

Знакомство с Android

Особенности Android.

Средства разработки и структура проекта.

Основные компоненты

Android. Основные особенности

- Первая действительно открытая и всеобъемлющая платформа для мобильных устройств и любого программного обеспечения, предназначенного для работы на мобильном телефоне, при этом без патентных ограничений, которые сдерживали развитие портативных устройств.

Android. Основные особенности

- Первая действительно открытая и всеобъемлющая платформа для мобильных устройств и любого программного обеспечения, предназначенного для работы на мобильном телефоне, при этом без патентных ограничений, которые сдерживали развитие портативных устройств.
- Исполнение программ на Dalvik VM

Android. Основные особенности

- Первая действительно открытая и всеобъемлющая платформа для мобильных устройств и любого программного обеспечения, предназначенного для работы на мобильном телефоне, при этом без патентных ограничений, которые сдерживали развитие портативных устройств.
- Исполнение программ на Dalvik VM
- Программы созданные для Java ME не работают на Android

Android. Основные особенности

- Первая действительно открытая и всеобъемлющая платформа для мобильных устройств и любого программного обеспечения, предназначенного для работы на мобильном телефоне, при этом без патентных ограничений, которые сдерживали развитие портативных устройств.
- Исполнение программ на Dalvik VM
- Программы созданные для Java ME не работают на Android
- Встроенная работа библиотек с открытым исходным кодом SQLite, OpenGL, Web Kit

Android. Терминология

- **Android SDK** — Software Development Kit — набор программного обеспечения для разработчика под ОС Android
- **AVD** — Android Virtual Device — виртуальное устройство Android
- **ADT** — Android Development Tools — набор инструментов Android разработчика
- **NDK** — Native Development Kit — библиотека для разработки нативных (C/C++) приложений для Android
- **ADB** — Android Device Bridge — инструмент для взаимодействия с Android устройством

Android. Средства разработки

- IDE Eclipse



Android. Средства разработки

- IDE Eclipse
- ADT (<https://dl-ssl.google.com/android/eclipse/>) + SDK



Android. Средства разработки

- IDE Eclipse
- ADT (<https://dl-ssl.google.com/android/eclipse/>) + SDK
- или Android Studio



Android. Средства разработки

- IDE Eclipse
- ADT (<https://dl-ssl.google.com/android/eclipse/>) + SDK
- или Android Studio
- Менеджер виртуальных устройств и эмулятор



Android. Средства разработки

- IDE Eclipse
- ADT (<https://dl-ssl.google.com/android/eclipse/>) + SDK
- или Android Studio
- Менеджер виртуальных устройств и эмулятор
- или Android устройство



Android. Средства разработки

- IDE Eclipse
- ADT (<https://dl-ssl.google.com/android/eclipse/>) + SDK
- или Android Studio
- Менеджер виртуальных устройств и эмулятор
- или Android устройство
- Полный набор документации

Android. Средства разработки

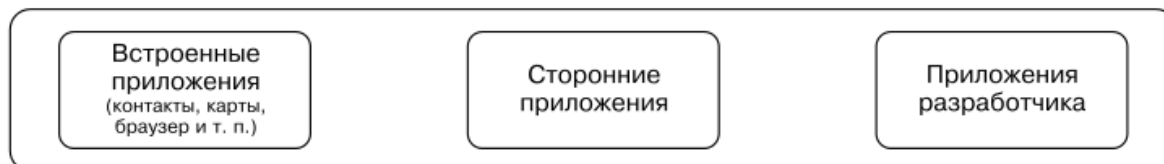
- IDE Eclipse
- ADT (<https://dl-ssl.google.com/android/eclipse/>) + SDK
- или Android Studio
- Менеджер виртуальных устройств и эмулятор
- или Android устройство
- Полный набор документации
- Примеры кода и online поддержка (developer.android.com, www.stackoverflow.com)

Android. Средства разработки

- IDE Eclipse
- ADT (<https://dl-ssl.google.com/android/eclipse/>) + SDK
- или Android Studio
- Менеджер виртуальных устройств и эмулятор
- или Android устройство
- Полный набор документации
- Примеры кода и online поддержка (developer.android.com, www.stackoverflow.com)
- **Знание английского языка** (must have)

Программный стек Android

Уровень приложений



Фреймворк приложений



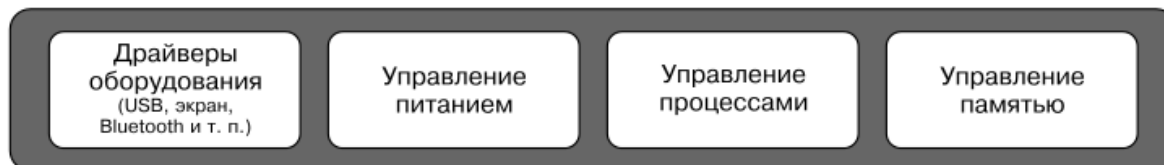
Библиотеки



Рабочая среда Android

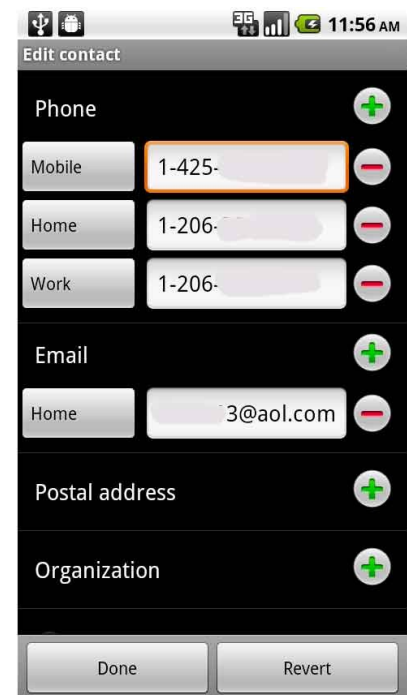


Linux-ядро



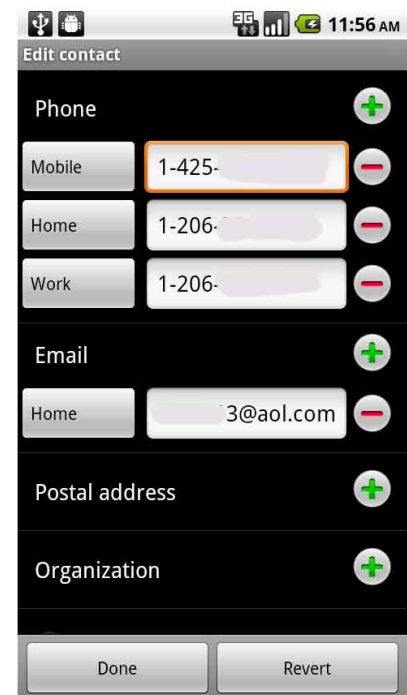
Android. Главные компоненты

- **Activity** — компонент, имеющий представление видимое пользователю.



Android. Главные компоненты

- **Activity** — компонент, имеющий представление видимое пользователю (интерфейс).
 - основа — xml-шаблон (layout), содержащий все визуальные элементы;
 - создается как наследник от класса **Activity**



Android. Главные компоненты

- **Activity** — компонент, имеющий представление видимое пользователю.
- **Service** — компонент, исполняющийся без видимого интерфейса.

A rectangular box with a thin gray border, containing the text "No Image Available" in a gray, sans-serif font, centered within the box.

No Image
Available

Android. Главные компоненты

- **Activity** — компонент, имеющий представление видимое пользователю.
- **Service** — компонент, исполняющийся без видимого интерфейса.
 - применяется для выполнения задач, не требующих взаимодействия с пользователем (проигрывание музыки, проверка обновлений,...)
 - может быть запущен из Activity
 - Activity может подключиться к Service для обмена данными



No Image
Available

Android. Главные компоненты

- **Activity** — компонент, имеющий представление видимое пользователю.
- **Service** — компонент, исполняющийся без видимого интерфейса.
- **Content Provider** — компонент, предоставляющий доступ к данным (другого приложения, БД, ...).

Android. Главные компоненты

- **Activity** — компонент, имеющий представление видимое пользователю.
- **Service** — компонент, исполняющийся без видимого интерфейса.
- **Content Provider** — компонент, предоставляющий доступ к данным (другого приложения, БД, ...).
 - позволяет приложениям получить доступ данным другого приложения
 - предоставляет единообразный доступ к различным данным (БД, файлы, интернет, ...)

Android. Главные компоненты

- **Activity** — компонент, имеющий представление видимое пользователю.
- **Service** — компонент, исполняющийся без видимого интерфейса.
- **Content Provider** — компонент, предоставляющий доступ к данным (другого приложения, БД, ...).
- **Broadcast Receiver** — компонент, принимающий широковещательные сообщения.

Android. Главные компоненты

- **Activity** — компонент, имеющий представление видимое пользователю.
- **Service** — компонент, исполняющийся без видимого интерфейса.
- **Content Provider** — компонент, предоставляющий доступ к данным (другого приложения, БД, ...).
- **Broadcast Receiver** — компонент, принимающий широковещательные сообщения.
 - позволяет приложению реагировать на системные события (выключение экрана, низкий заряд батареи, ...)

