

Android Studio. Создание простого проекта

Рассматриваемые вопросы

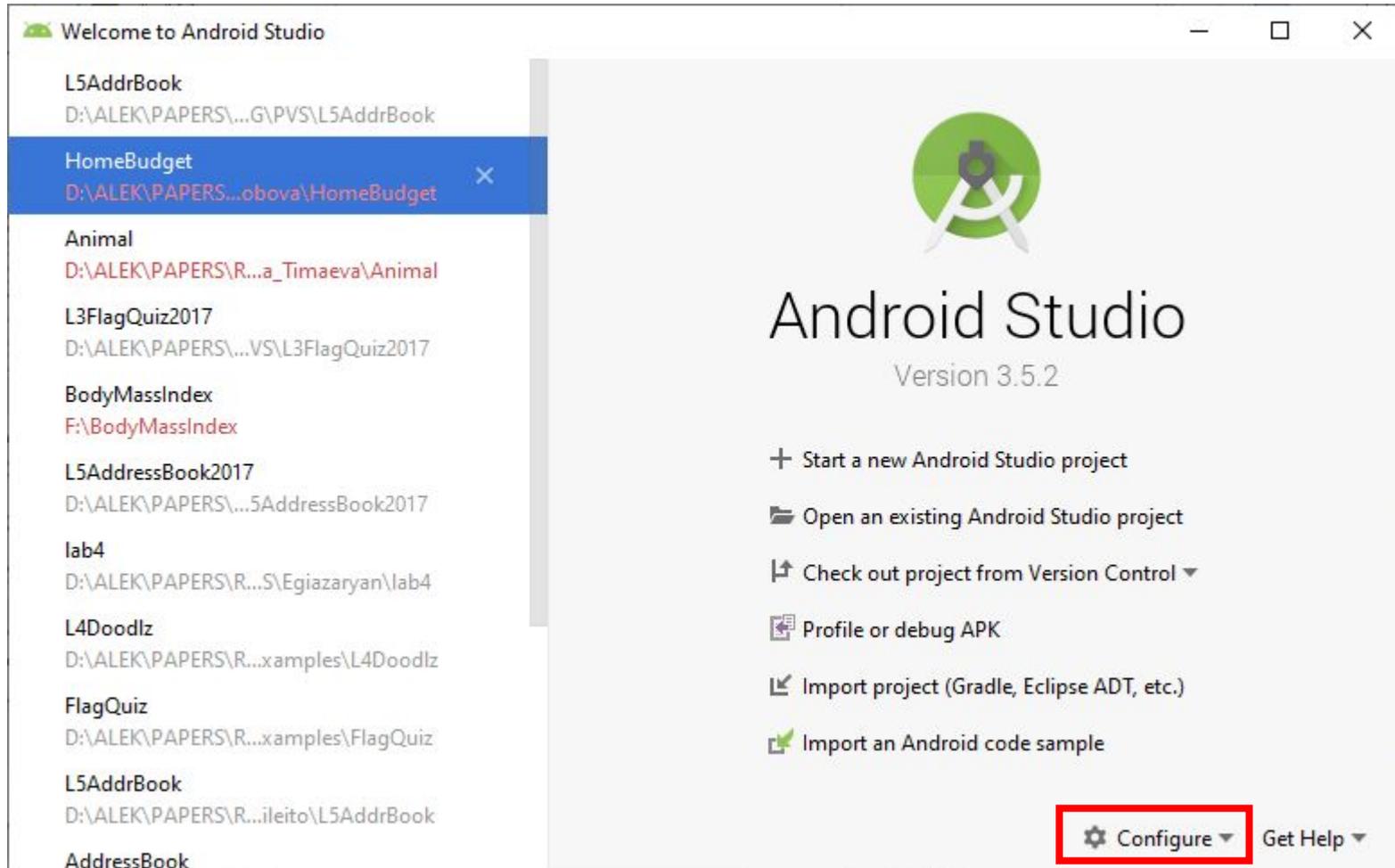
- Настройка Android Studio при нехватке ресурсов
- Создание нового проекта приложения в среде разработки Android Studio
- Визуальное построение графического интерфейса пользователя в макетном редакторе
- Вывод текста и графики в графическом интерфейсе
- Изменение свойств компонентов графического интерфейса
- Создание простого приложения Android и его выполнение в эмуляторе

Настройка AS при нехватке ресурсов

Системные требования Android Studio 4.x.

- Microsoft® Windows® 7/8/10 (64-битная)
- Минимум 4 ГБ RAM, рекомендуется 8 ГБ RAM;
- Минимум 2 ГБ свободного дискового пространства,
рекомендуется 4 ГБ (500 МБ для IDE + 1.5 ГБ для Android SDK и образа системы эмулятора)
- Минимальное разрешение экрана 1280 x 800

Настройка AS при нехватке ресурсов



Настройка AS при нехватке

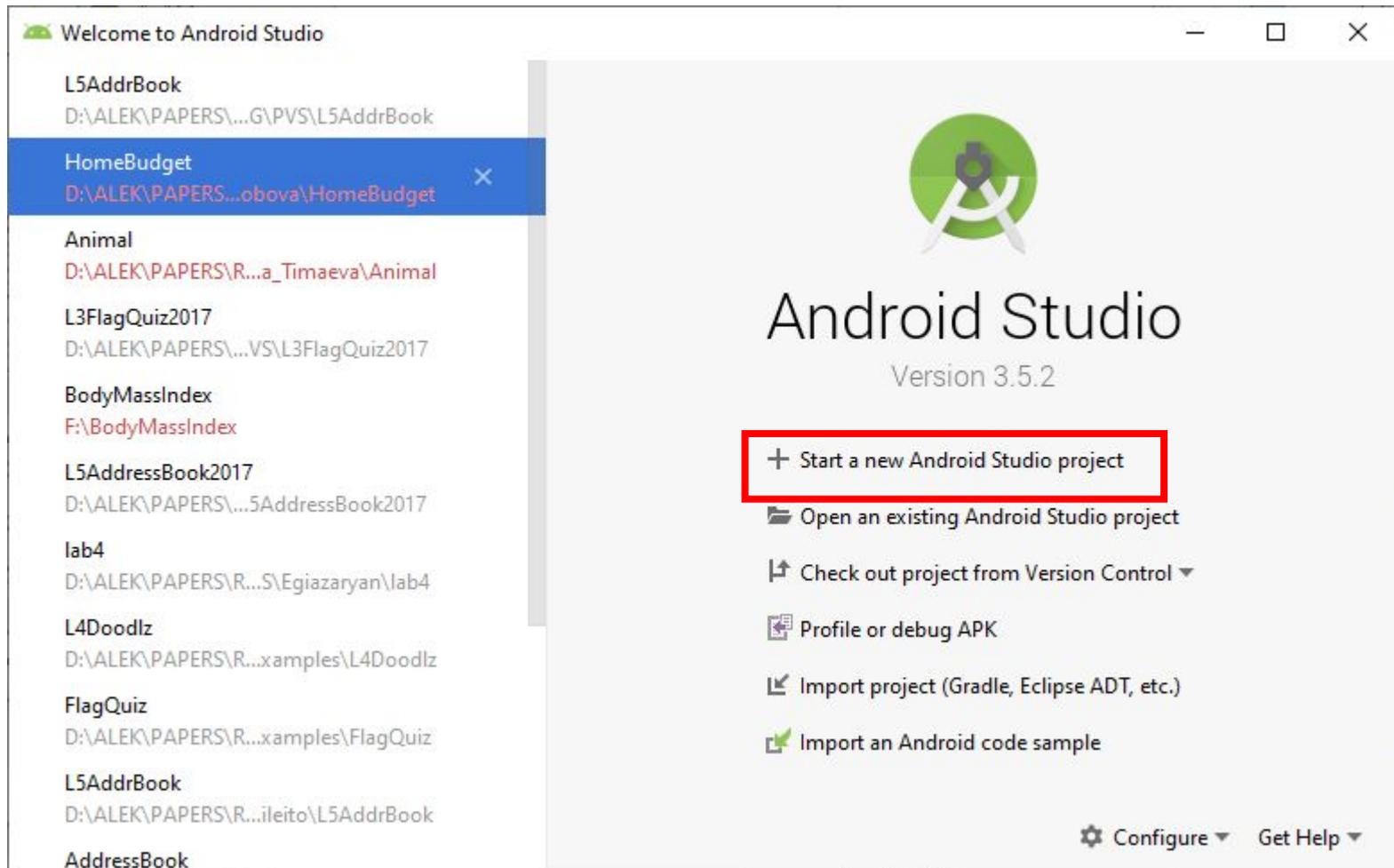
ресурсов

- Уменьшить размер доступной для AS кучи до 512 МБ
- Обновить плагины
- Включить режим экономии энергии
- Отключить ненужные проверки lint
- Отлаживать на физическом устройстве
- Подключать только необходимые сервисы Google Play
- Уменьшить доступный размер кучи для Gradle
- Не разрешать параллельную компиляцию

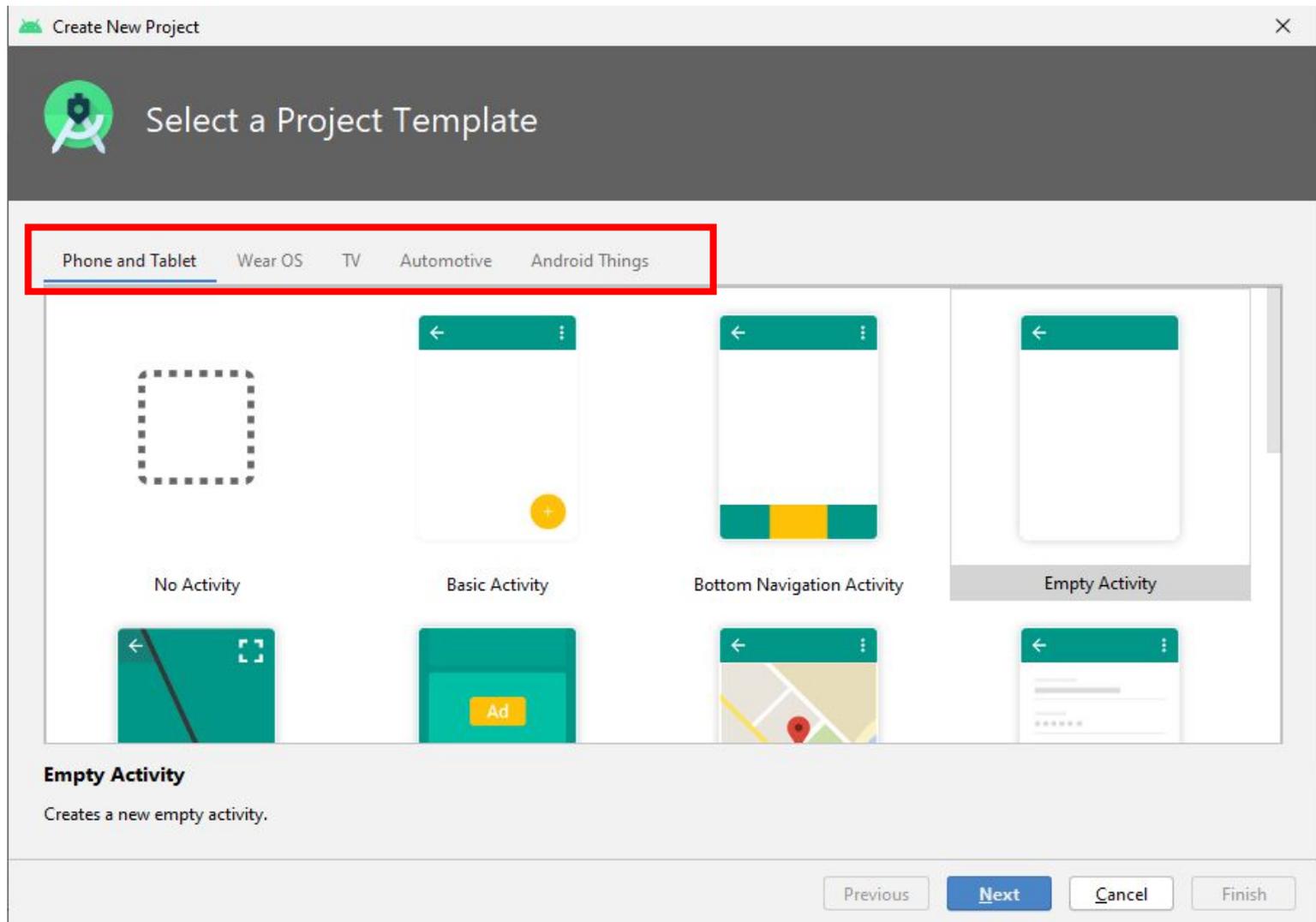
Подробнее:

https://developer.android.com/studio/intro/studio-config.html#low_memory

Создание проекта



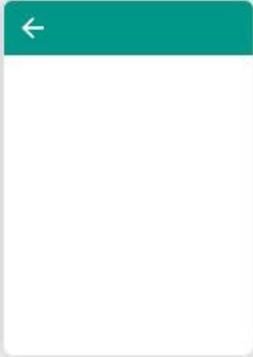
Создание проекта



Создание проекта

Create New Project

Configure Your Project



Empty Activity
Creates a new empty activity.

Name
Welcome2020

Package name
com.somewhere.welcome2020

Save location
D:\ALEK\PAPERS\REITING\PVS\2020\Welcome2020

Language
Java

Minimum SDK
API 23: Android 6.0 (Marshmallow)

уникальный идентификатор приложения в Google Play

недопустимы пробелы

? Your app will run on approximately **84,9%** of devices.
[Help me choose](#)

