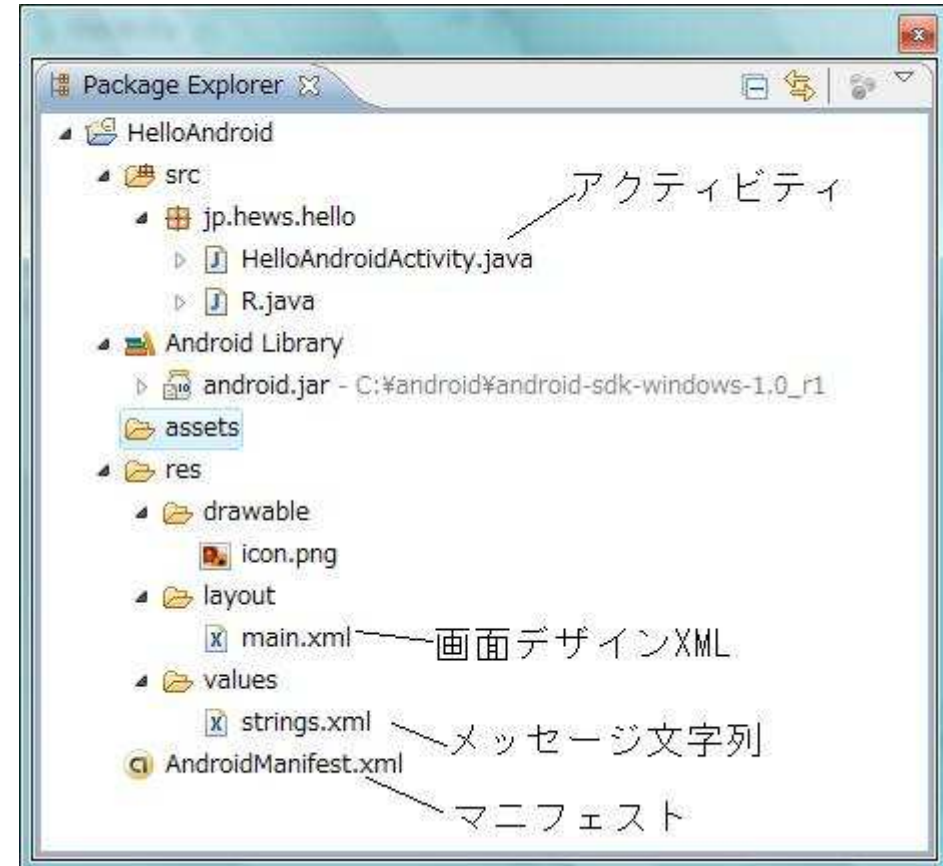
# プログラムの実行

- Run > Runメニュー (Ctrl + F11)を選択
- Android Applicationを選択
- エミュレータが実行される



# 開発ディレクトリの構成

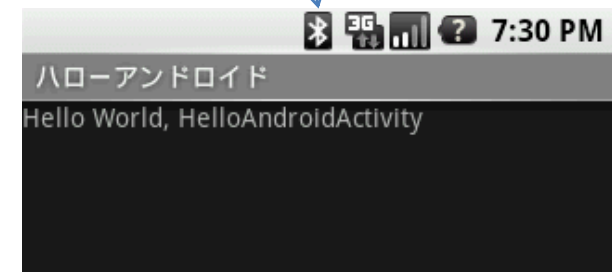
- HelloAndroidActivity.java
  - アプリケーションのロジックを実装する
- R.java
  - XMLで記述されたリソースやイメージファイルを参照する
- Icon.png
  - アプリケーションのアイコン
- main.xml
  - アプリケーションの画面を記述
- strings.xml
  - アプリケーションが表示する文字列
- AndroidManifest.xml
  - アプリケーションの属性を記述



# アクティビティクラス

- 一つの画面を制御するクラスです
- onCreateから始まります

```
public class HelloAndroidActivity extends Activity {  
  
    public void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.main);  
    }  
}
```

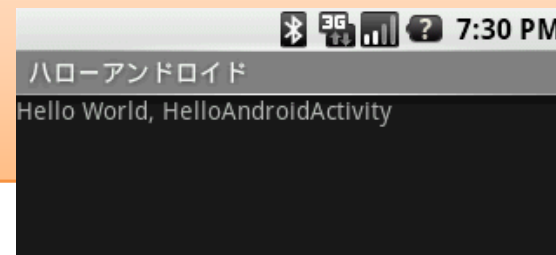


# XMLによる画面と文字の定義

- main.xmlで、画面の構成を記述しています
- strings.xmlで表示する文字列を定義してしま

```
<LinearLayout xmlns:android="...">  
  <TextView  
    android:text="@string/hello" />  
</LinearLayout>
```

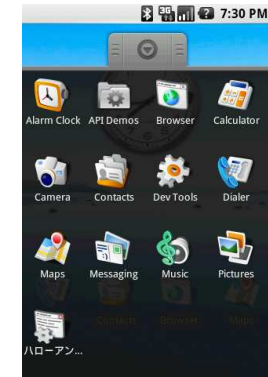
main.xml



```
<resources>  
  <string name="hello">Hello World, ...</string>  
</resources>
```

# マニフェストファイル

- アプリケーションの属性を記述する
  - どのアイコンをランチャーに表示するか？
  - どのクラスから開始するか？
  - インターネットにアクセスするか？



```
<manifest xmlns:android="..." package="jp.hews.hello">
  <application
    android:icon="@drawable/icon"
    android:label="@string/app_name">
    <activity android:name=".HelloAndroidActivity"
      android:label="@string/app_name">
      <intent-filter>...</intent-filter>
    </activity>
  </application>
</manifest>
```