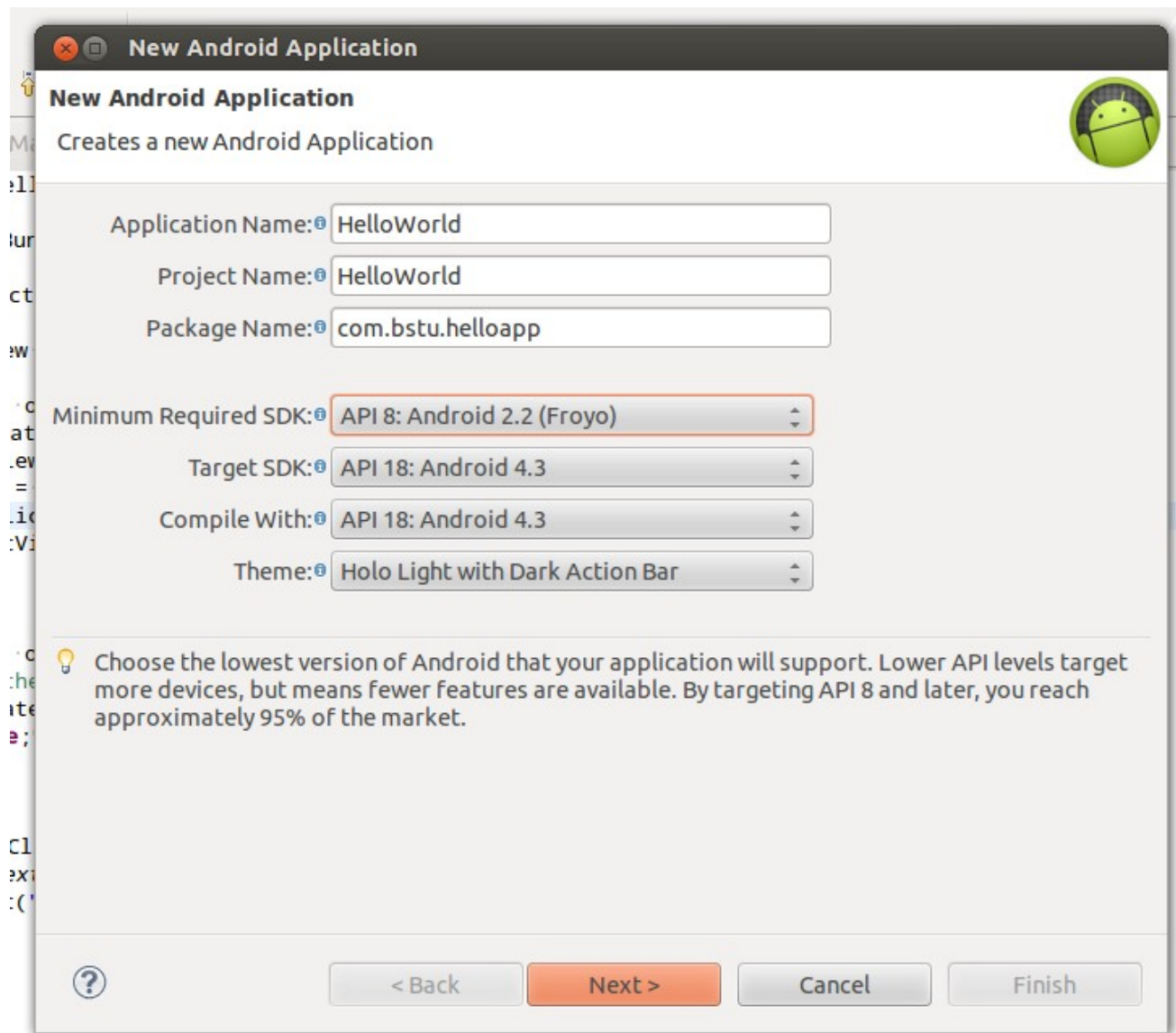
Android. Главные компоненты

- **Activity** — компонент, имеющий представление видимое пользователю.
- **Service** — компонент, исполняющийся без видимого интерфейса.
- **Content Provider** — компонент, предоставляющий доступ к данным (другого приложения, БД, ...).
- **Broadcast Receiver** — компонент, принимающий широковещательные сообщения.
- **Каждый** их этих четырех компонентов является **точкой входа** в приложение. (нету метода `main()`)

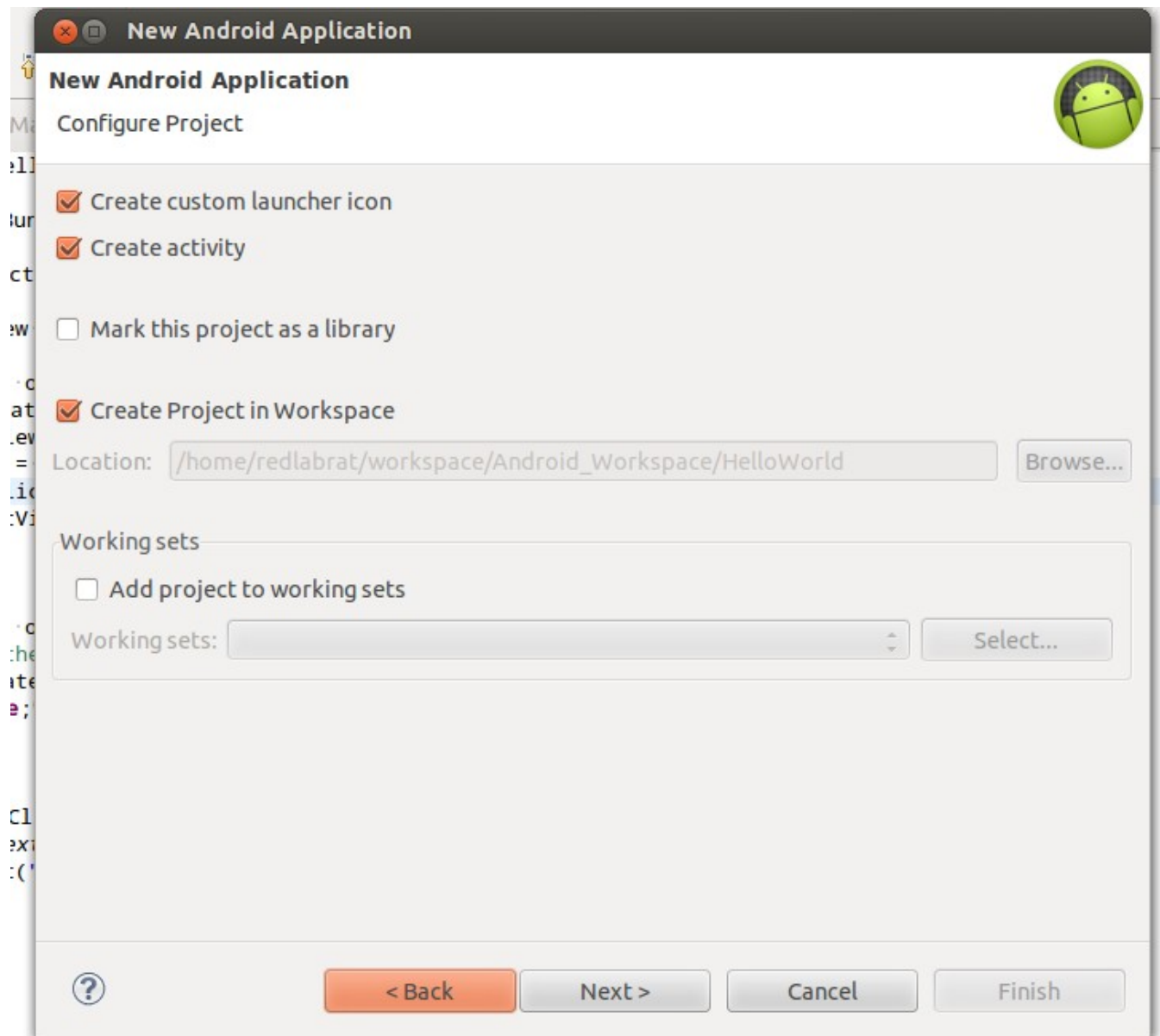
Android. Запуск компонентов

- **Intent** — асинхронное сообщение, связывающее 2 компонента между собой.
 - позволяет запускать Activity (`startActivity`);
 - позволяет запускать Service (`startService`);
 - позволяет подключиться к Service (`bindService`);
 - позволяет отправить широковещательное сообщение (`sendBroadcast`).