Use legacy android.support libraries **?**

Previous Next Cancel **Finish**

Версии Android SDK и уровни

API

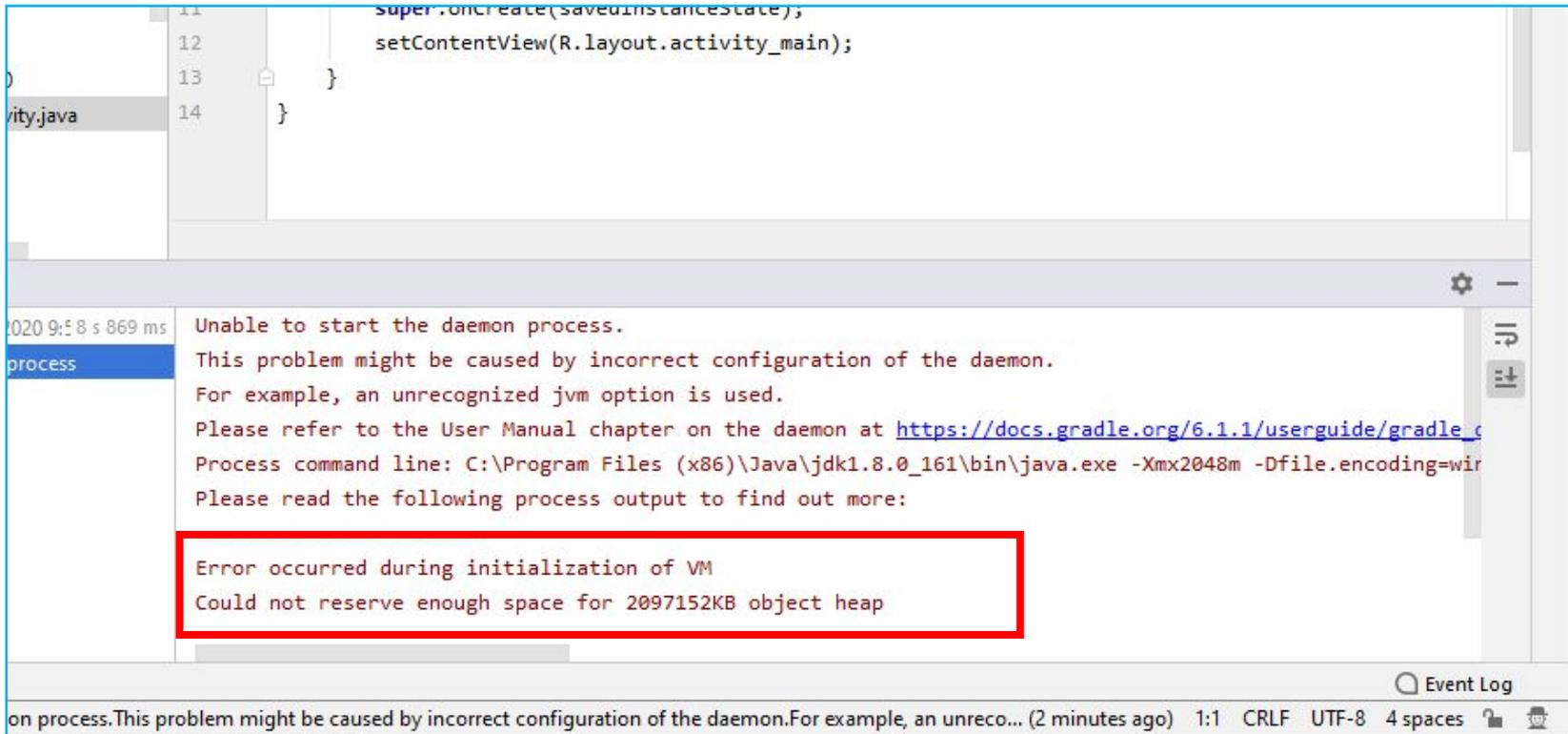
Версия SDK	Уровень API
11.0 (R)	30
10.0 (Q)	29
9.0 (Pie – Пирог)	28
8.0, 8.1 (Oreo – Орео)	26, 27
7.0, 7.1.1 (Nougat – Нуга)	24, 25
6.0 (Marshmallow – Зефир)	23
5.0, 5.1 (Lollipop – Леденец)	21, 22
4.4 (KitKat – Кит-Кат)	19
4.1.x, 4.2.x, 4.3 (Jelly Bean – Жевательная Конфета)	16, 17, 18
4.0.3-4.0.4 (IceCreamSandwich – СэндвичМороженое)	15
2.3, 2.3.3-2.3.7 (Gingerbread – Имбирный Пряник)	9, 10

Меньшие значения Minimum SDK для проекта позволят приложению выполняться на большем количестве устройств

Шаблоны Activity (выборочно)

Шаблон	Описание
Blank Activity	Используется для одноэкранных приложений, в которых большая часть графического интерфейса строится разработчиком. Включает панель действий в верхней части приложения, в которой выводится имя приложения, а также могут отображаться элементы управления для взаимодействия с приложением. Также на панели отображается кнопка <code>FloatingActionButton</code> материального дизайна
Fullscreen Activity	Используется для одноэкранных приложений (по аналогии с <code>Blank Activity</code>), занимающих весь экран, но с возможностью переключения режима видимости панели состояния устройства и панели действий приложения
Master/Detail Flow	Используется для приложений с главным списком, из которого пользователь может выбрать один вариант для просмотра подробной информации (как во встроенных приложениях <code>Email</code> и <code>Contacts</code>). Включает основную логику выбора элементов в главном списке и отображения детализированного представления этого элемента. На планшетах главный список и детализация выводятся рядом друг с другом на одном экране. На телефонах главный список выводится на одном экране, а при выборе варианта детализация отображается на отдельном экране

Создание проекта



The screenshot shows an IDE interface. The top part is a code editor with a file named 'activity.java' open. The code is as follows:

```
11 super.onCreate(savedInstanceState);
12 setContentView(R.layout.activity_main);
13 }
14 }
```

Below the code editor is a console window. The console shows the following output:

```
020 9:58 s 869 ms
process Unable to start the daemon process.
This problem might be caused by incorrect configuration of the daemon.
For example, an unrecognized jvm option is used.
Please refer to the User Manual chapter on the daemon at https://docs.gradle.org/6.1.1/userguide/gradle\_c
Process command line: C:\Program Files (x86)\Java\jdk1.8.0_161\bin\java.exe -Xmx2048m -Dfile.encoding=wir
Please read the following process output to find out more:

Error occurred during initialization of VM
Could not reserve enough space for 2097152KB object heap
```

The error message is highlighted with a red box. At the bottom of the console window, there is a status bar with the text: 'on process.This problem might be caused by incorrect configuration of the daemon.For example, an unreco... (2 minutes ago) 1:1 CRLF UTF-8 4 spaces'.

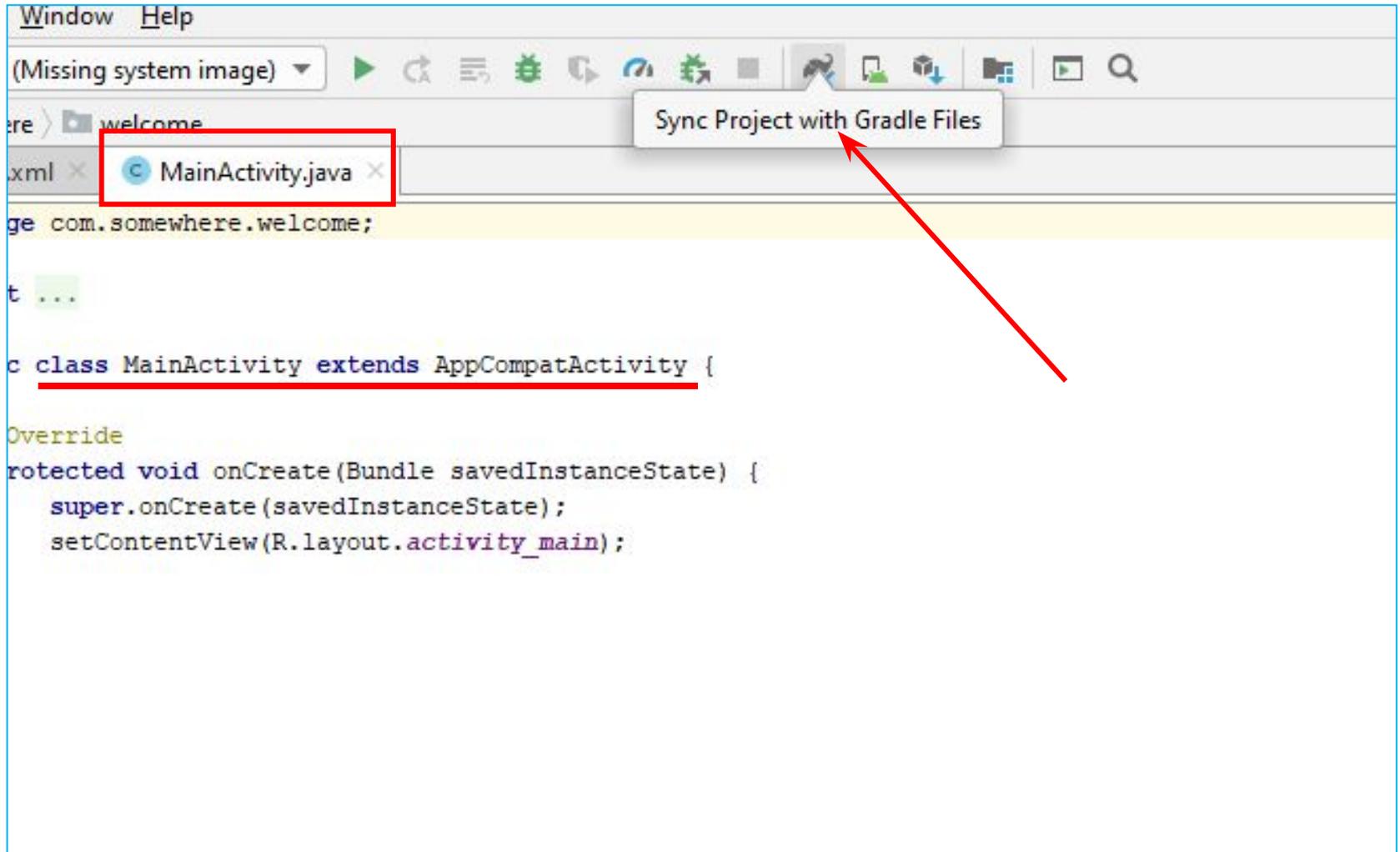
Создание проекта

The screenshot displays the IDE interface with the following components:

- Resource Manager:** Lists files like `gradlew`, `gradlew.bat`, `local.properties`, `settings.gradle`, `Welcome.iml`, and **Gradle Scripts** (including `build.gradle`, `gradle-wrapper.properties`, `gradle.properties`, `settings.gradle`, and `local.properties`).
- Build Variants:** Shows `Build: Sync`.
- Build Log:** Displays a failure: `Welcome: sync failed at 21.11.2019 21:39 with 1 error`. The error details are:
 - Starting Gradle Daemon
 - Gradle Sync Issues: (1 error)
 - Unable to start the daemon process. This problem might be cau
- Terminal:** Shows the error output:

```
ERROR: Unable to start the daemon process.
This problem might be caused by incorrect configuration o
For example, an unrecognized jvm option is used.
Please refer to the User Manual chapter on the daemon at
Process command line: C:\Program Files (x86)\Java\jdk1.8.
Please read the following process output to find out more
-----
Error occurred during initialization of VM
Could not reserve enough space for 1572864KB object heap
```
- Code Editor:** Shows the `gradle.properties` file with the line `org.gradle.jvmargs=-Xmx1024m` highlighted in yellow and circled in red.

Создание проекта



Создание проекта

Create New Project

Customize the Activity

Creates a new empty activity

Activity Name: MainActivity

Generate Layout File

Layout Name: activity_main

Backwards Compatibility (AppCompat)

Empty Activity

The name of the activity class to create

Previous Next Cancel Finish

имя subclasses Activity, управляющего выполнением приложения

имя XML-файла макета

Resource Manager

- app
 - Gradle Scripts
 - build.gradle (Project: Welcome2020)
 - build.gradle (Module: app)
 - gradle-wrapper.properties (Gradle Version)
 - proguard-rules.pro (ProGuard Rules for a
 - gradle.properties (Project Properties)
 - settings.gradle (Project Settings)
 - local.properties (SDK Location)

1: Project

2: Favorites

Build Variants

Structure

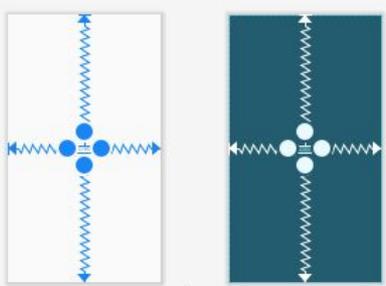
activity_main.xml x MainActivity.java x

Palette

- Common
 - Ab TextView
- Text
 - Button
- Buttons
 - ImageView
- Widgets
 - RecyclerView
- Layouts
 - <> <fragment>
- Container
 - ScrollView
 - Switch

Component Tree

- ConstraintLayout
 - Ab TextView "Hello World!"



Code Split Design

Attributes

Ab <unnamed> TextView

id

Declared Attributes

layout_width	wrap_content	0
layout_height	wrap_content	0
layout_constraintB...	parent	0
layout_constraintL...	parent	0
layout_constraintRi...	parent	0
layout_constraintT...	parent	0
text	Hello World!	0

Layout

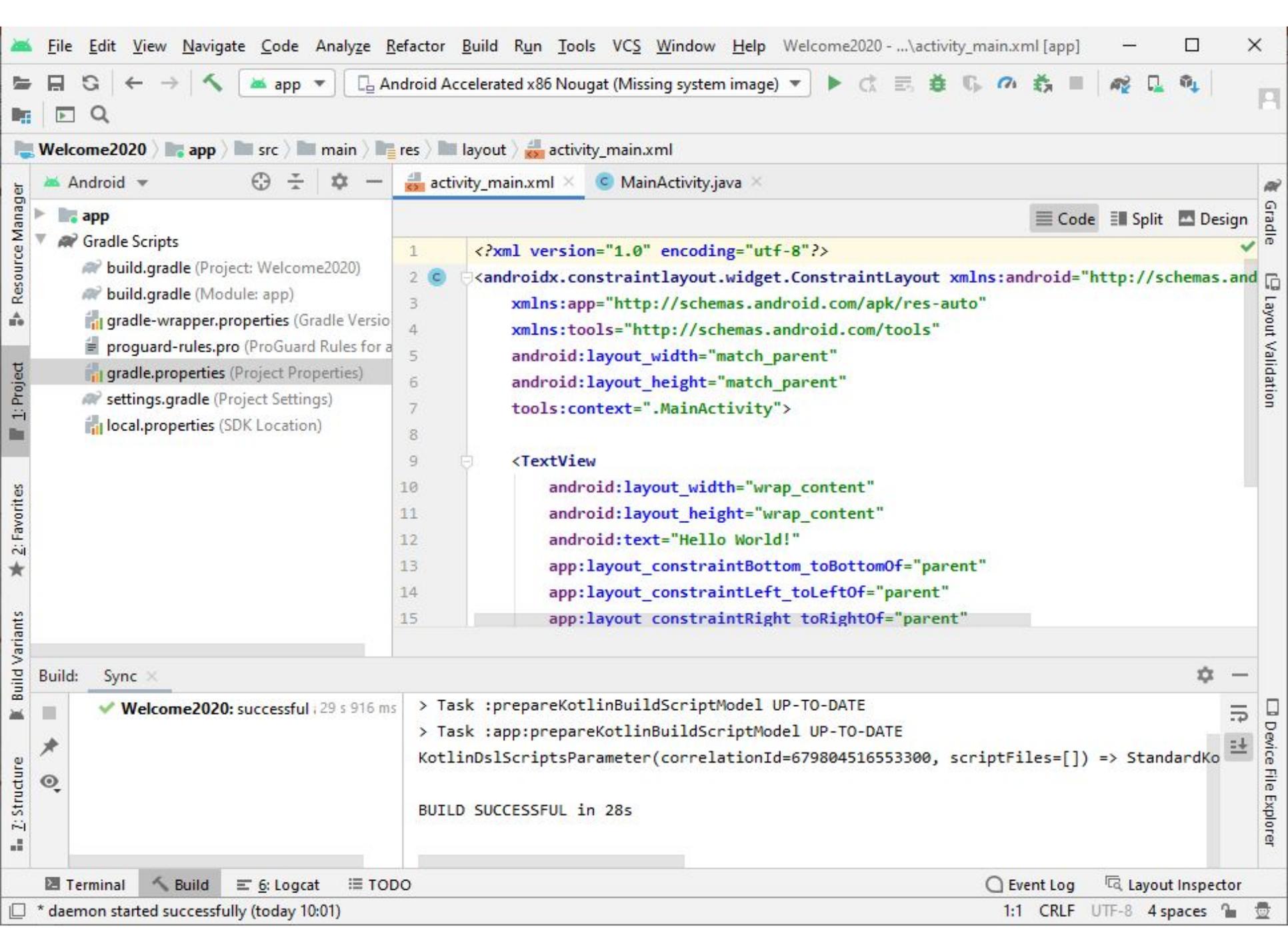
Constraint Widget

0

Build: Sync x

Welcome2020: successful at 29.09.2020 10:01 29 s 916 ms

```
> Task :prepareKotlinBuildScriptModel UP-TO-DATE
> Task :app:prepareKotlinBuildScriptModel UP-TO-DATE
KotlinDslScriptsParameter(correlationId=679804516553300, scriptFiles=[]) => StandardKotlinDslScriptsModel(scripts=[], commonMod
BUILD SUCCESSFUL in 28s
```



Resource Manager

- app
 - Gradle Scripts
 - build.gradle (Project: Welcome2020)
 - build.gradle (Module: app)
 - gradle-wrapper.properties (Gradle Version)
 - proguard-rules.pro (ProGuard Rules for a
 - gradle.properties (Project Properties)
 - settings.gradle (Project Settings)
 - local.properties (SDK Location)

1: Project

2: Favorites

Build Variants

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res-auto"
3     xmlns:app="http://schemas.android.com/apk/res-auto"
4     xmlns:tools="http://schemas.android.com/tools"
5     android:layout_width="match_parent"
6     android:layout_height="match_parent"
7     tools:context=".MainActivity">
8
9     <TextView
10         android:layout_width="wrap_content"
11         android:layout_height="wrap_content"
12         android:text="Hello World!"
13         app:layout_constraintBottom_toBottomOf="parent"
14         app:layout_constraintLeft_toLeftOf="parent"
15         app:layout_constraintRight_toRightOf="parent">
```

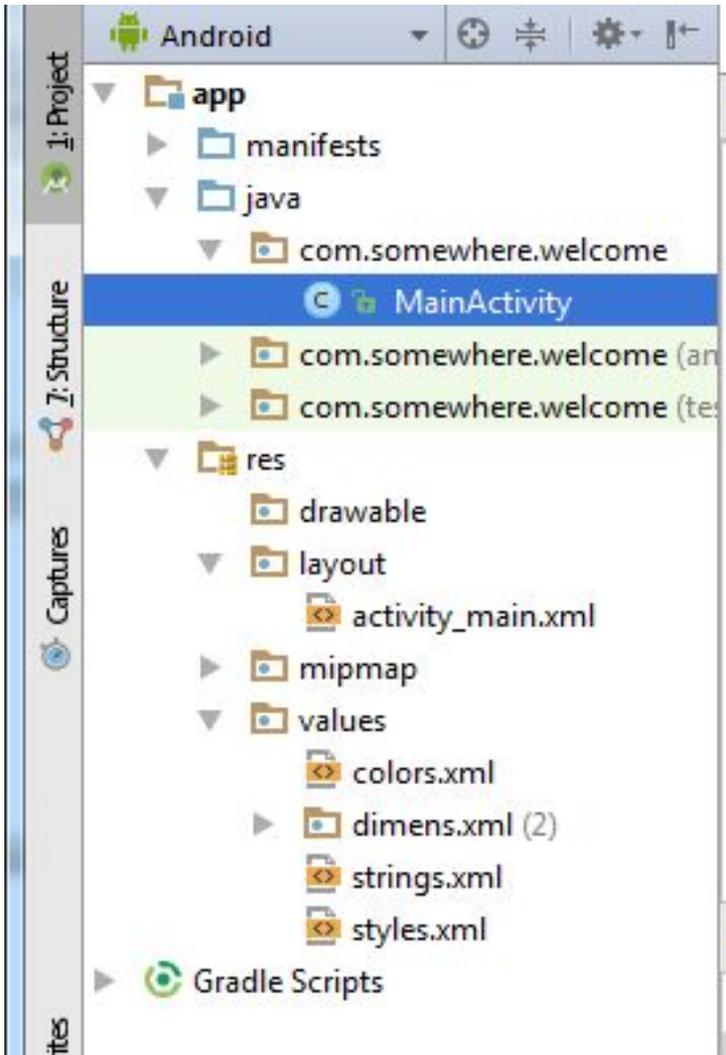
Build: Sync

✓ Welcome2020: successful ; 29 s 916 ms