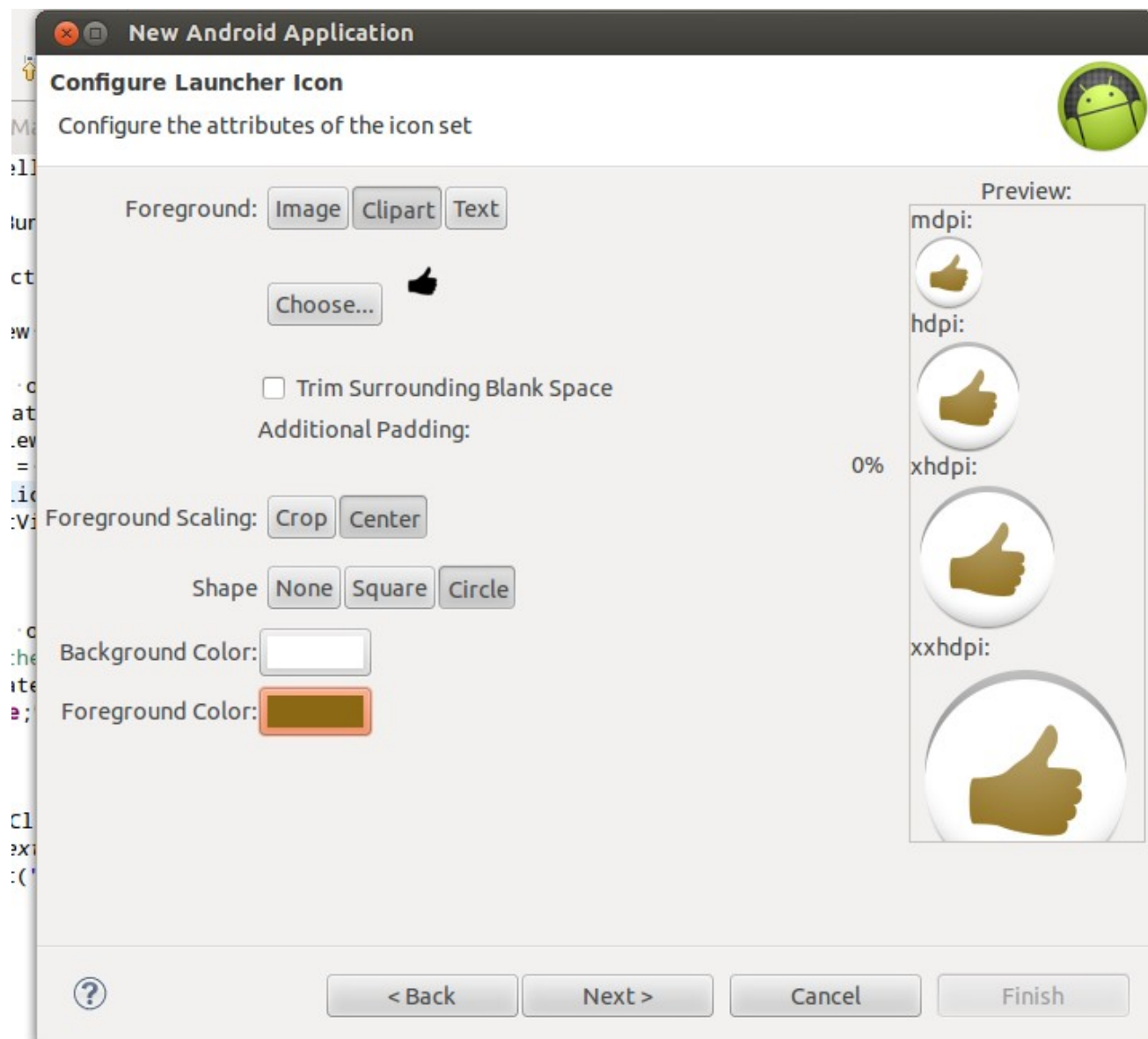
Android. Создание проекта



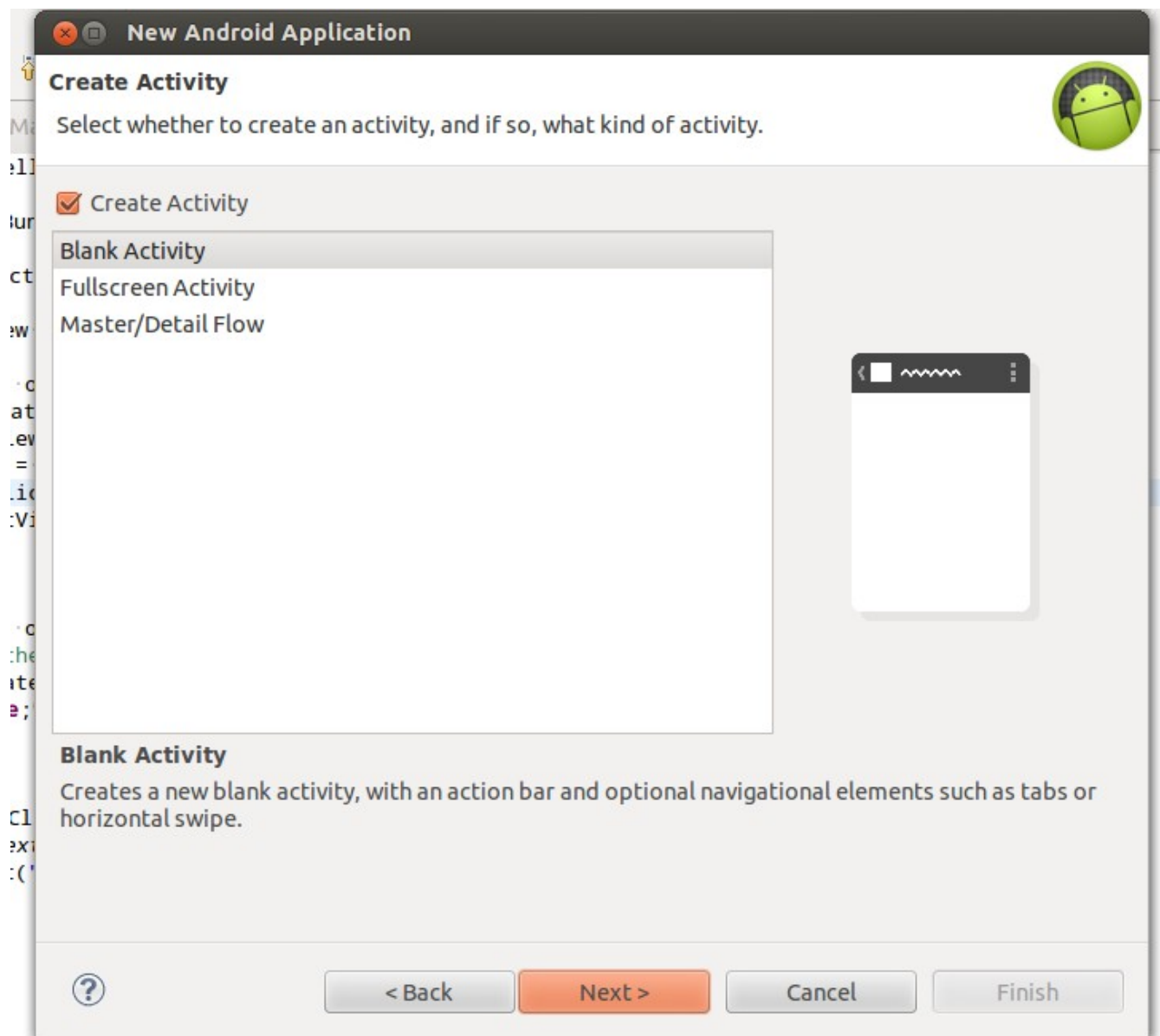
Android. Создание проекта



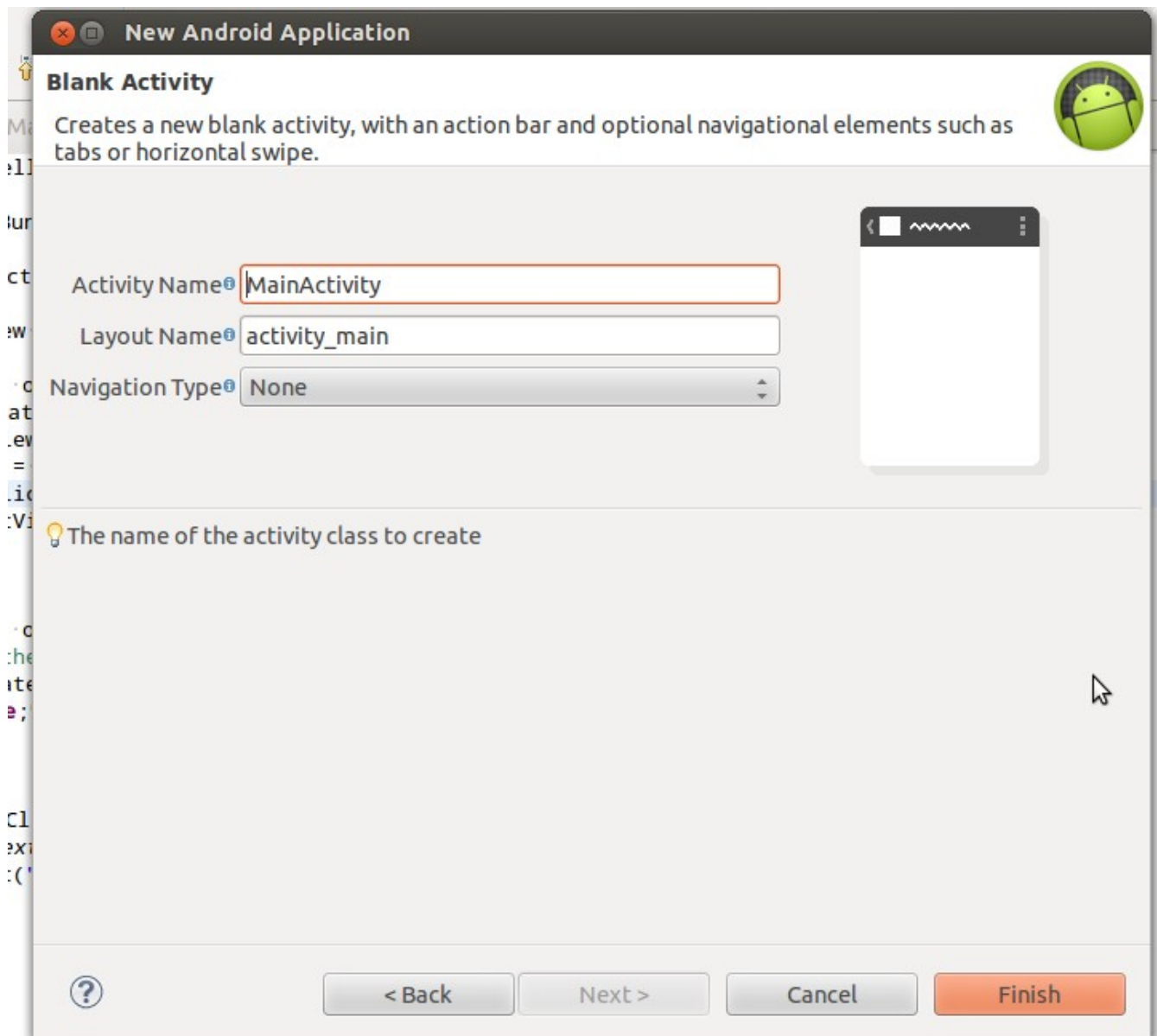
Android. Создание проекта



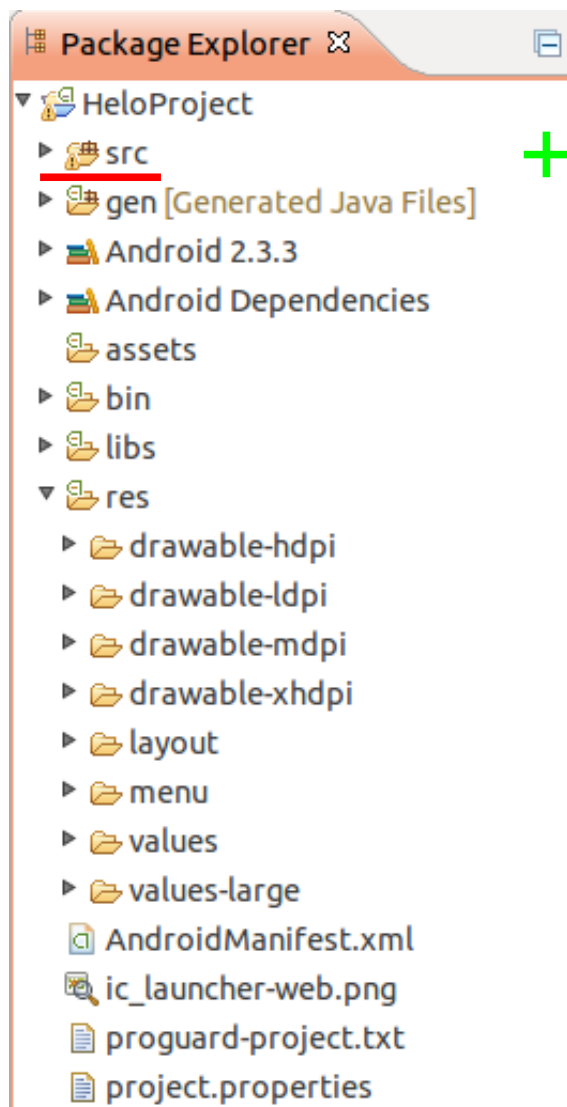
Android. Создание проекта



Android. Создание проекта

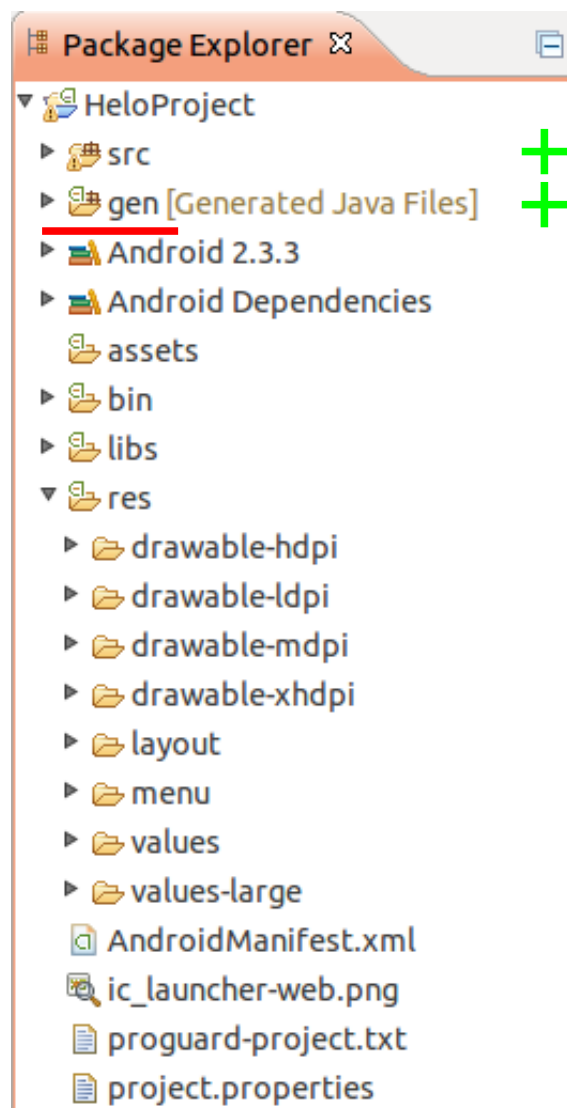


Структура проекта



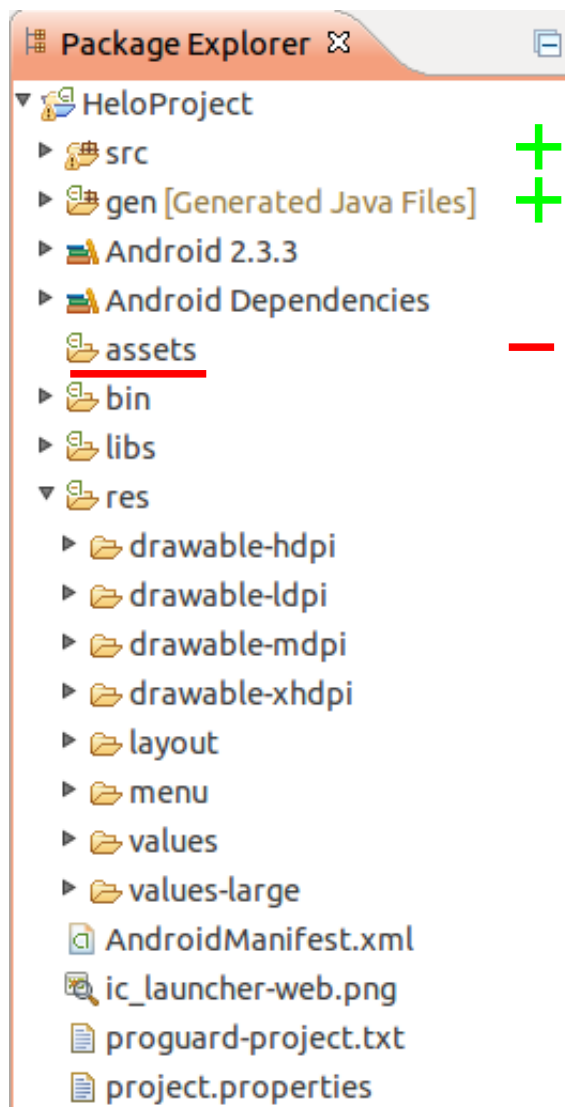
- src — каталог исходного кода приложения. Включает подкаталоги.

Структура проекта



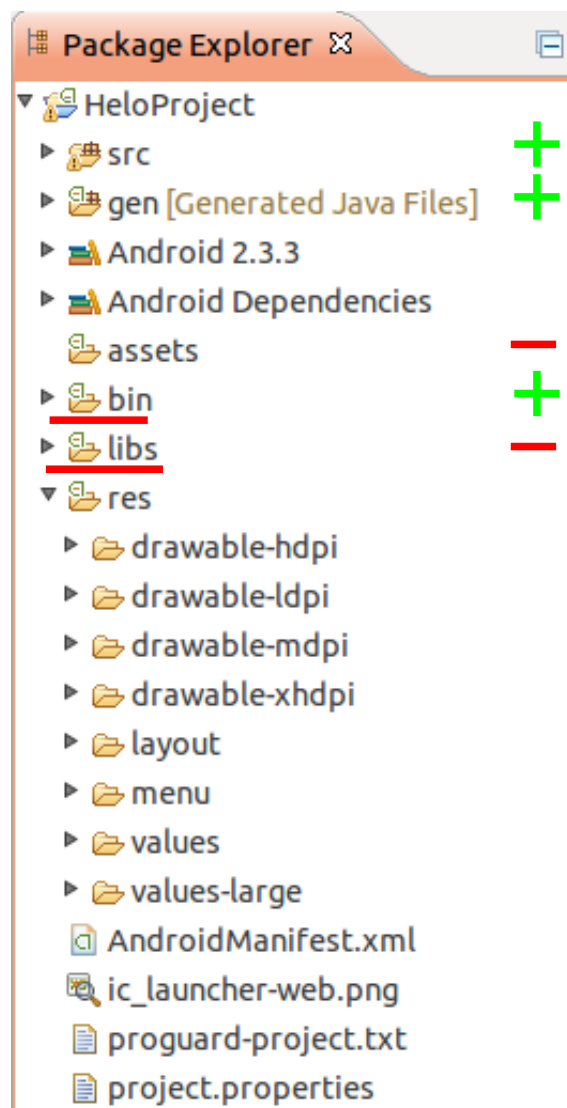
- src — каталог исходного кода приложения. Включает подкаталоги.
- gen — каталог файлов, сгенерированных ADT в процессе разработки

Структура проекта



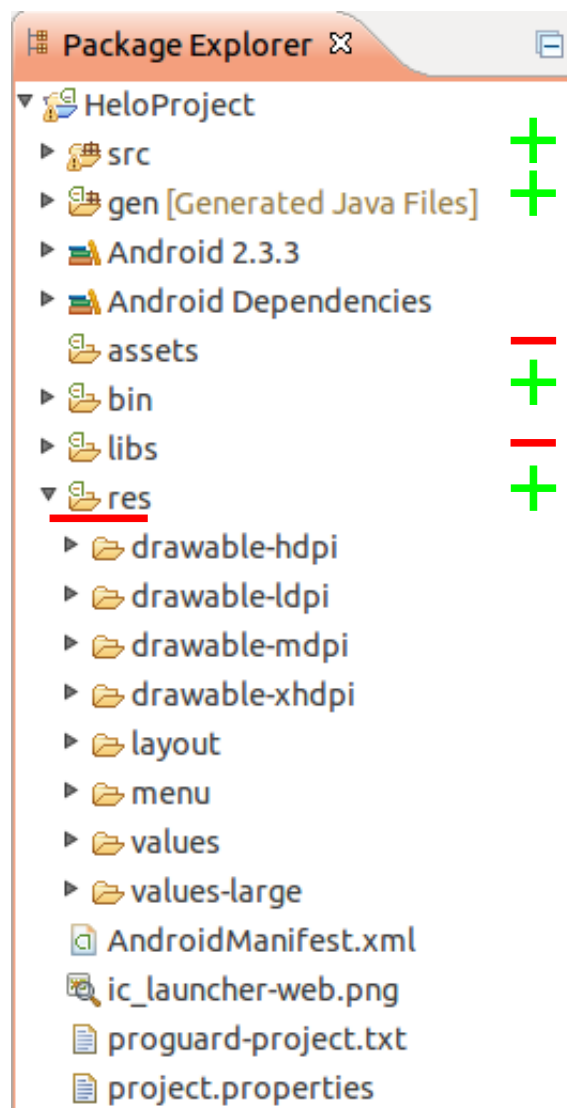
- src — каталог исходного кода приложения. Включает подкаталоги.
- gen — каталог файлов, сгенерированных ADT в процессе разработки.
- assets — собрание произвольных файлов и каталогов.

Структура проекта



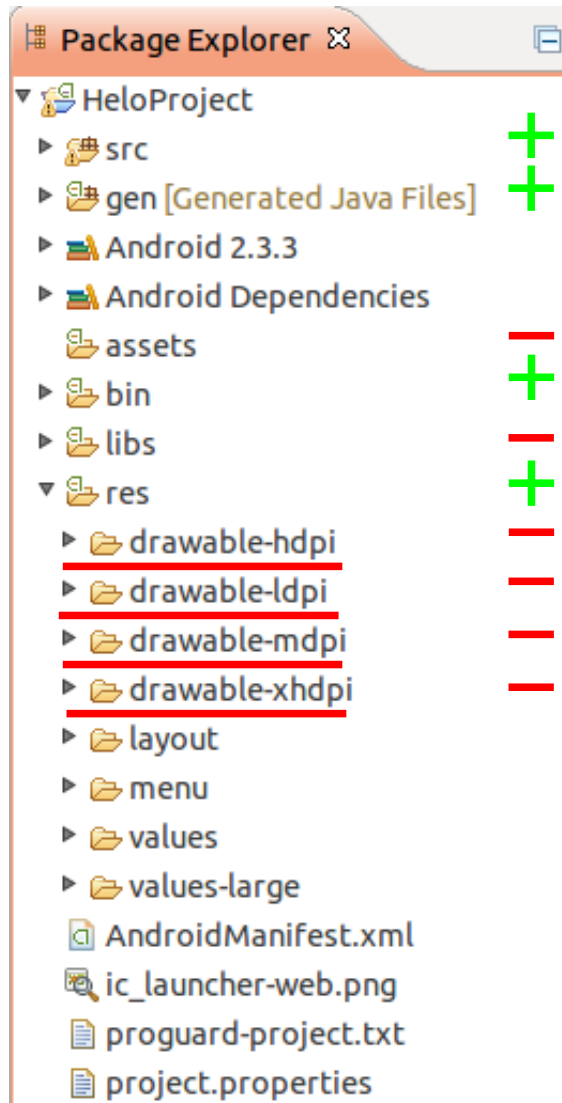
- src — каталог исходного кода приложения. Включает подкаталоги.
- gen — каталог файлов, сгенерированных ADT в процессе разработки.
- assets — собрание произвольных файлов и каталогов.
- bin — скомпилированные файлы проекта
- libs — дополнительные библиотеки