```
> Task :prepareKotlinBuildScriptModel UP-TO-DATE
> Task :app:prepareKotlinBuildScriptModel UP-TO-DATE
KotlinDslScriptsParameter(correlationId=679804516553300, scriptFiles=[]) => StandardKo
BUILD SUCCESSFUL in 28s
```

Device File Explorer

Файлы ресурсов



Файлы макетов — такие как `activity_main.xml`, — считаются ресурсами приложения и хранятся в папке `res` проекта.

В этой папке находятся вложенные папки для разных типов ресурсов.

Файлы ресурсов

Папка	Описание
drawable	Папки с именами, начинающимися с <code>drawable</code> , обычно содержат изображения. В них также могут храниться файлы XML, описывающие геометрические фигуры и другие типы экранных объектов (например, изображения, представляющие нажатое и ненажатое состояние кнопки)
layout	Папки с именами, начинающимися с <code>layout</code> , содержат файлы XML с описанием графического интерфейса, например файл <code>activity_main.xml</code>
values	Папки с именами, начинающимися с <code>values</code> , содержат файлы со значениями массивов (<code>arrays.xml</code>), цветов (<code>colors.xml</code>), размеров (<code>dimen.xml</code> ; такие значения, как ширина, высота и размеры шрифтов), строки (<code>strings.xml</code>) и стили (<code>styles.xml</code>). Эти имена файлов используются по общепринятым соглашениям, но не являются обязательными: на самом деле все ресурсы этих типов можно разместить в одном файле. Все эти жестко фиксированные данные рекомендуется определять в виде ресурсов, чтобы их можно было изменить без модификации исходного кода Java. Например, если ссылки на некую метрику встречаются в нескольких местах кода, проще один раз отредактировать файл ресурсов, чем искать все вхождения фиксированного значения в исходных файлах Java

Макетный редактор

The screenshot displays the Android Studio layout editor for the `activity_main.xml` file. The main workspace shows a design view of a `TextView` widget on a Nexus 6 device. The widget is a dark teal rectangle with the text "Hello World!". The design view is overlaid on a light gray background with blue dashed lines representing constraints. A red arrow points to the `0dp` constraint value in the top toolbar. The right-hand side of the interface features the **Attributes** panel, which lists the widget's properties, including `id`, `layout_wid...`, `layout_hei...`, `layout_con...`, and `text`. Below the attributes is the **Layout** section, which shows a `Constraint Widget` diagram with numerical values for constraints.

Resource Manager

- app
 - manifests
 - java
 - com.somewhere.v
 - MainActivity
 - res
 - drawable
 - layout
 - activity_main.x
 - mipmap
 - values
 - colors.xml
 - strings.xml
 - styles.xml
- Gradle Scripts
 - build.gradle (Project)
 - build.gradle (Module)
 - gradle-wrapper.prope
 - proguard-rules.pro (P
 - gradle.properties (Pro
 - settings.gradle (Projec
 - local.properties (SDK

Component Tree

Design

Code Split Design

Attributes

Ab <unnamed> TextView

id

Declared Attributes

Attribute	Value
layout_wid...	wrap_conter
layout_hei...	wrap_conter
layout_con...	parent
text	Hello World!

Layout

Constraint Widget

0

0

0

0

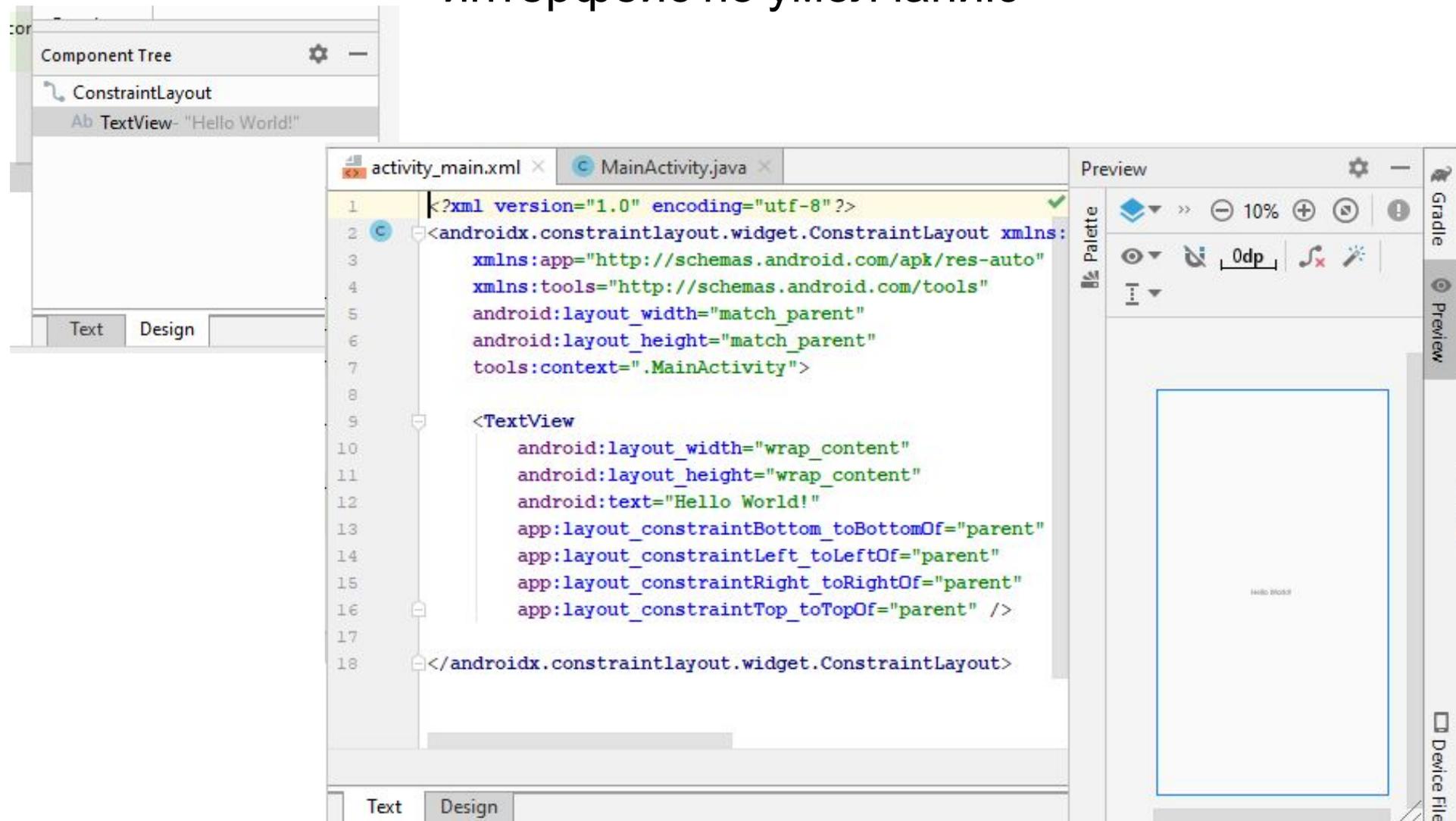
Terminal Build Logcat TODO

Event Log Layout Inspector

1:1 CRLF UTF-8 4 spaces

* daemon started successfully (today 10:01)

Построение графического интерфейса интерфейс по умолчанию



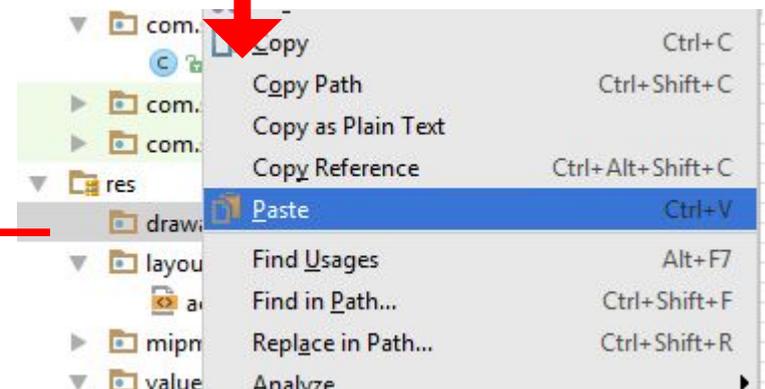
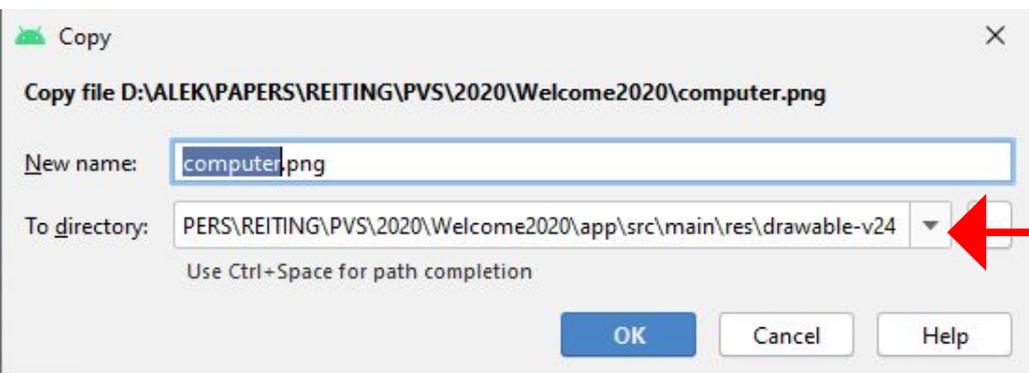
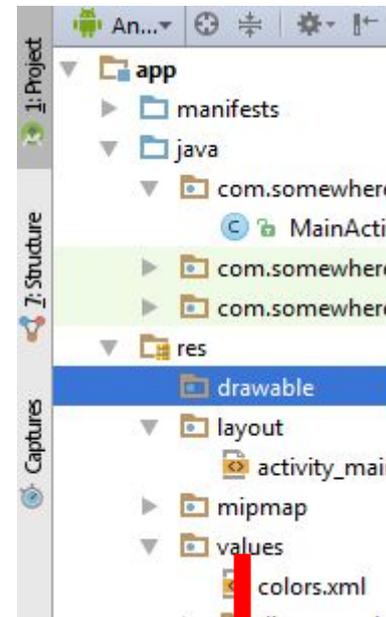
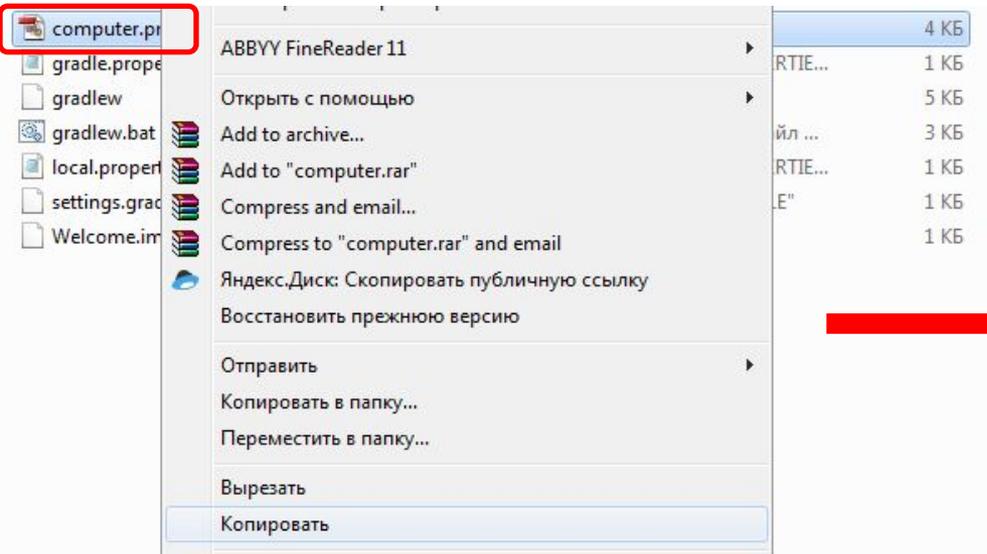
Построение графического интерфейса. Добавление изображения в проект

- Имена файлов изображений и других ресурсов должны записываться в нижнем регистре!

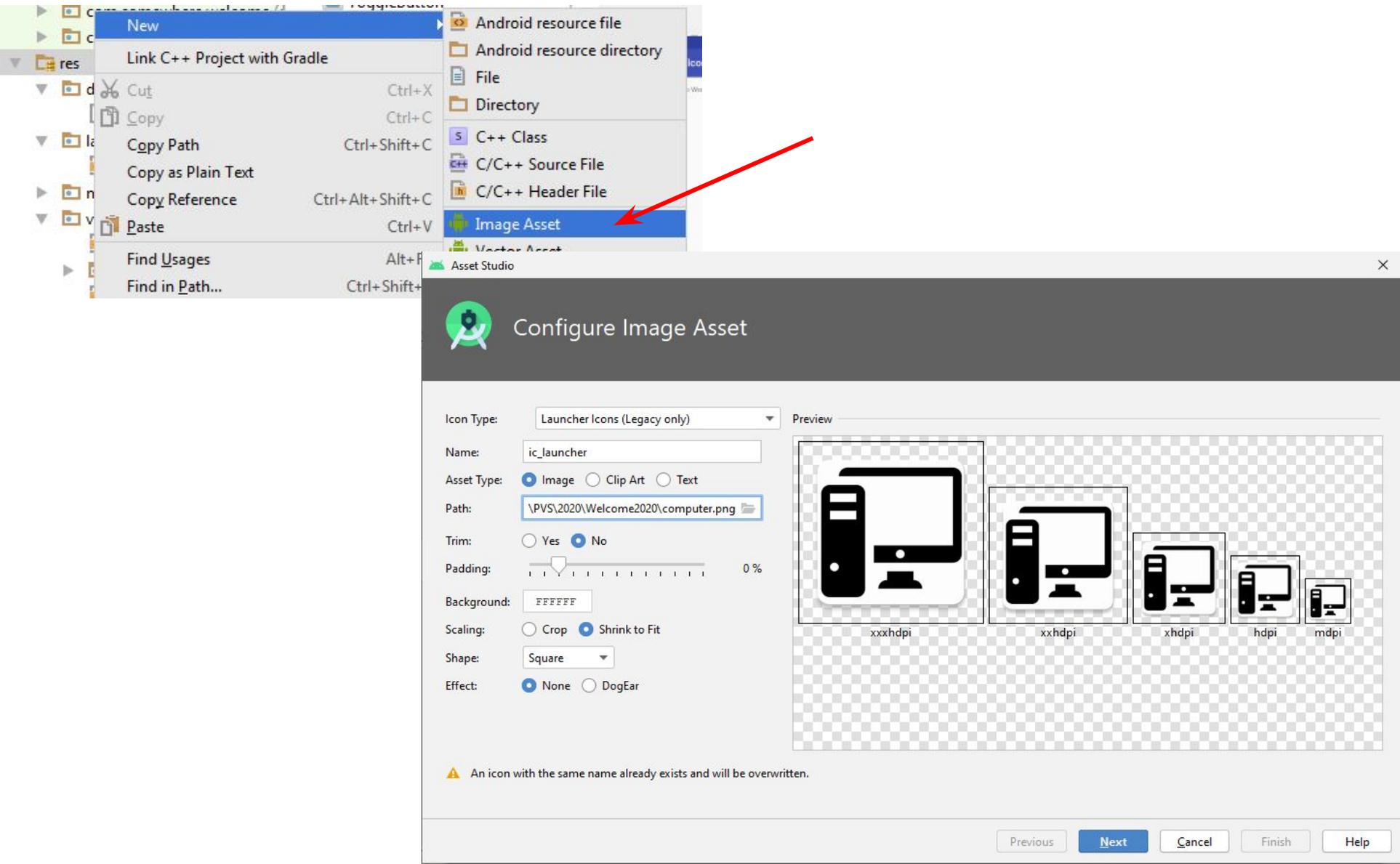
Плотность	Описание
drawable-ldpi	Низкая плотность — приблизительно 120 точек на дюйм
drawable-mdpi	Средняя плотность — приблизительно 160 точек на дюйм
drawable-hdpi	Высокая плотность — приблизительно 240 точек на дюйм
drawable-xhdpi	Очень высокая плотность — приблизительно 320 точек на дюйм
drawable-xxhdpi	Сверхвысокая плотность — приблизительно 480 точек на дюйм
drawable-xxxhdpi	Ультравысокая плотность — приблизительно 640 точек на дюйм

- Android Studio отображает только одну папку `drawable` с графическими ресурсами приложения, даже если проект содержит ресурсы для разных значений плотности.

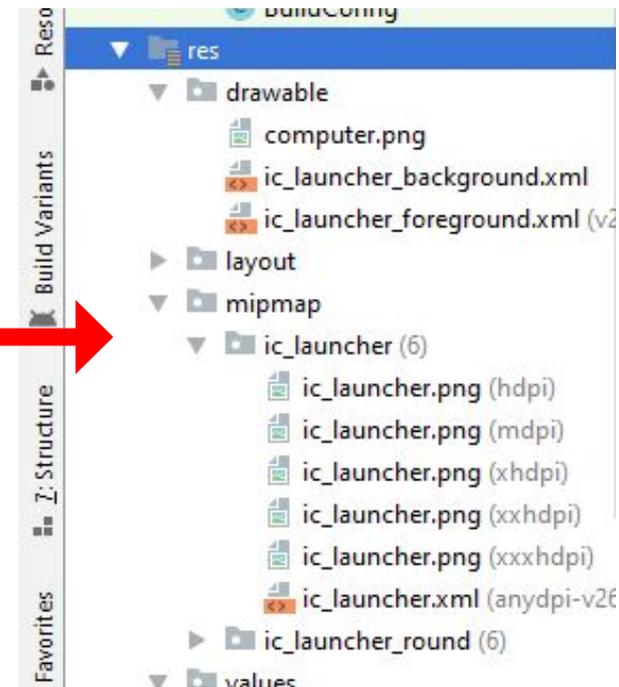
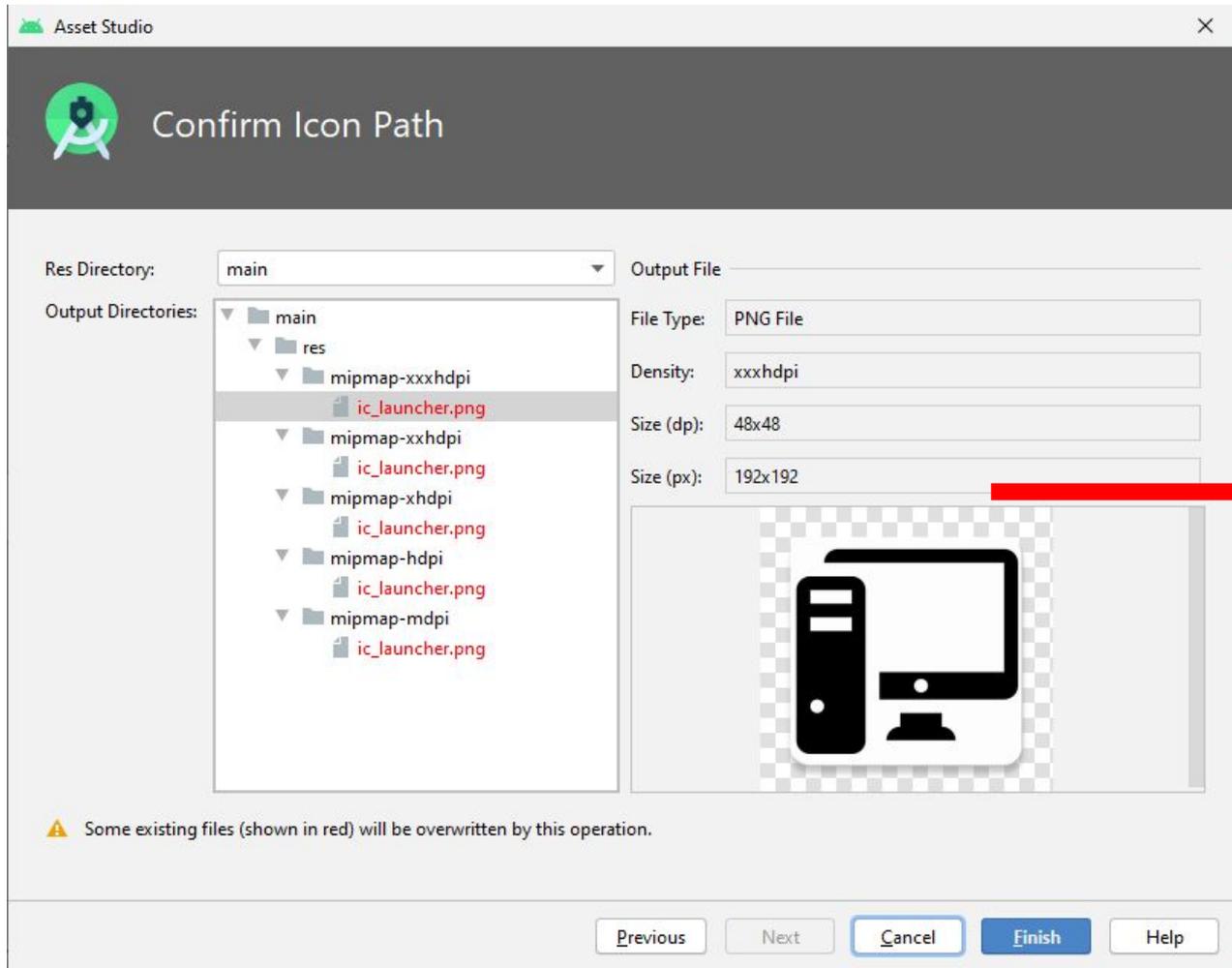
Построение графического интерфейса. Добавление изображения в проект



Построение графического интерфейса. Добавление значка приложения



Построение графического интерфейса. Добавление значка приложения



Построение графического интерфейса. Замена ConstraintLayout на LinearLayout

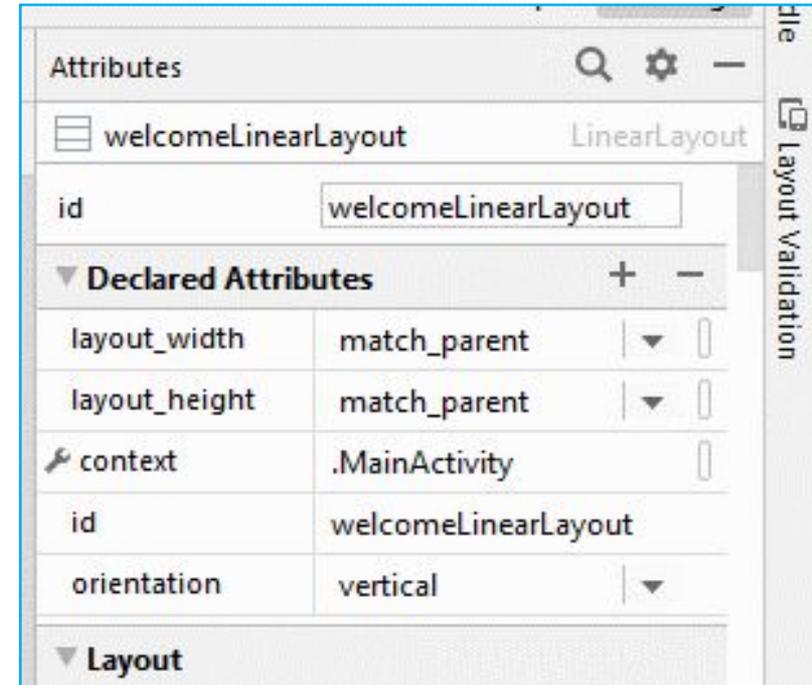
```
activity_main.xml x MainActivity.java x
1 <?xml version="1.0" encoding="utf-8"?>
2 <androidx.constraintlayout.widget.ConstraintLayout xmlns:
3   xmlns:app="http://schemas.android.com/apk/res-auto"
4   xmlns:tools="http://schemas.android.com/tools"
5   android:layout_width="match_parent"
6   android:layout_height="match_parent"
7   tools:context=".MainActivity">
8
9   <TextView
10     android:layout_width="wrap_content"
11     android:layout_height="wrap_content"
12     android:text="Hello World!"
13     app:layout_constraintBottom_toBottomOf="parent"
14     app:layout_constraintLeft_toLeftOf="parent"
15     app:layout_constraintRight_toRightOf="parent"
16     app:layout_constraintTop_toTopOf="parent" />
17
18 </androidx.constraintlayout.widget.ConstraintLayout>
```