Структура проекта



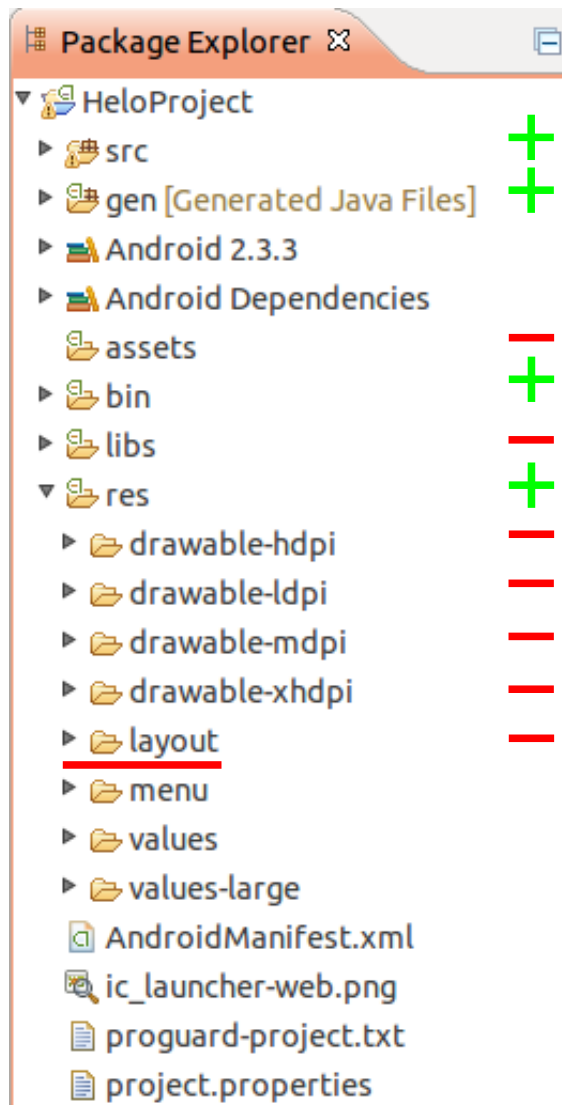
- src — каталог исходного кода приложения. Включает подкаталоги.
- gen — каталог файлов, сгенерированных ADT в процессе разработки.
- assets — собрание произвольных файлов и каталогов.
- bin — скомпилированные файлы проекта
- libs — дополнительные библиотеки
- res — каталог хранения ресурсов проекта

Структура проекта. Каталог /res



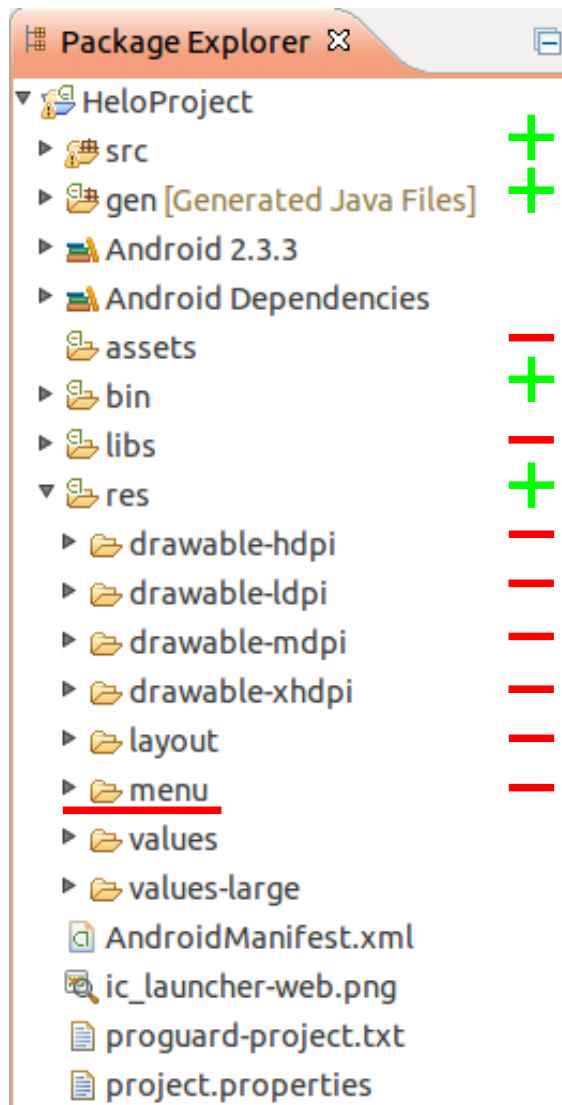
- /res/drawable — каталог содержащий изображения или файлы цветов.

Структура проекта. Каталог /res



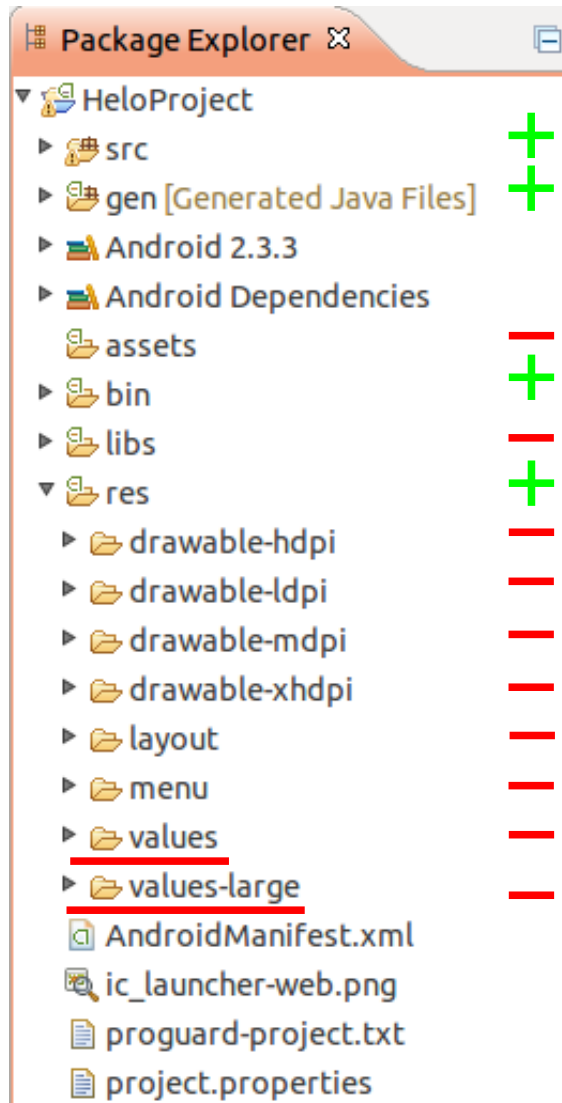
- /res/drawable - каталог содержащий изображения или файлы цветов.
- /res/layout — содержит файлы разметки

Структура проекта. Каталог /res



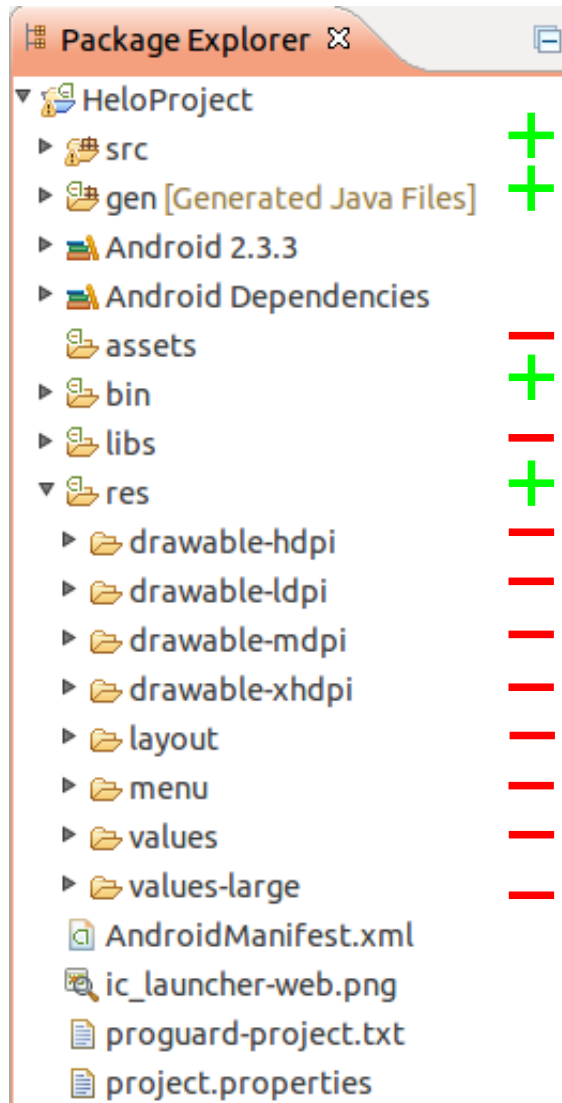
- /res/drawable - каталог содержащий изображения или файлы цветов.
- /res/layout — содержит файлы разметки
- /res/menu — меню, заданное через XML

Структура проекта. Каталог /res



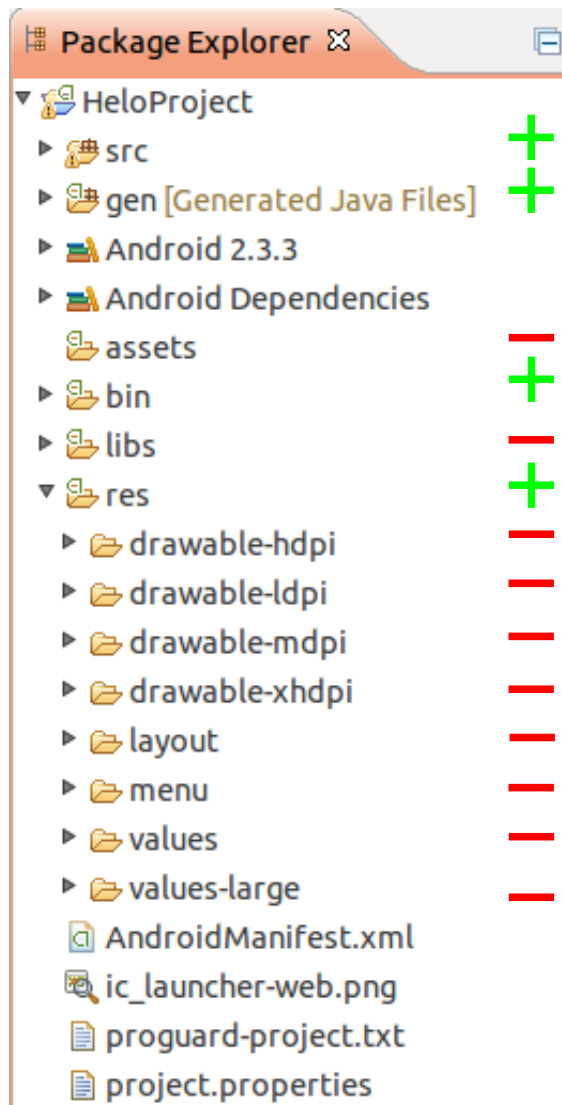
- /res/drawable - каталог содержащий изображения или файлы цветов.
- /res/layout — содержит файлы разметки
- /res/menu — меню, заданное через XML
- /res/values — содержит файлы строковых ресурсов