```
activity_main.xml x MainActivity.java x
1 <?xml version="1.0" encoding="utf-8"?>
2 <LinearLayout xmlns:android="http://schemas.android.com/apk/res
3   LinearLayout
4   androidx.appcompat.widget.LinearLayoutCompat
5   Press Ctrl+Space to view tags from other namespaces
6   android:layout_height="match_parent"
7   tools:context=".MainActivity">
8
9   <TextView
10     android:layout_width="wrap_content"
11     android:layout_height="wrap_content"
12     android:text="Hello World!"
13     app:layout_constraintBottom_toBottomOf="parent"
14     app:layout_constraintLeft_toLeftOf="parent"
15     app:layout_constraintRight_toRightOf="parent"
16     app:layout_constraintTop_toTopOf="parent" />
17
18 </LinearLayout>
```

Построение графического интерфейса. Настройка свойств

Лучше настраивать в режиме “Design”

- LinearLayout.
id: welcomeLinearLayout
orientation: vertical
- TextView.
id: welcomeTextView
text: ...



Построение графического интерфейса. Настройка свойств

- welcomeTextView.text

The image illustrates the steps to configure the text property of a `welcomeTextView` widget in an Android application. It shows three overlapping windows from the Android Studio IDE:

- Left Window (Attributes Panel):** Shows the 'Common Attributes' for a `TextView` widget. The 'text' attribute is currently set to 'Hello World!'. A red arrow points from this attribute to the 'Pick a Resource' dialog.
- Top Window (Pick a Resource):** A dialog titled 'Pick a Resource' for the 'Module: app'. It lists 'String Resource File' and 'String Value'. The 'String Value' option is selected and circled in red. A red arrow points from this selection to the 'New String Value' dialog.
- Bottom Window (New String Value):** A dialog titled 'New String Value' with the following fields:
 - Resource name: `welcome`
 - Resource value: `Welcome to Android App!`
 - Source set: `main`
 - File name: `strings.xml`
 - Under 'Create the resource in directories:', the `values` directory is checked.A red arrow points from the 'Resource value' field to the 'text' attribute in the widget's 'Common Attributes' panel at the bottom.

At the bottom, the 'Common Attributes' panel for the `welcomeTextView` widget shows the 'text' attribute now set to `@string/welcome`, indicating that the resource has been successfully linked.

Построение графического интерфейса. Настройка свойств

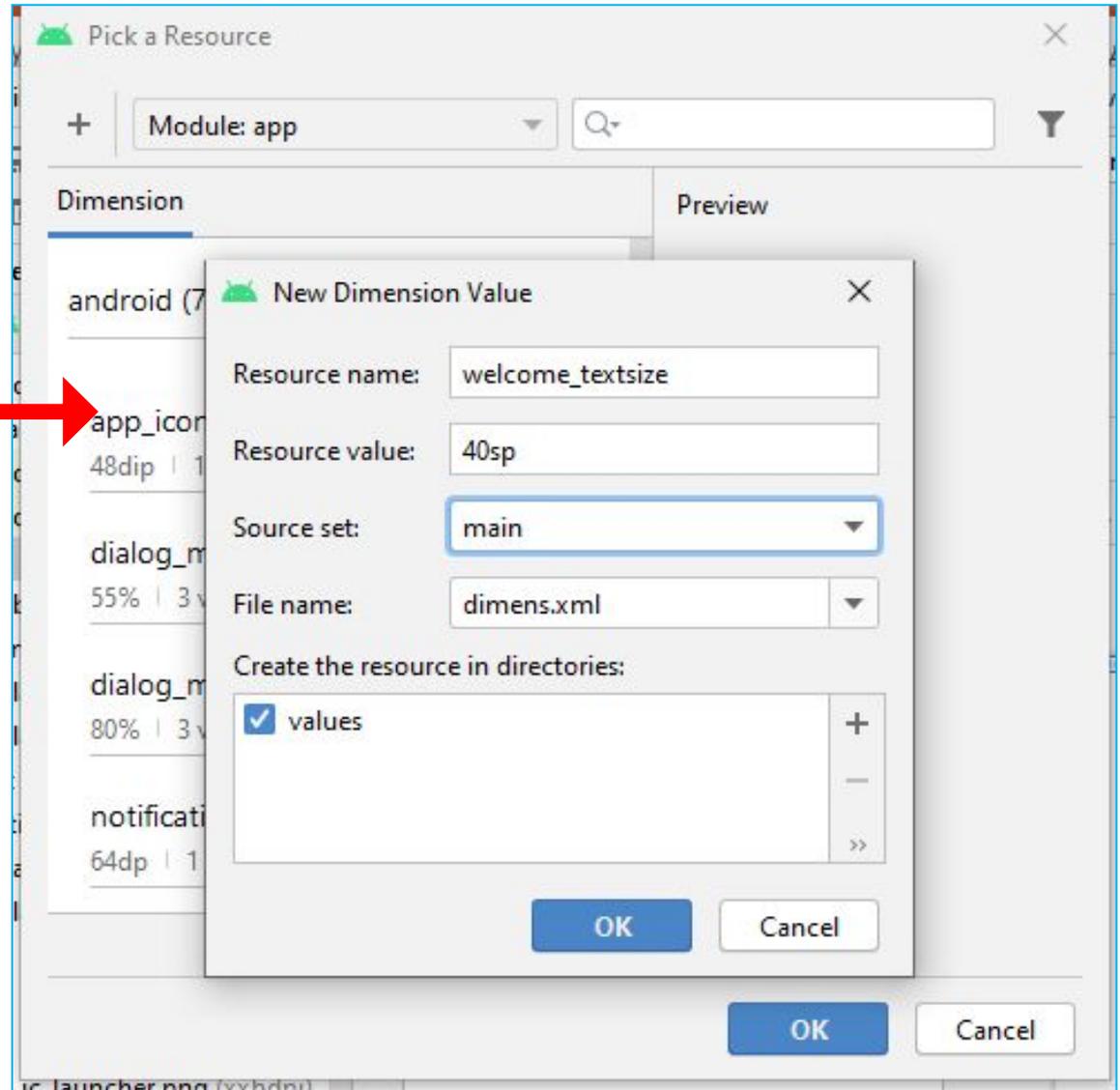
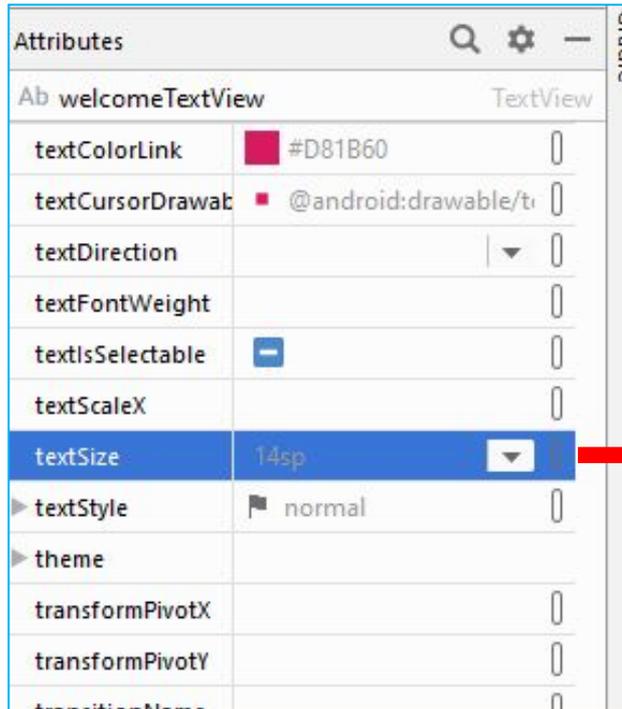
- `welcomeTextView.textSize`

компоненты GUI и другие экранные
элементы

Единица измерения	Описание
px	Пиксел
dp или dip	Пиксел, независимый от плотности
sp	Пиксел, независимый от масштабирования
in	Дюймы
mm	Миллиметры

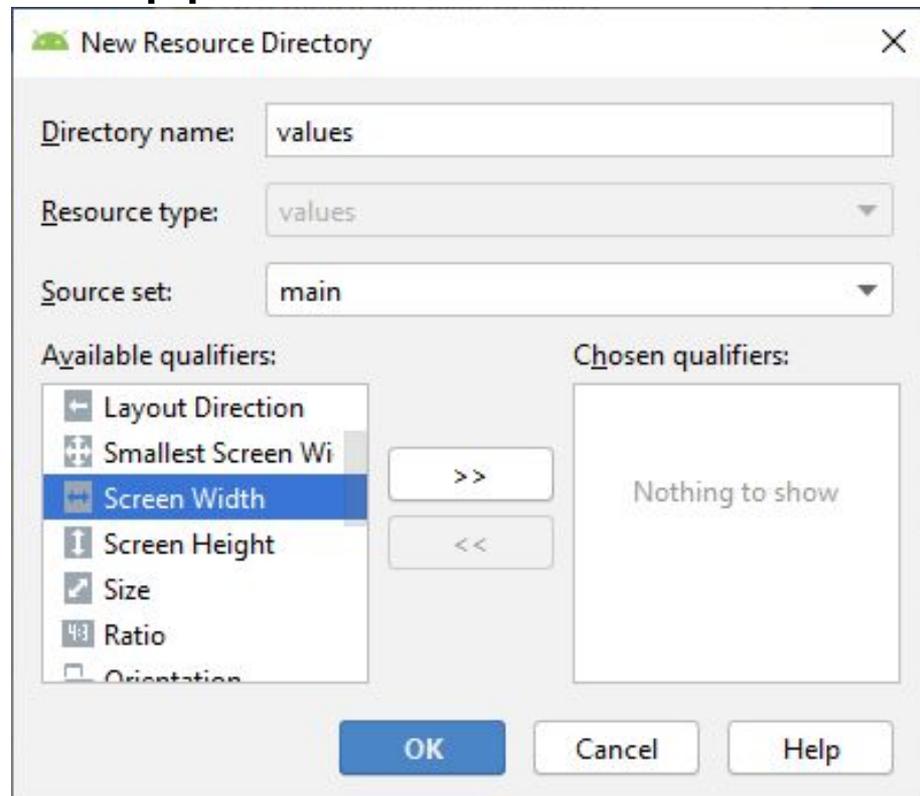
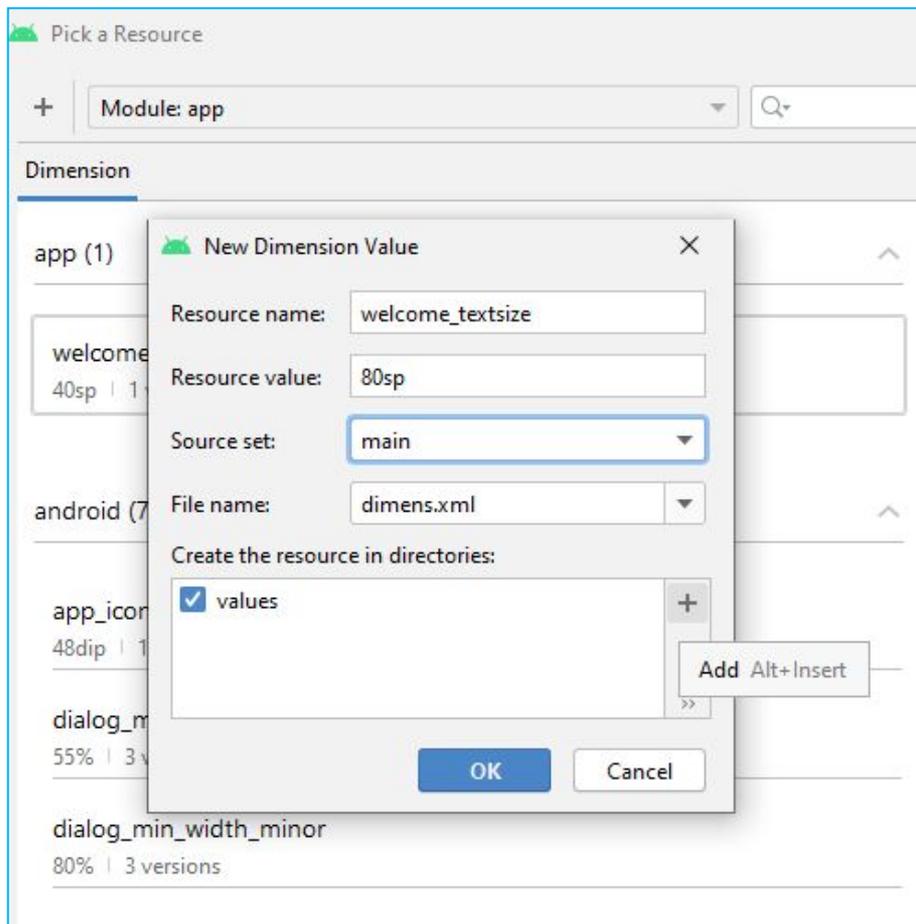
шрифты

Построение графического интерфейса. Настройка свойств



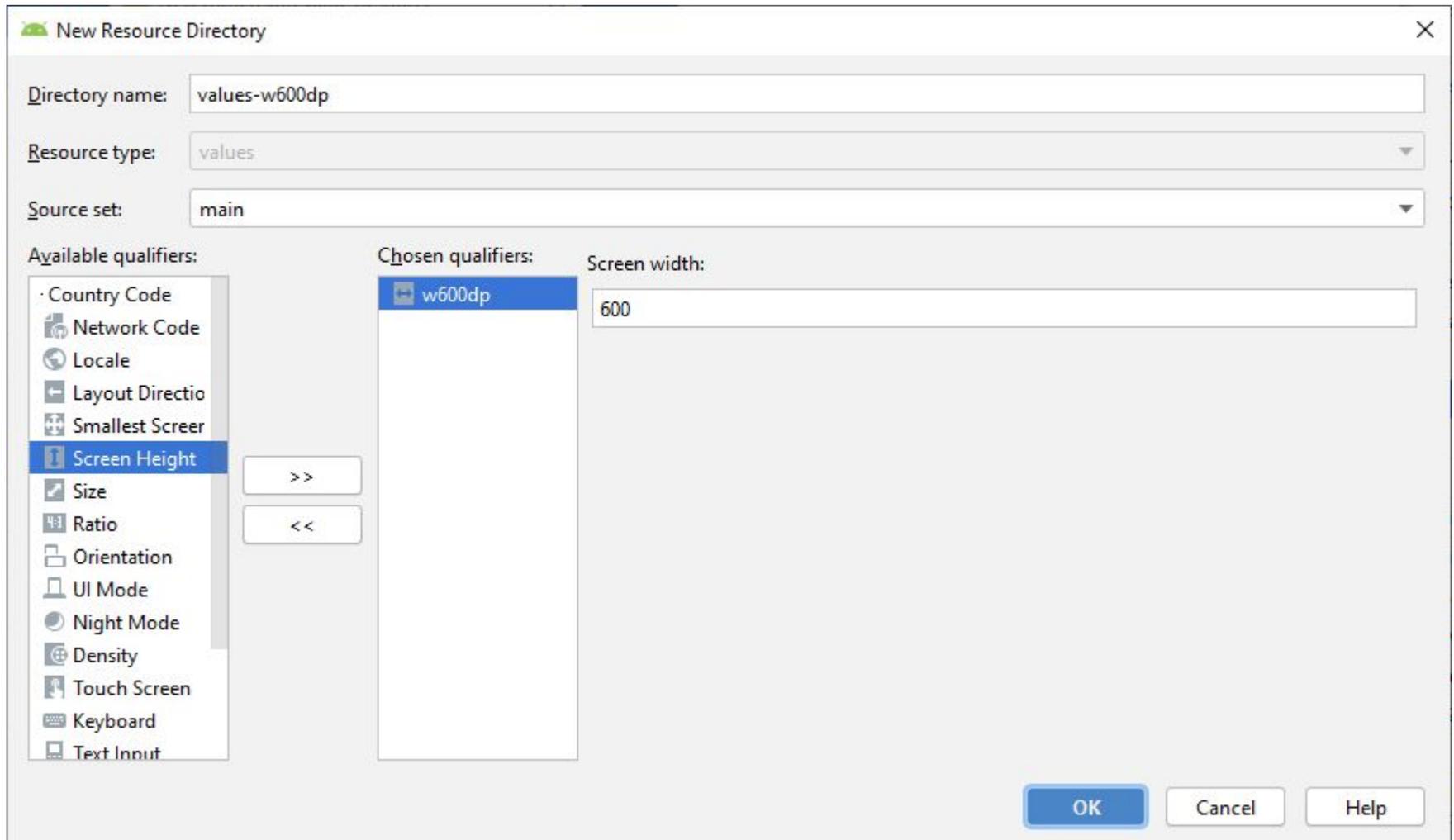
Построение графического интерфейса. Настройка свойств

- `welcomeTextView.textSize` для планшета



Построение графического интерфейса. Настройка свойств

- `welcomeTextView.textSize` для планшета



Построение графического интерфейса. Настройка свойств

- `welcomeTextView.textSize` для планшета

The image shows two dialog boxes from the Android Studio IDE. The larger dialog, titled "New Resource Directory", is used to create a new resource directory. It has the following fields and options:

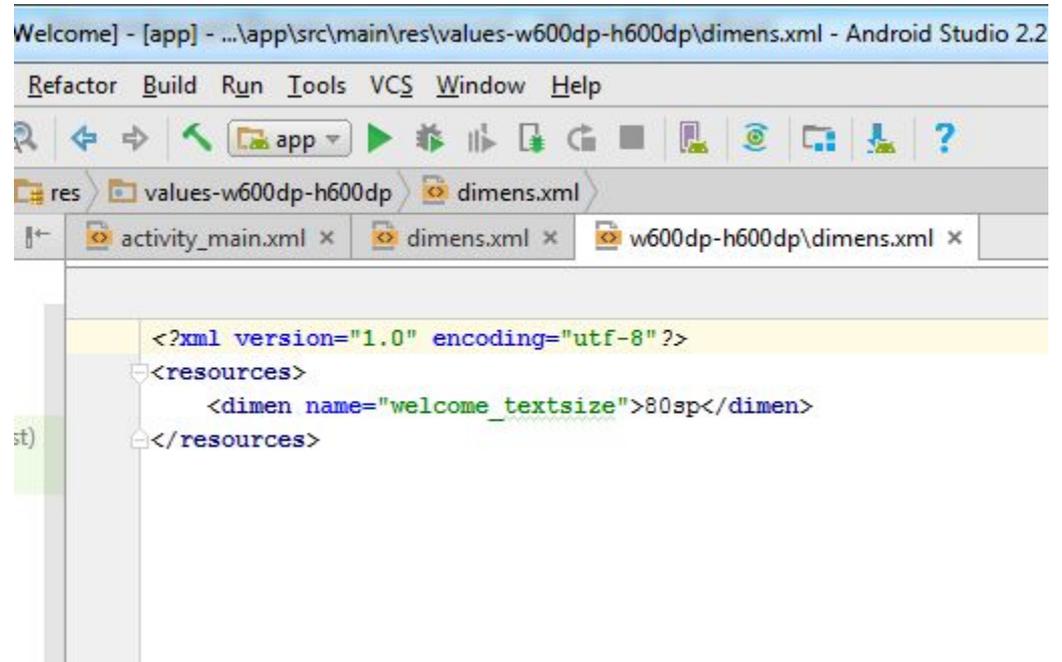
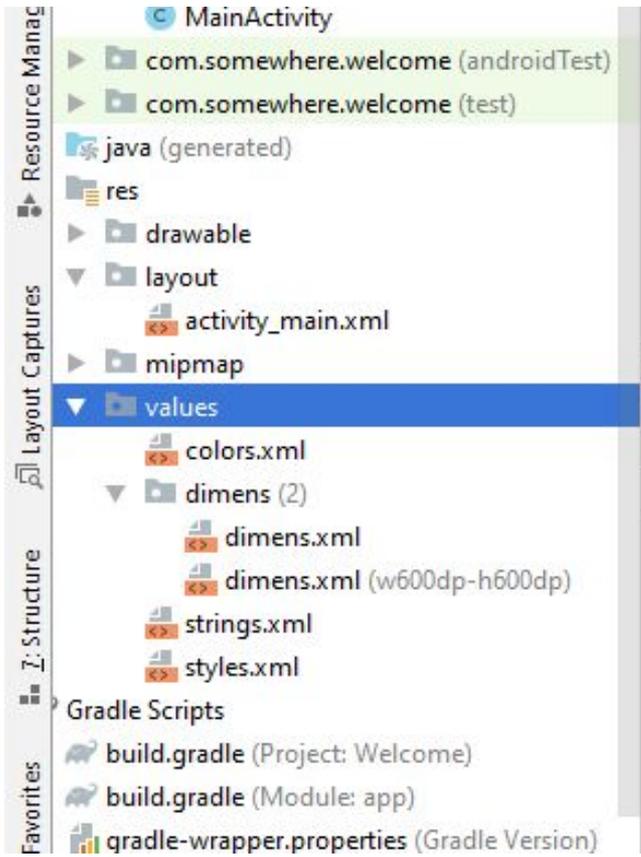
- Directory name: `values-w600dp-h600dp`
- Resource type: `values`
- Source set: `main`
- Available qualifiers: A list of qualifiers including Country Code, Network Code, Locale, Layout Directio, Smallest Screer, Size (selected), Ratio, Orientation, UI Mode, Night Mode, Density, Touch Screen, Keyboard, Text Input, and Navigation Stat.
- Chosen qualifiers: A list containing `w600dp` and `h600dp`.
- Screen height: `600`
- Buttons: `>>`, `<<`, and `OK`.

The smaller dialog, titled "New Dimension Value Resource", is used to create a new dimension value resource. It has the following fields and options:

- Resource name: `welcome_textsize`
- Resource value: `80sp`
- Source set: `main`
- File name: `dimens.xml`
- Create the resource in directories: A list of directories with checkboxes. `values` is unchecked, and `values-w600dp-h600dp` is checked.
- Buttons: `OK` and `Cancel`.

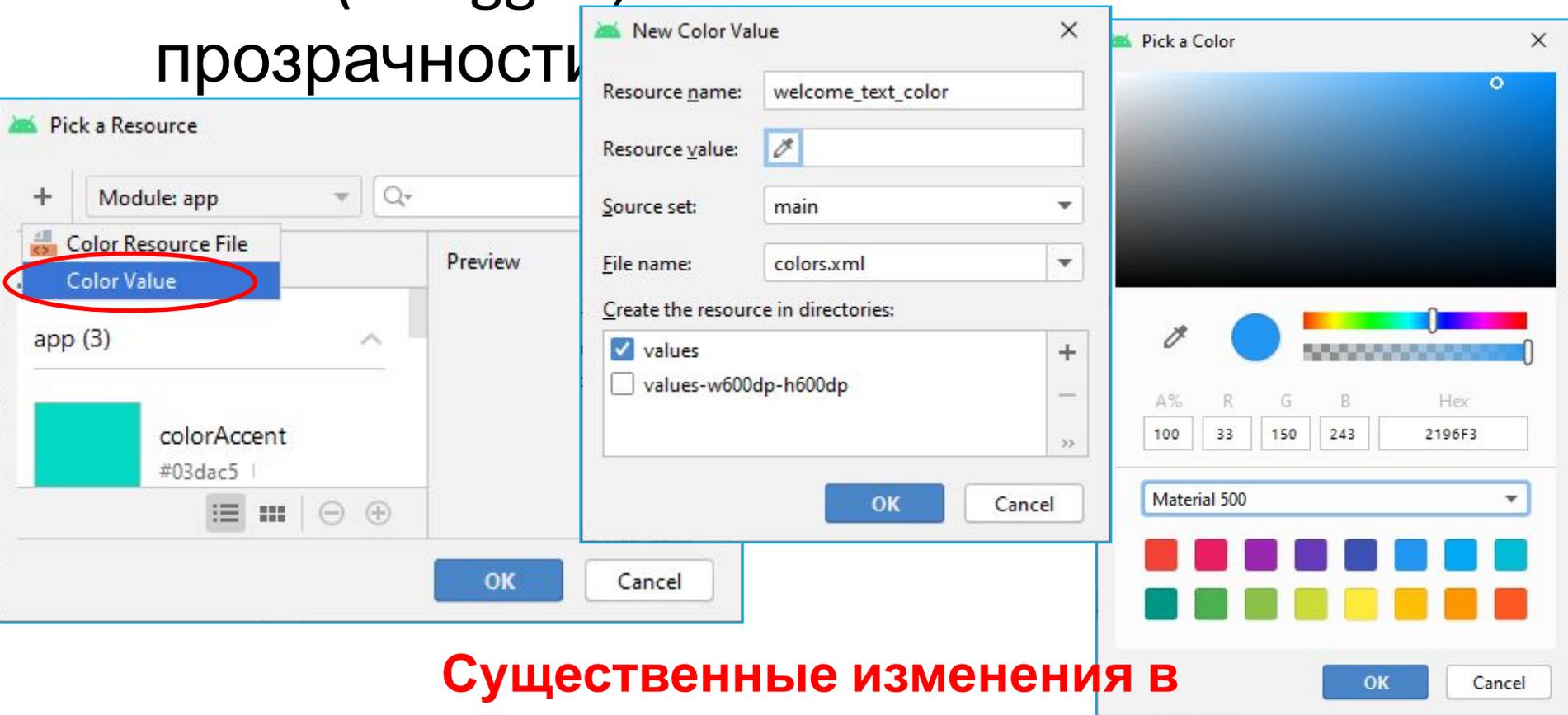
Построение графического интерфейса. Настройка свойств

- `welcomeTextView.textSize` для планшета



Построение графического интерфейса. Настройка свойств

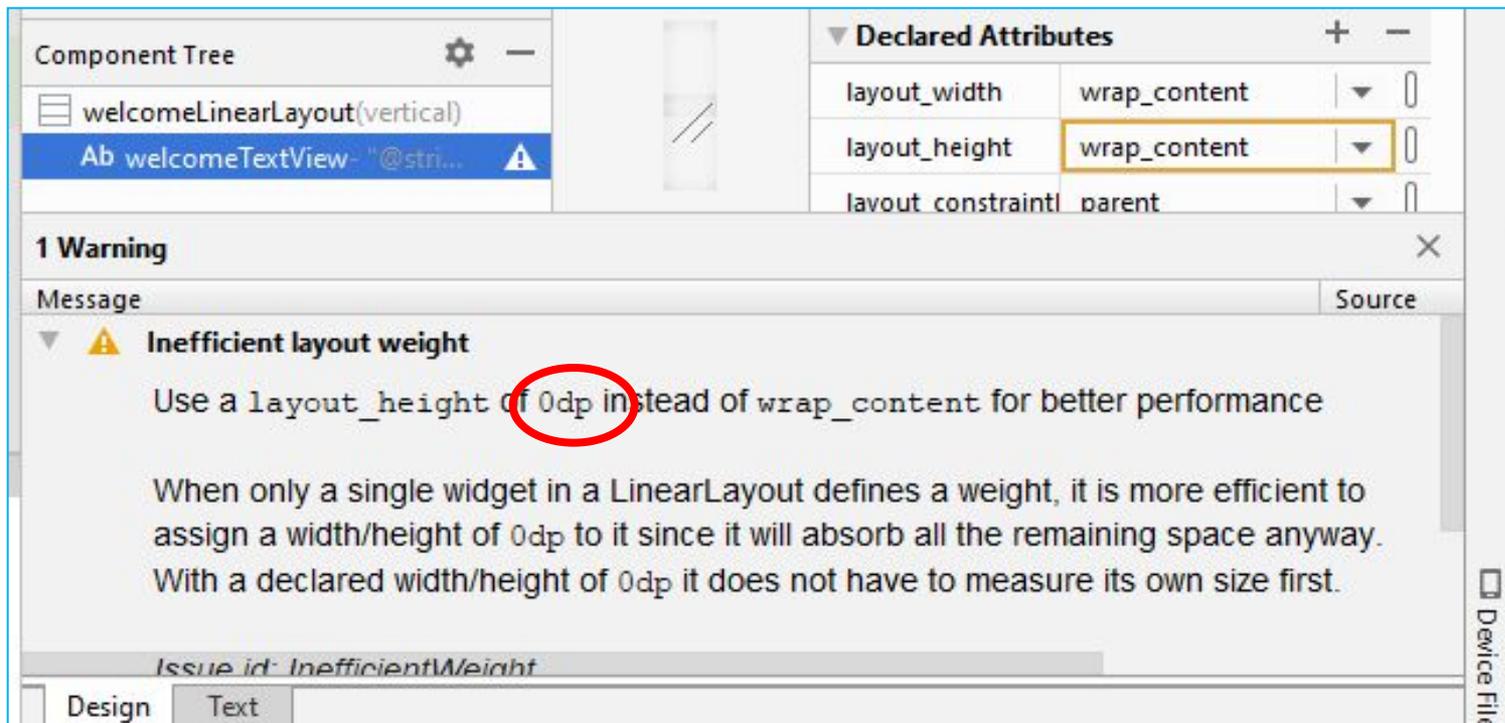
- `welcomeTextView.textColor`
RGB (rrggbb, без прозрачности)
ARGB (aarrggbb, aa – степень
прозрачности)



Существенные изменения в

Построение графического интерфейса. Настройка свойств

- `welcomeTextView.gravity: center`
- `welcomeTextView.layout_gravity: center_horizontal`
- `welcomeTextView.layout_weight: 1`



The screenshot shows the Android Studio interface. The Component Tree on the left shows a `welcomeTextView` selected. The Declared Attributes panel on the right shows `layout_height` set to `wrap_content`. A warning message is displayed in the center, stating: "1 Warning: Inefficient layout weight. Use a `layout_height` of `0dp` instead of `wrap_content` for better performance." The text "`0dp`" is circled in red. The warning message also includes a detailed explanation: "When only a single widget in a LinearLayout defines a weight, it is more efficient to assign a width/height of `0dp` to it since it will absorb all the remaining space anyway. With a declared width/height of `0dp` it does not have to measure its own size first." The issue ID is `InefficientWeight`. The Design and Text tabs are visible at the bottom.