Структура проекта. Каталог /res



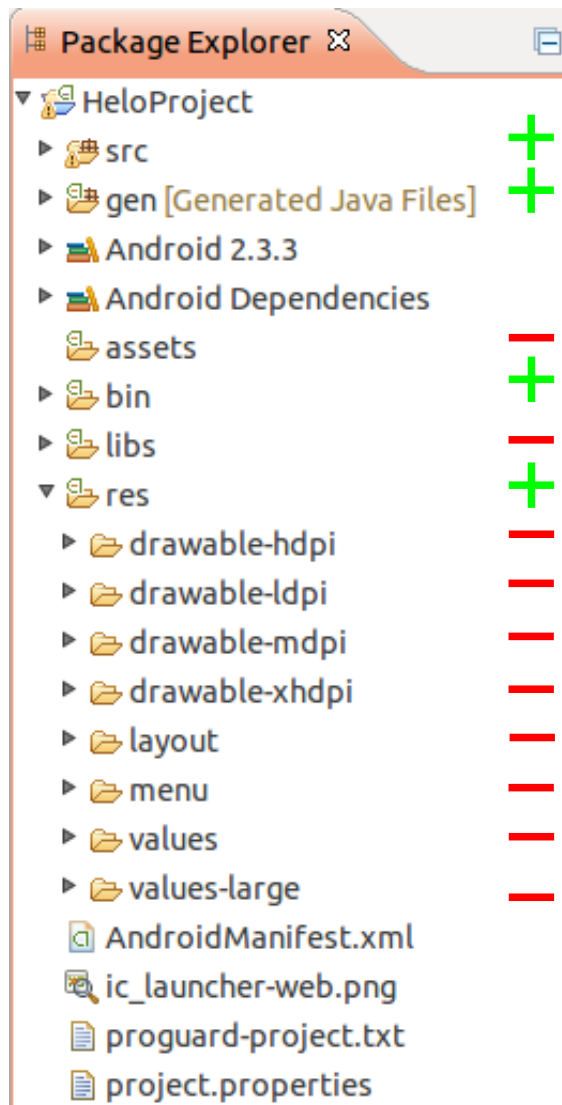
- /res/drawable - каталог содержащий изображения или файлы цветов.
- /res/layout — содержит файлы разметки
- /res/menu — меню, заданное через XML
- /res/values — содержит файлы строковых ресурсов
- /res/raw — любые данные не в формате XML

Структура проекта. Каталог /res



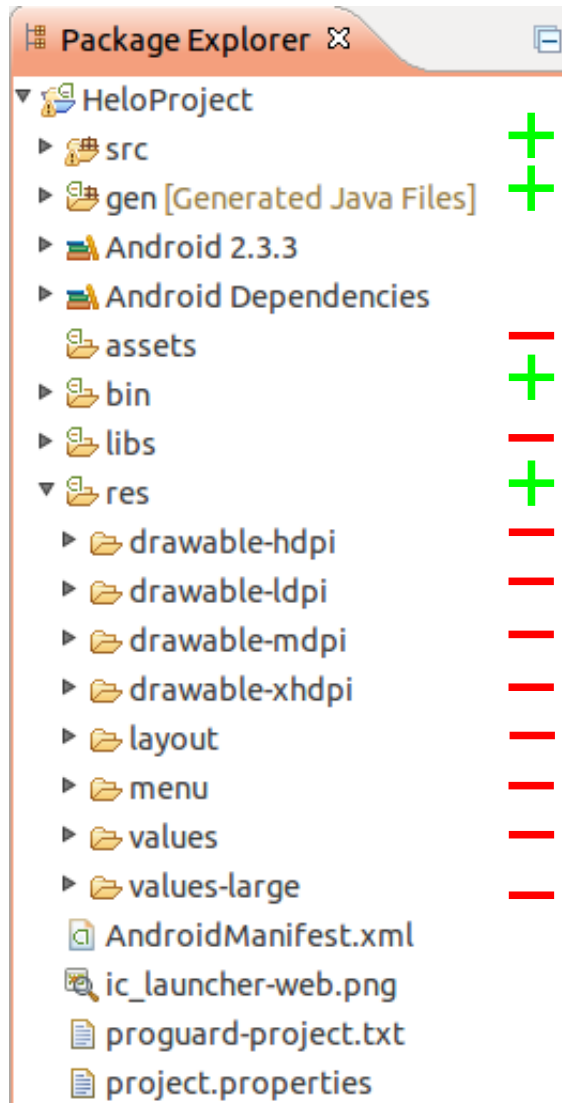
- /res/drawable - каталог содержащий изображения или файлы цветов.
- /res/layout — содержит файлы разметки
- /res/menu — меню, заданное через XML
- /res/values — содержит файлы строковых ресурсов
- /res/raw — любые данные не в формате XML
- /res/xml — произвольные XML-файлы

Структура проекта. Каталог /res



- /res/drawable - каталог содержащий изображения или файлы цветов.
- /res/layout — содержит файлы разметки
- /res/menu — меню, заданное через XML
- /res/values — содержит файлы строковых ресурсов
- /res/raw — любые данные не в формате XML
- /res/xml — произвольные XML-файлы
- /res/anim — файлы анимации в формате XML

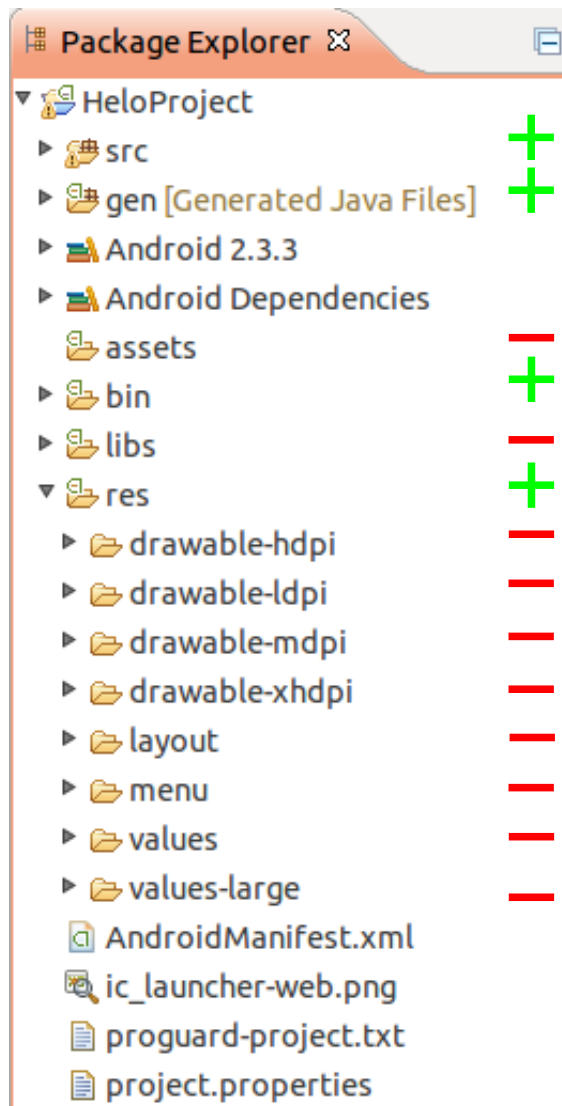
Структура проекта. Каталог /res



- Общая структура файлов выглядит следующим образом:

/res/values/strings.xml
/colors.xml
/dimens.xml
/styles.xml
/arrays.xml
/drawable/*.png
/*.jpg
/*.gif
/*.9.png
/anim/*.xml
/layout/*.xml
/raw/*.*
/xml/*.xml

Структура проекта. Каталог /res



- Обращение к ресурсам из программы:

R.anim — ID для файлов из каталога res/anim/ (анимация);

R.array — ID для массивов в файле arrays.xml из каталога res/values/;

R.bool — ID для булевых значений в файле с произвольным именем из каталога res/values/;

R.integer — ID для целочисленных значений в файле с произвольным именем из каталога res/values/;

R.color — ID для файлов colors.xml из каталога res/values/ (цвета);

R.dimen — ID для файлов dims.xml из каталога res/values/ (размеры);

R.drawable — ID для файлов из каталога res/drawable/ (изображения);

R.id — ID представлений и групп представлений для файлов XML-разметки из каталога res/layout/;

R.layout — ID для файлов разметки из каталога res/layout/;

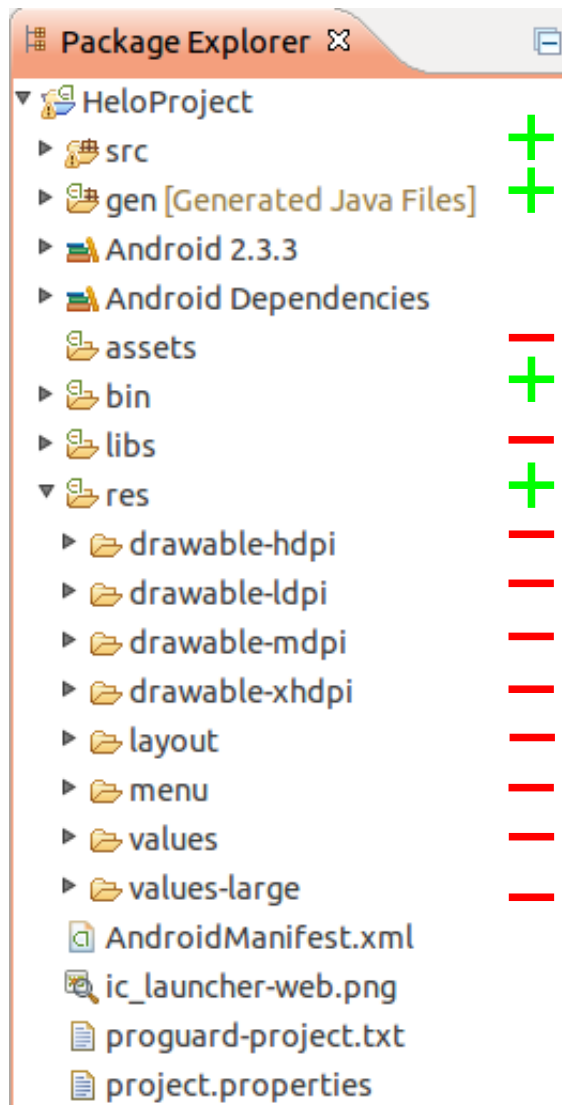
R.raw — ID для файлов из каталога res/raw/;

R.string — ID для файлов strings.xml из каталога res/values/ (строки);

R.style — ID для файлов styles.xml из каталога res/values/ (стили);

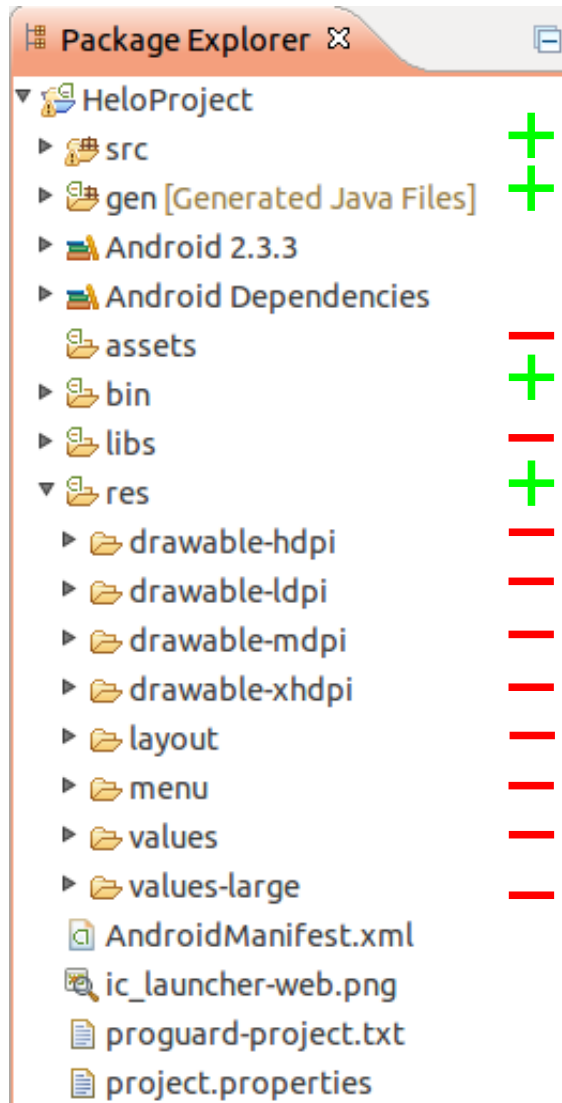
R.xml — ID для файлов из каталога res/xml/.