Declared Attributes	
<code>layout_width</code>	<code>wrap_content</code>
<code>layout_height</code>	<code>wrap_content</code>
<code>layout_constraint</code>	<code>parent</code>

1 Warning

Inefficient layout weight

Use a `layout_height` of `0dp` instead of `wrap_content` for better performance

When only a single widget in a `LinearLayout` defines a weight, it is more efficient to assign a width/height of `0dp` to it since it will absorb all the remaining space anyway. With a declared width/height of `0dp` it does not have to measure its own size first.

Issue id: `InefficientWeight`

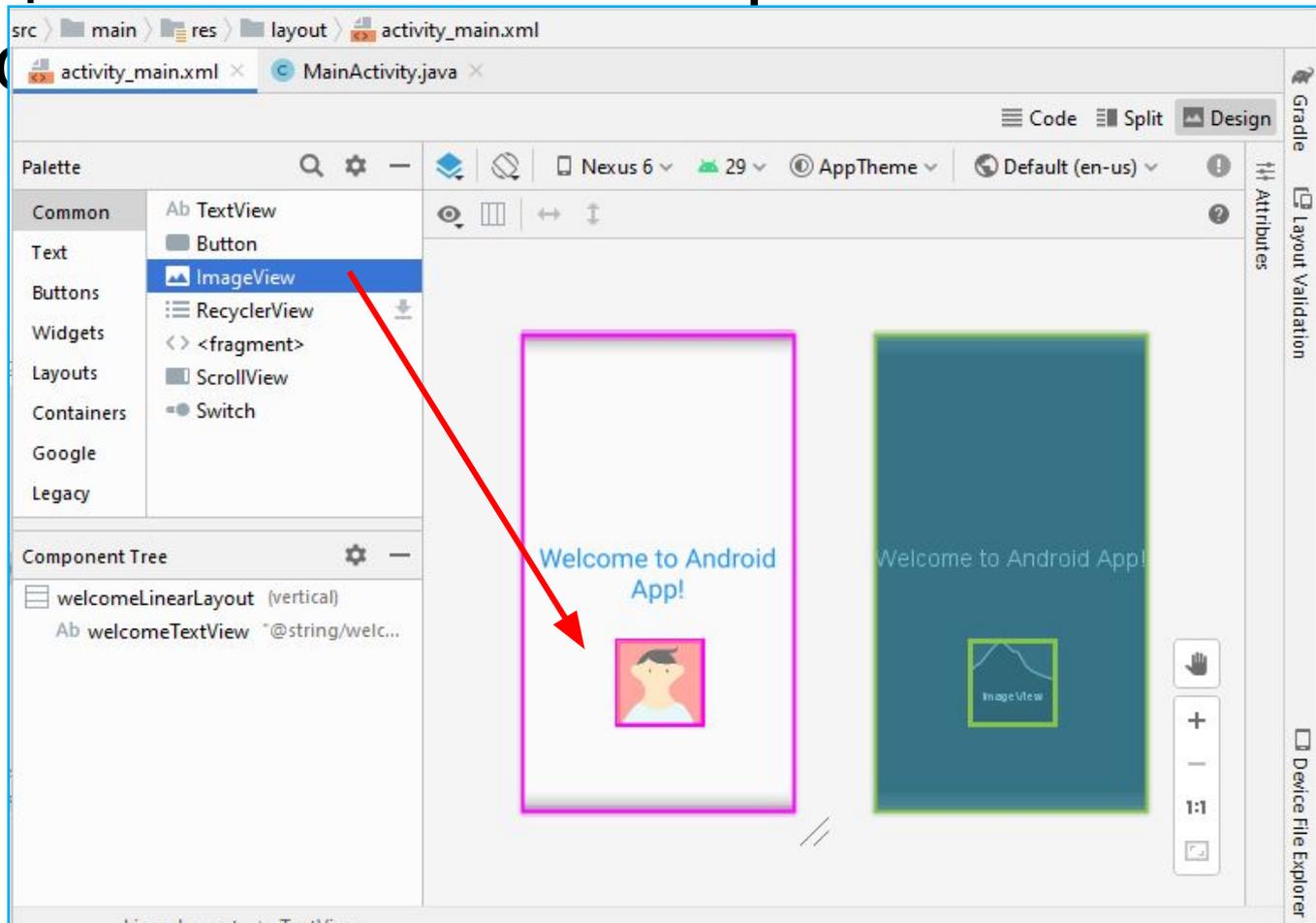
Построение графического интерфейса.

Добавление компонента ImageView

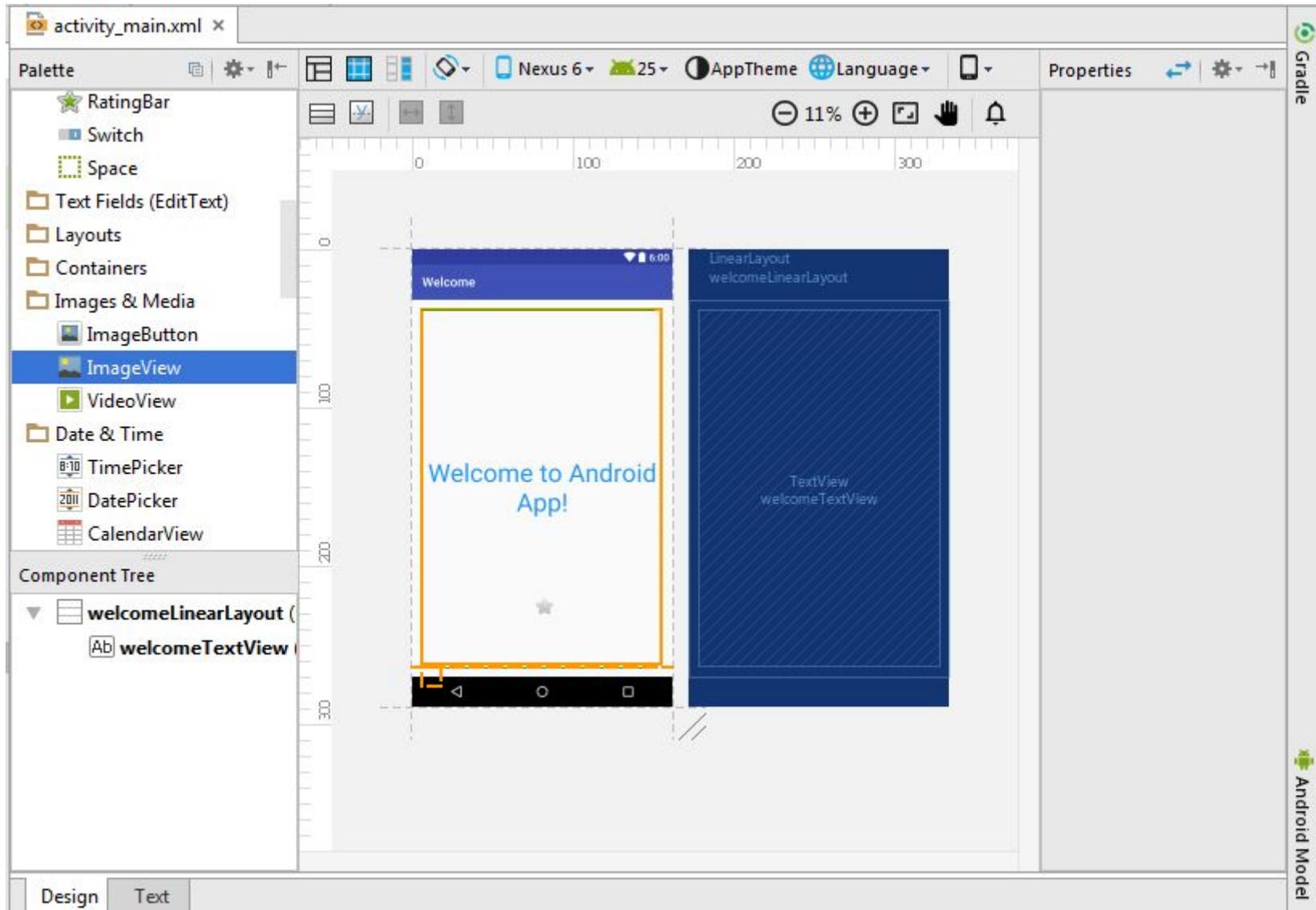
- Перетащить компонент ImageView из раздела Common палитры в область холста
- Оранжевые линейки представляют границы каждого существующего представления в макете
- Зеленые линейки указывают позицию нового представления относительно существующих представлений. По умолчанию новые представления добавляются к нижнему краю вертикального компонента LinearLayout, если не навести указатель мыши на оранжевый прямоугольник, ограничивающий верхнее представление макета
- Подсказка (tooltip) сообщает, как будет настроен компонент, если отпустить его в текущей позиции мыши

Построение графического интерфейса. Добавление компонента ImageView

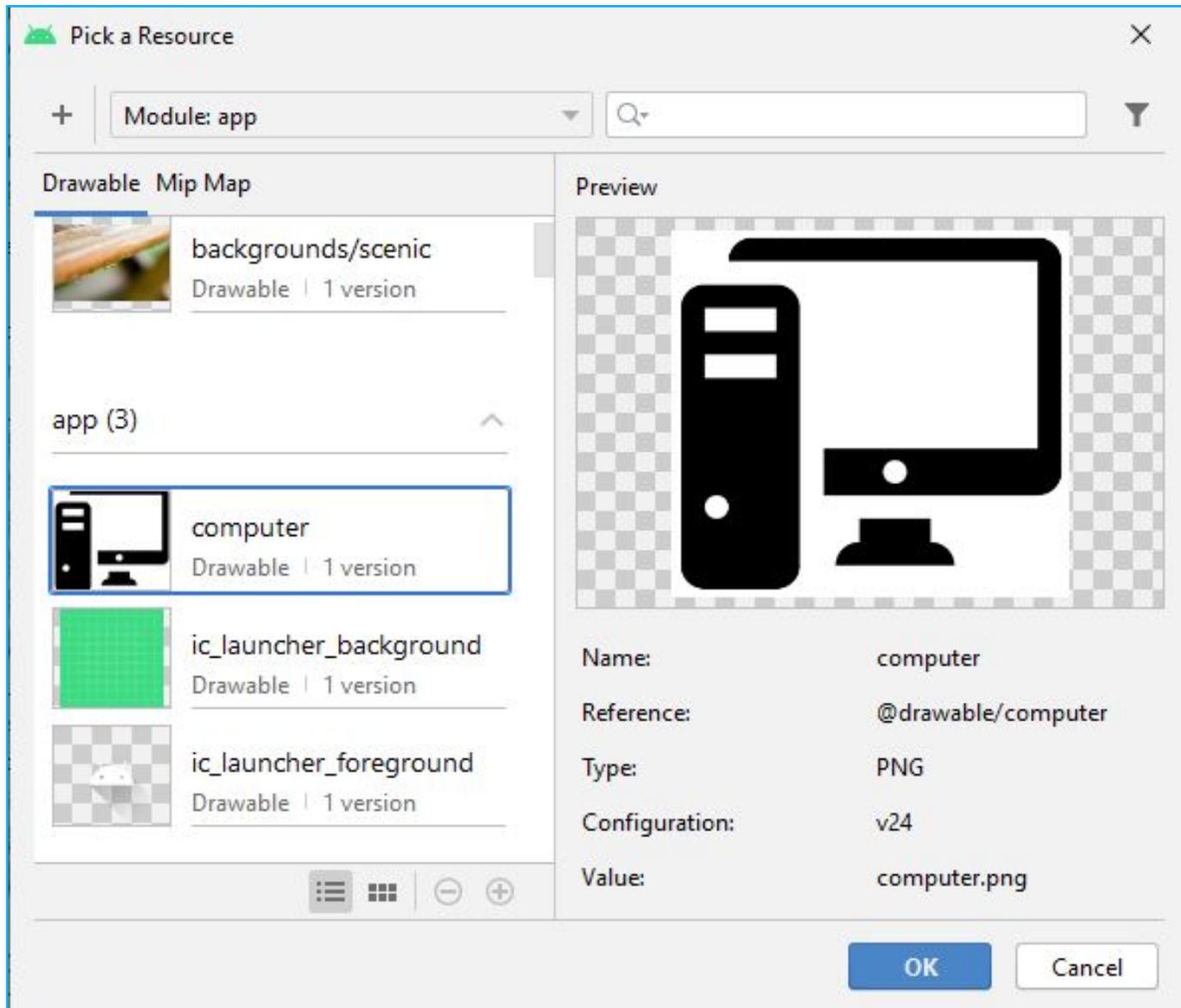
- Перетащить компонент ImageView из раздела Common палитры в область холста



Построение графического интерфейса. Добавление компонента ImageView



Построение графического интерфейса. Добавление компонента ImageView



Построение графического интерфейса. Добавление компонента ImageView

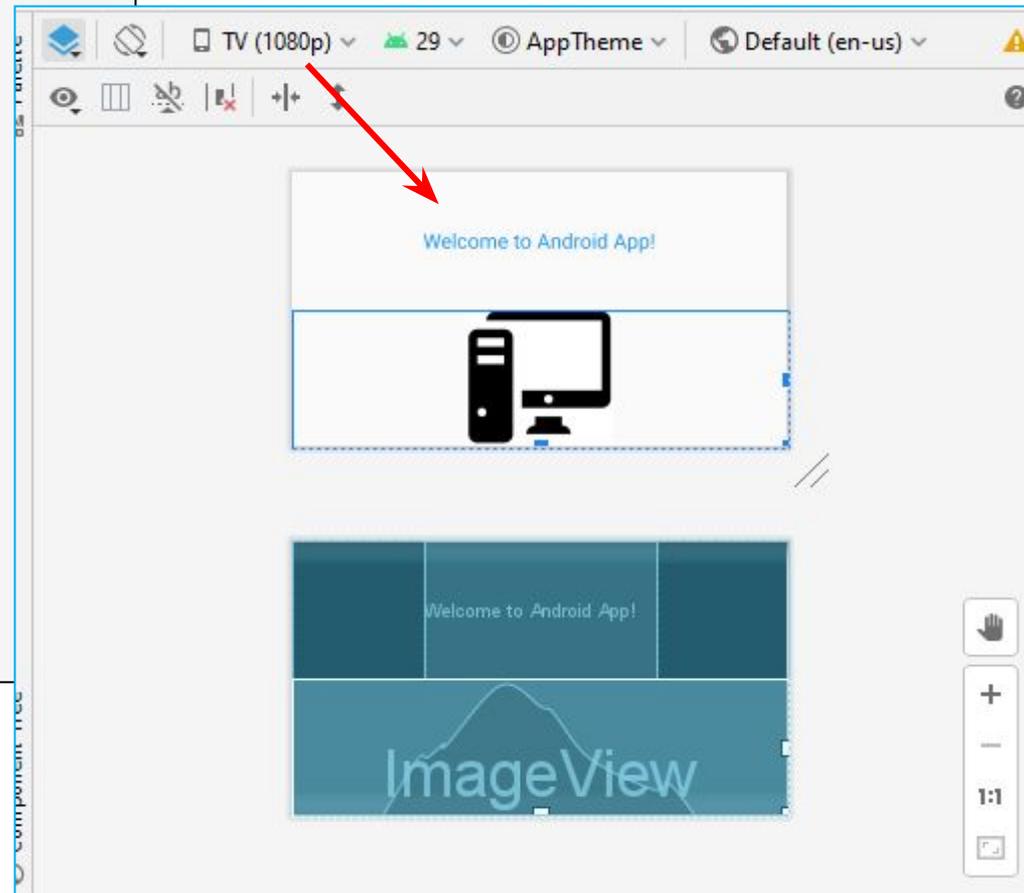
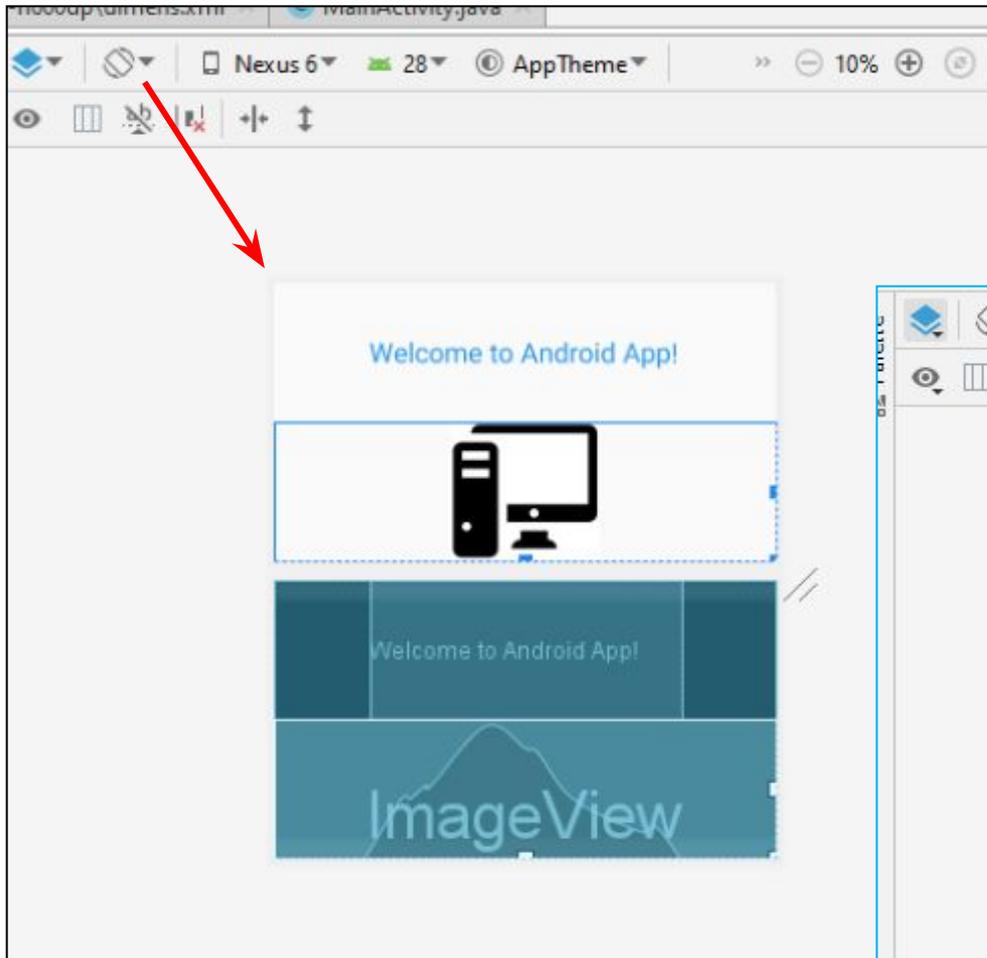
The screenshot displays the Android Studio interface during the design phase of an app. The top bar shows the current files: `activity_main.xml` and `MainActivity.java`. The main workspace is in 'Design' mode, showing a preview of the app's layout on a Nexus 6 device. The text 'Welcome to Android App!' is visible. A red dashed box highlights a new `ImageView` component being added to the layout, which contains an icon of a computer monitor and tower. The 'Palette' on the left lists various UI components, with `ImageView` selected. The 'Attributes' panel on the right shows the properties for the selected `complmageView` widget. The `id` attribute is set to `complmageView` and is circled in red. Below the 'Declared Attributes' section, the 'Layout' section shows `layout_width` set to `match_paren`, `layout_height` set to `0dp`, and `layout_weight` set to `1`. The 'Common Attributes' section shows `srcCompat` set to `drawable/compu`. The 'Component Tree' at the bottom left shows the hierarchy: `welcomeLinearLayout` (vertical) containing `welcomeTextView` and `complmageView`.