Структура проекта. Каталог /res



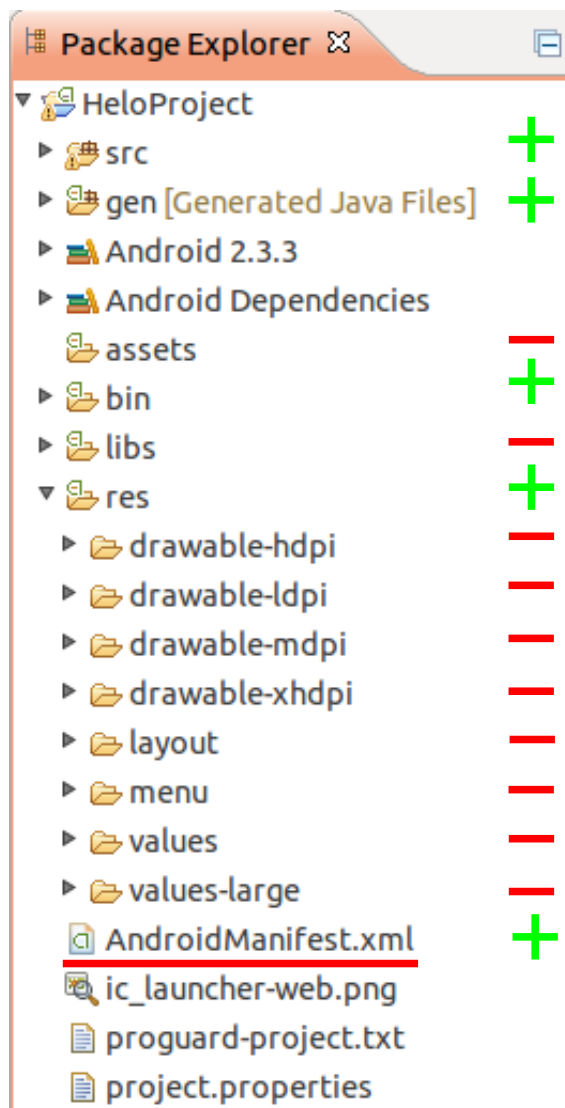
- Например:
 - drawable-en-rUS
 - drawable-en-keyshidden
 - values-ru
 - layout-sw420-land-mdpi-v10
 - drawable-long-land-notouch-nokeys
 - drawable-rUS-en
 - drawable-nonav-dpad

Структура проекта. Каталог /res



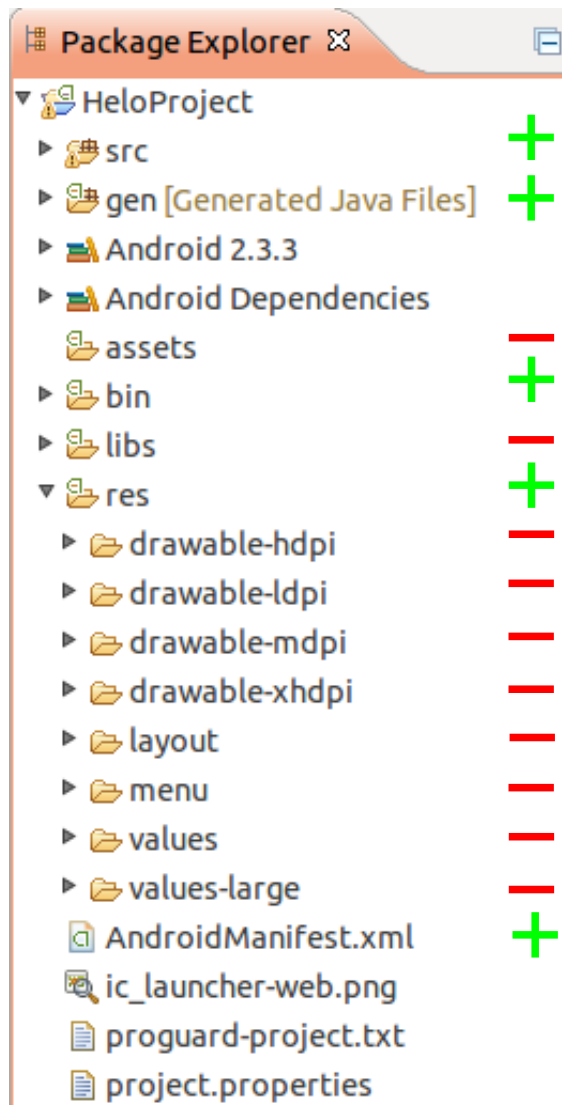
- Например:
 - drawable-en-rUS
 - drawable-en-keyshidden
 - values-ru
 - layout-sw420-land-mdpi-v10
 - drawable-long-land-notouch-nokeys
 - drawable-rUS-en (нарушен порядок)
 - drawable-nonav-dpad (несколько значений для одного квалификатора)

Структура проекта. AndroidManifest



```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.bstu.example.heloproject"
    android:versionCode="1"
    android:versionName="1.0" >
    <uses-sdk
        android:minSdkVersion="8"
        android:targetSdkVersion="15" />
    <application
        android:icon="@drawable/ic_launcher"
        android:label="@string/app_name"
        android:theme="@style/AppTheme" >
        <activity
            android:name=".MainActivity"
            android:label="@string/app_name" >
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER"
            />
            </intent-filter>
        </activity>
    </application>
</manifest>
```

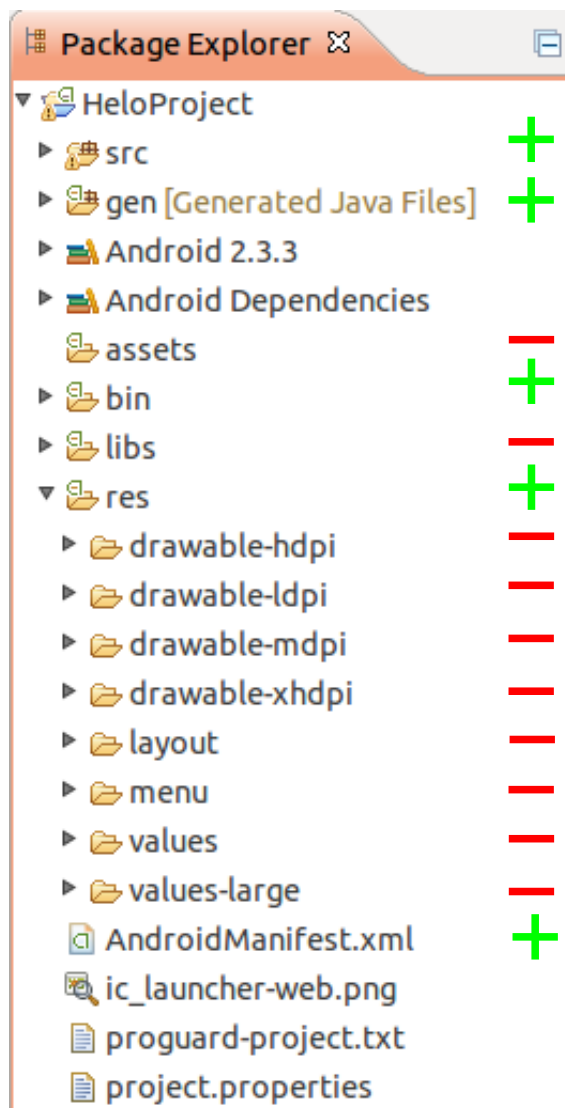
Структура проекта. AndroidManifest



```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.bstu.example.heloproject"
    android:versionCode="1"
    android:versionName="1.0" >
```

- Корневой, обязательный элемент. Может быть только один элемент `<manifest>` в файле. Должен обязательно содержать элемент `<application>`
- Обязательные атрибуты:
 - **xmlns:android** — должен всегда содержать значение «<http://schemas.android.com/apk/res/android>»
 - **package** — полное имя пакета для приложения.
- **android:installLocation** — место установки приложения по-умолчанию. Возможные варианты:
 - **internalOnly**
 - **auto**
 - **preferExternal**

Структура проекта. AndroidManifest

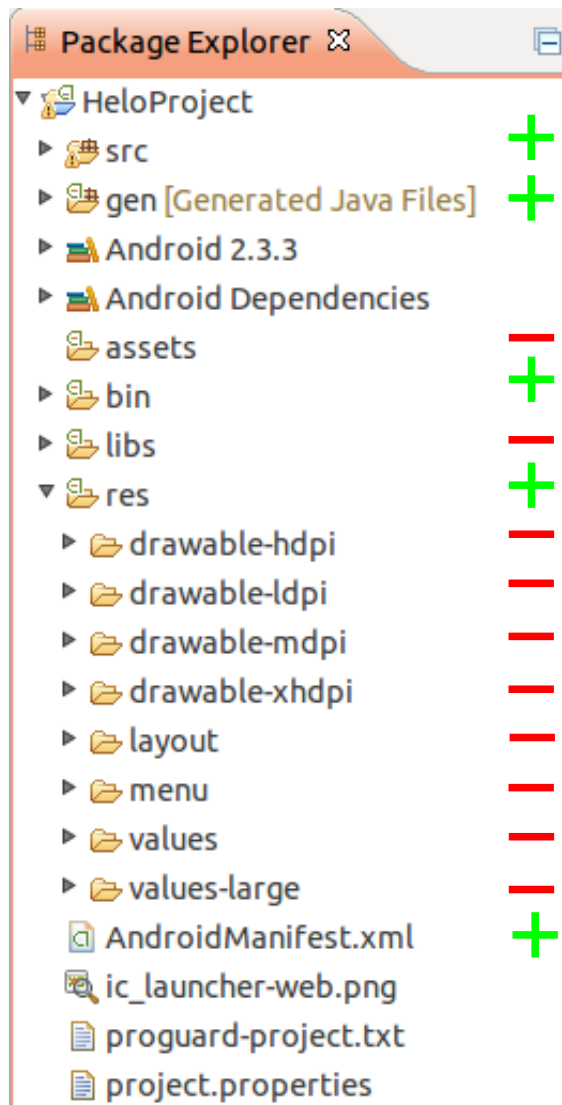


<uses-sdk

```
android:minSdkVersion="8"  
android:targetSdkVersion="15" />
```

- Несмотря на имя, указывает на версию API, а не SDK
- **android:minSdkVersion** — Android не установит приложение, если версия API устройства ниже указанного значения. Если не задано, то по-умолчанию присваивается значение 1.
- **android:targetSdkVersion** — указывает на версию API, на которой приложение тестировалось и система не должна выполнять какие-либо действия для совместимости.
- **android:maxSdkVersion** — максимальная версия API. Приложение не установится и может быть удалено после обновления системы (не рекомендуется использовать). Версии Android старше 2.0.1 (Level 6) не проверяют этот параметр.

Структура проекта. AndroidManifest

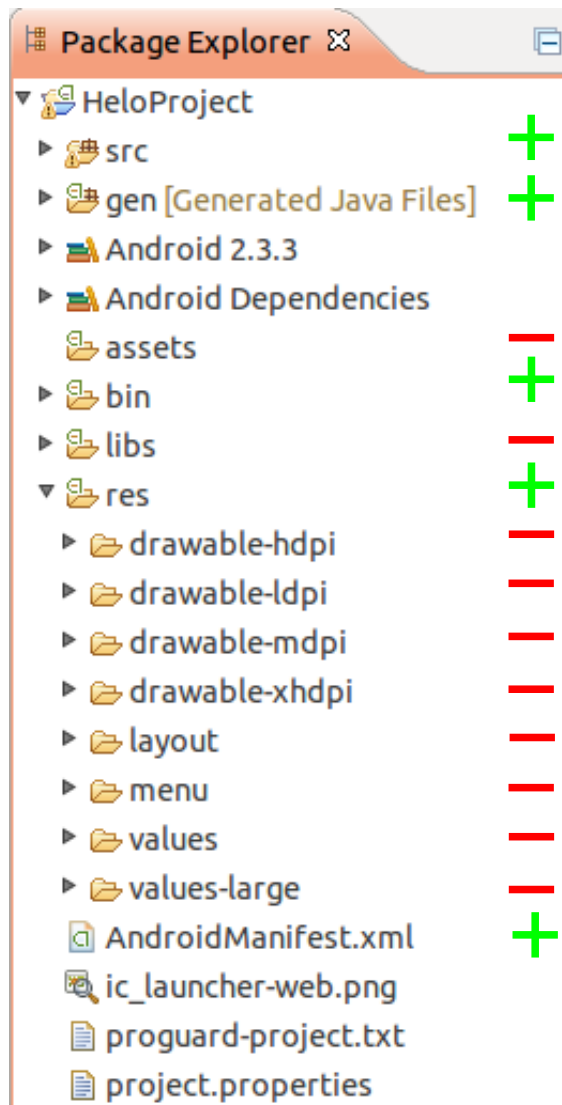


<application

```
android:icon="@drawable/ic_launcher"  
android:label="@string/app_name"  
android:theme="@style/AppTheme" >
```

- Также обязательный элемент файла AndroidManifest.xml. Может быть только один такой элемент.
- Атрибуты:
 - android:icon
 - android:label
 - android:theme
 - android:name
 - android:permission
 - android:debuggable

Структура проекта. AndroidManifest

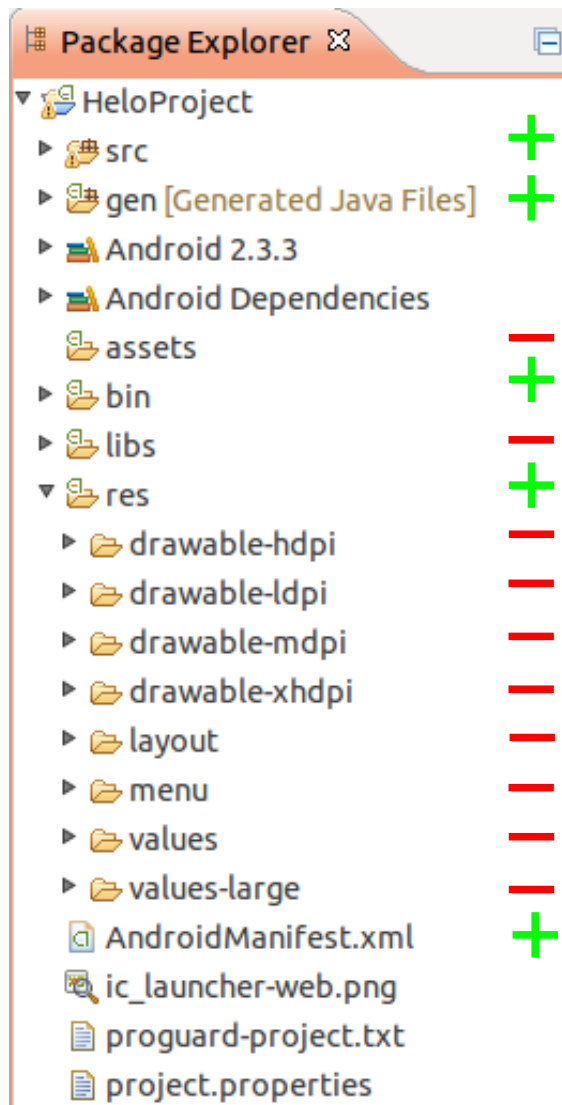


<activity

```
    android:name=".MainActivity"  
    android:label="@string/app_name" >
```

- Все созданные в приложении activity должны быть описаны в файле манифеста этим тэгом. Не объявленные activity неизвестны системе и не могут быть запущены.
- Атрибуты:
 - android:name — обязательный атрибут
 - android:label
 - android:configChanges (onConfigurationChanged())
 - android:screenOrientation
 - android:launchMode

Структура проекта. AndroidManifest

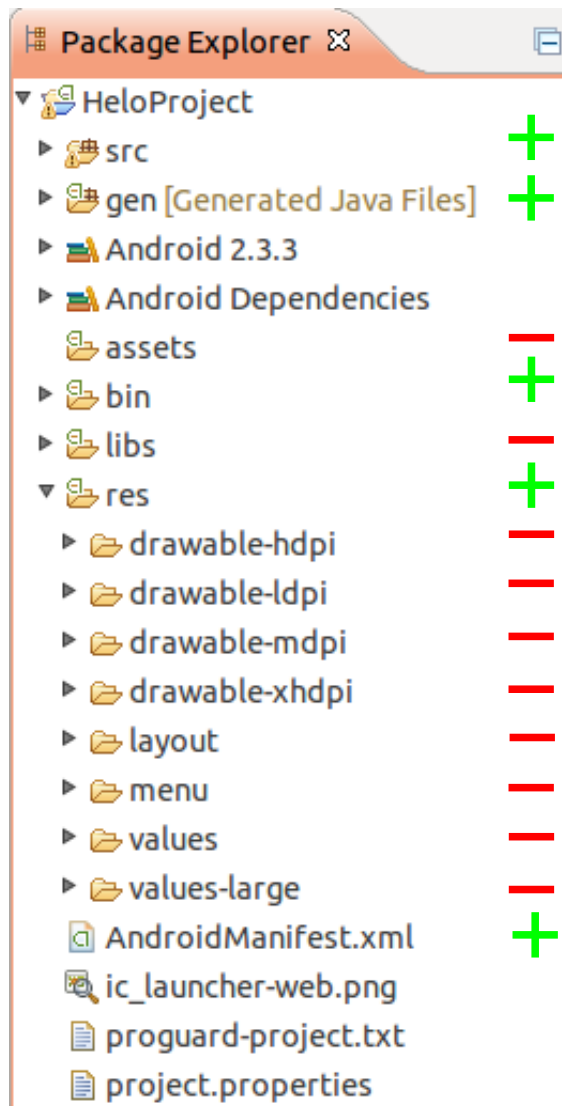


<intent-filter>

```
<action android:name="android.intent.action.MAIN" />  
<category android:name="android.intent.category.LAUNCHER" />
```

- Определяет типы намерений (intent), на которые родительский элемент (activity, service, broadcast receiver) могут реагировать.
- Атрибуты:
 - android:icon
 - android:label
 - android:priority
- Обязательный элемент **<action>**. Intent filter должен содержать хотя бы один элемент <action>
 - атрибут android:name — может содержать одно из стандартных значений или уникальное пользовательское значение.
- Дополнительные, необязательные элементы <category> и <data>.

Структура проекта. AndroidManifest



`<uses-permission android:name="string" />`

- Указывает на права, необходимые приложению для корректной работы.
- Пользователь предоставляет приложению эти права во время установки.
- Наиболее распространенные разрешения
- **INTERNET** - доступ к интернету
- **READ_CONTACTS** - чтение (но не запись) данных из адресной книги пользователя
- **WRITE_CONTACTS** - запись (но не чтение) данных из адресной книги пользователя
- **RECEIVE_SMS** - обработка входящих SMS
- **ACCESS_COARSE_LOCATION** - использование приблизительного определения местонахождения при помощи вышек сотовой связи или точек доступа Wi-Fi
- **ACCESS_FINE_LOCATION** - точное определение местонахождения при помощи GPS

Android. Литература

- **Pro Android 2** — Sayed Y. Hashimi, Satya Komatineni, Dave MacLean

Android. Литература

- **Pro Android 2** — Sayed Y. Hashimi, Satya Komatineni, Dave MacLean
- d.android.com

Android. Литература

- **Pro Android 2** — Sayed Y. Hashimi, Satya Komatineni, Dave MacLean
- d.android.com
- stackoverflow.com

Android. Литература

- **Pro Android 2** — Sayed Y. Hashimi, Satya Komatineni, Dave MacLean
- d.android.com
- stackoverflow.com
- google.com

Android. Литература

- **Pro Android 2** — Sayed Y. Hashimi, Satya Komatineni, Dave MacLean
- d.android.com
- stackoverflow.com
- google.com
- Рето Майер - **Android 2. Программирование приложений**, 2011
- Голощапов А. Л. - **Google Android программирование для мобильных устройств**, 2011