Declared Attributes	
<code>layout_width</code>	<code>match_paren</code>
<code>layout_hei...</code>	<code>0dp</code>
<code>layout_wei...</code>	<code>1</code>
<code>id</code>	<code>complmageView</code>
<code>srcCompat</code>	<code>drawable/compu</code>

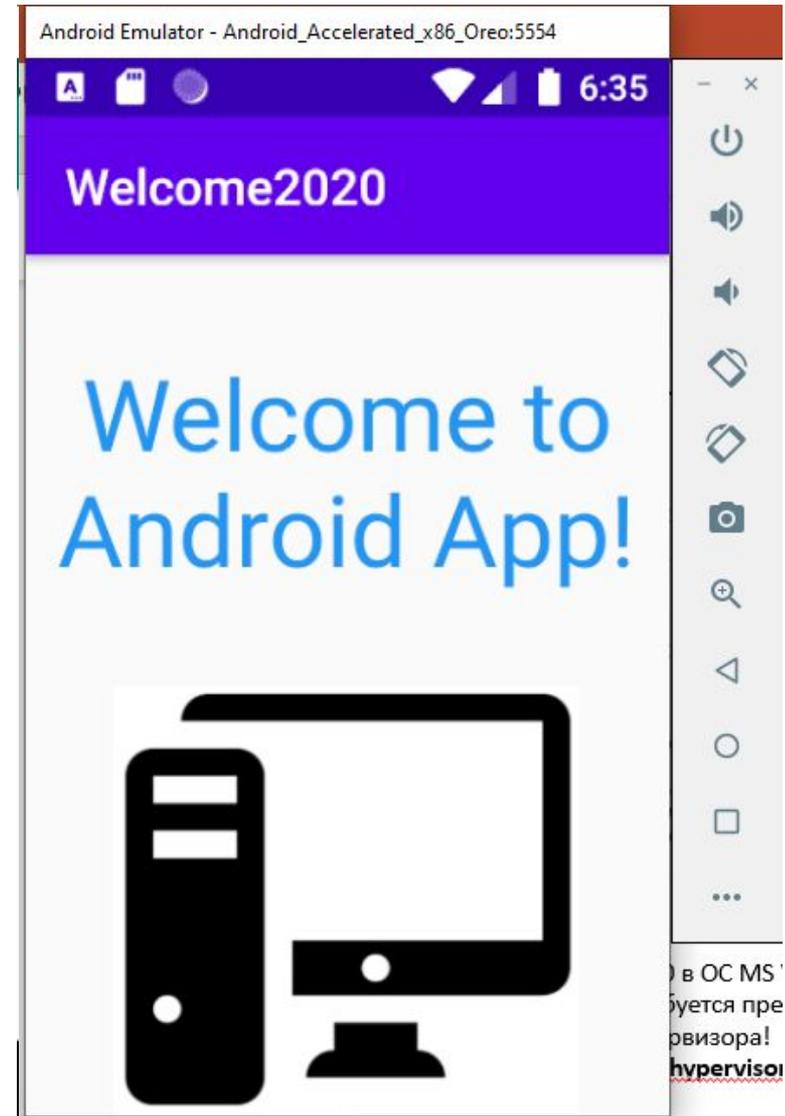
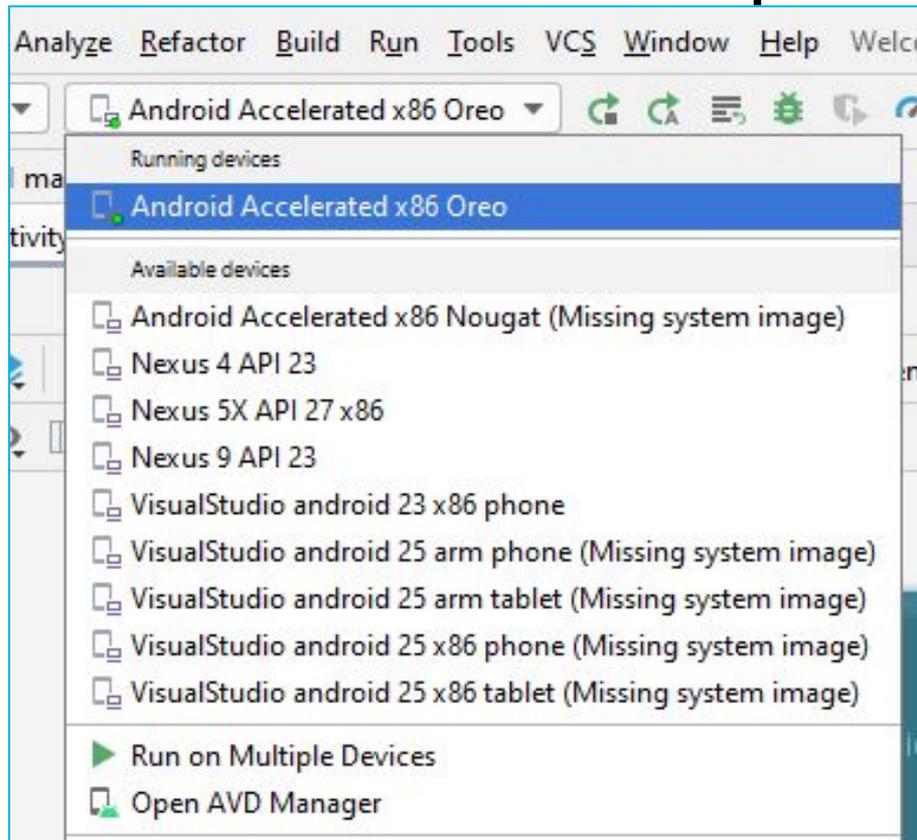
Layout	
<code>layout_width</code>	<code>match_paren</code>
<code>layout_height</code>	<code>0dp</code>
<code>layout_weight</code>	<code>1</code>
<code>visibility</code>	
<code>visibility</code>	

Common Attributes	
<code>srcCompat</code>	<code>drawable/compu</code>

Предварительный просмотр и выполнение приложения



Предварительный просмотр и выполнение приложения



У версий AS версий до 4.0 в ОС MS Windows 8.1, 10 для работы НАХМ требуется предварительное отключение службы гипервизора!

- **system32>bcdedit /set hypervisorlaunchtype off**
- перезагрузка ОС

в ОС MS'
буется пре
визора!
hypervisor

Интернационализация

- Для каждого локального контекста создаётся свой набор ресурсов
- Когда пользователь запускает приложение, Android автоматически находит и загружает ресурсы, соответствующие настройкам локального контекста устройства
- Проектирование приложений с возможностью такой настройки называется интернационализацией.
- Адаптация ресурсов приложения для каждого локального контекста называется локализацией

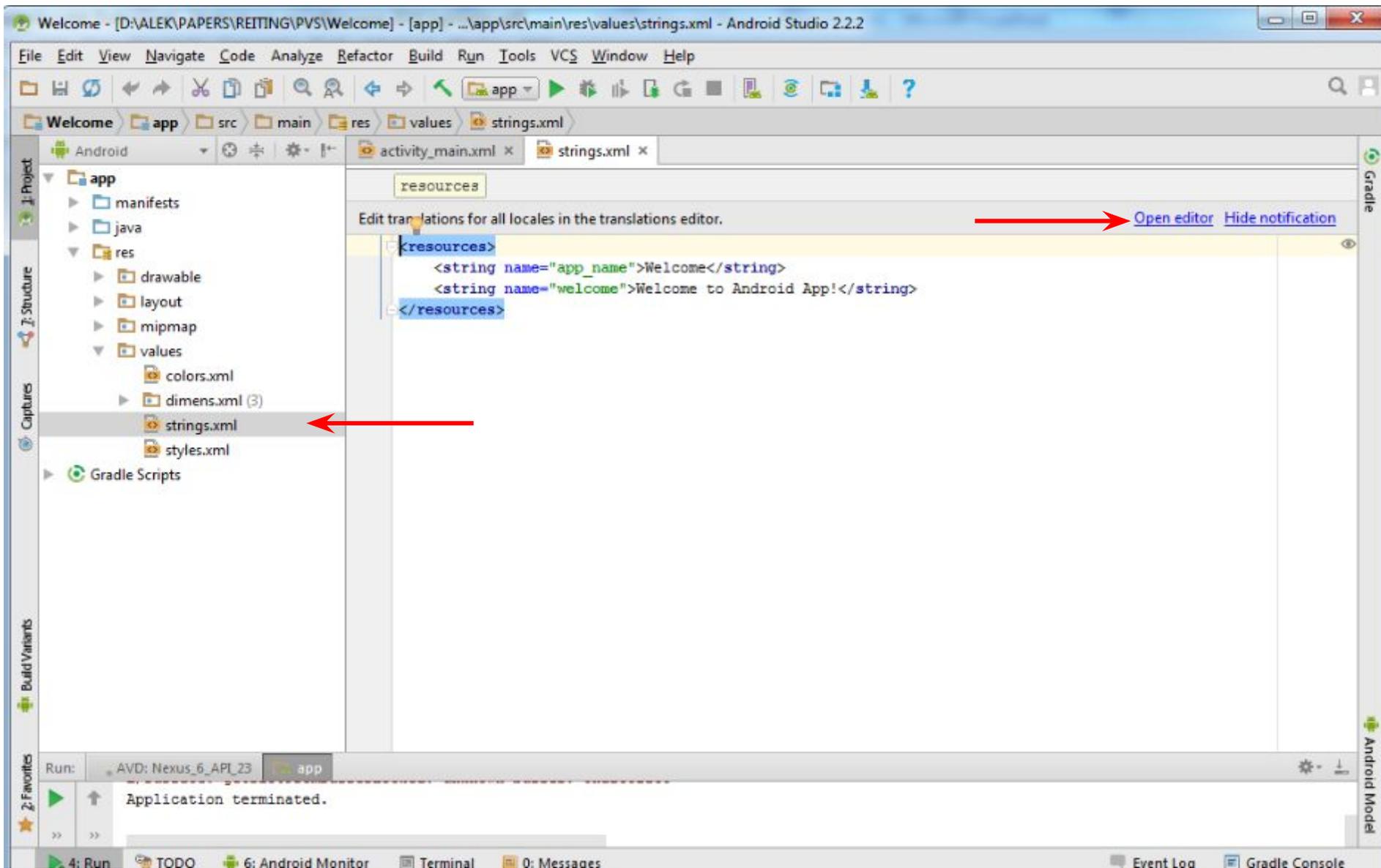
Интернационализация

- Создаются дополнительные файлы ресурсов в формате XML для разных языков
- Во всех файлах используются одни и те же имена ресурсов строк, но с разными переводами
- Android выбирает ресурсный файл в зависимости от основного языка, выбранного на устройстве пользователя

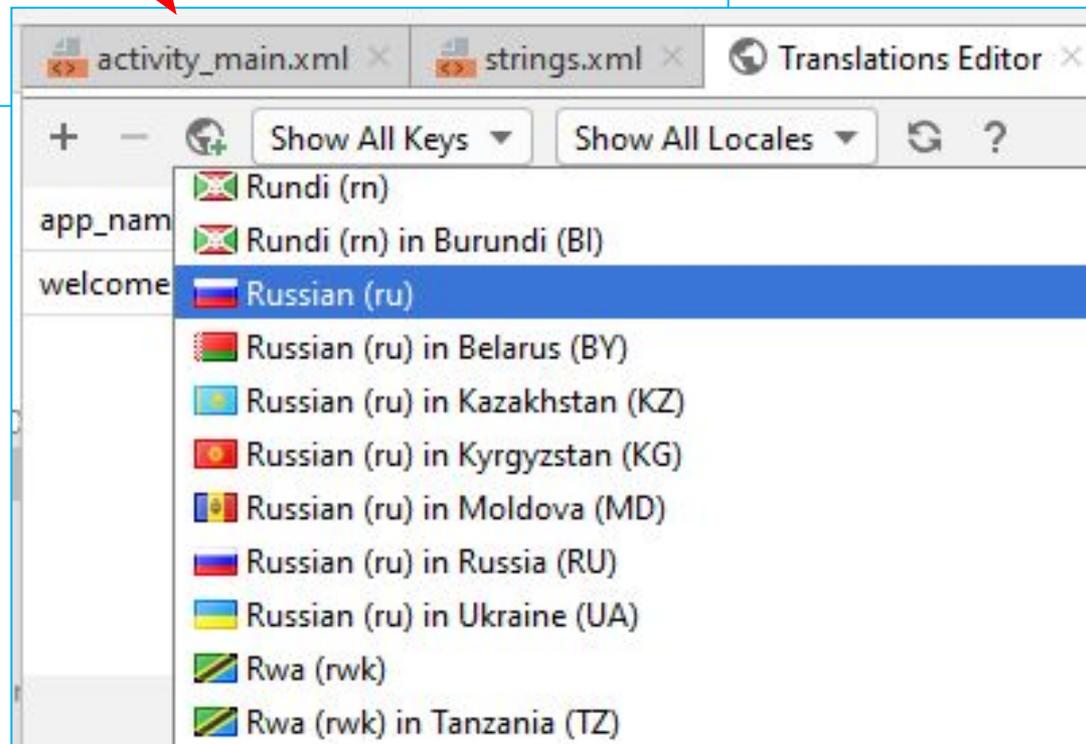
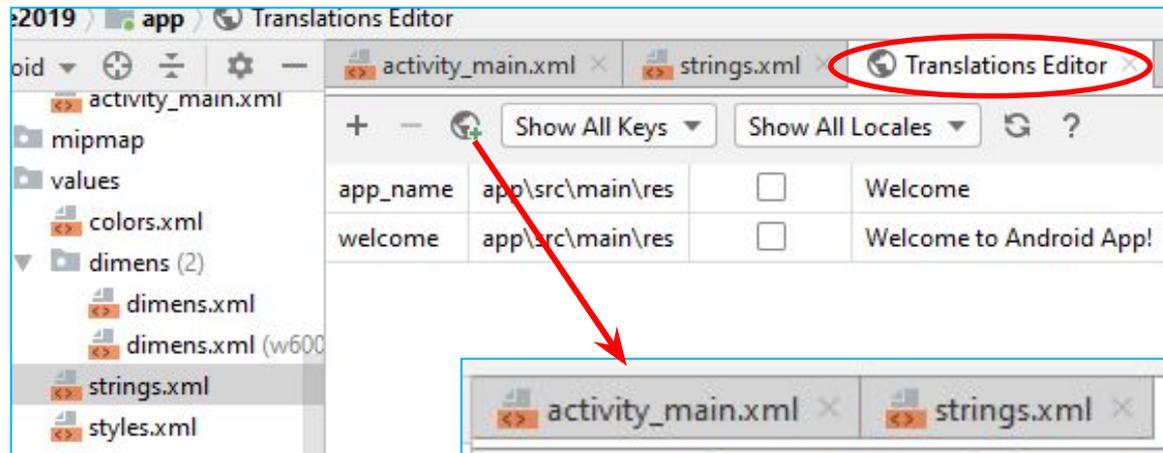
Интернационализация

- XML-файлы ресурсов, содержащие локализованные строки, размещаются во вложенных папках папки res проекта
- Android использует специальные правила назначения имен папок (например, values-fr содержит файл strings.xml для французского языка, а папка values-es содержит файл strings.xml для испанского языка)
- В именах папок также может присутствовать региональная информация (values-en-rUS , values-en-rGB)
- Если локализованные ресурсы для нужного локального контекста отсутствуют, Android использует ресурсы приложения по умолчанию (папка values в папке res)

Интернационализация



Интернационализация



Интернационализация

The screenshot displays the Android Studio interface with the Translations Editor open for the strings.xml file. The table below shows the current state of the translations:

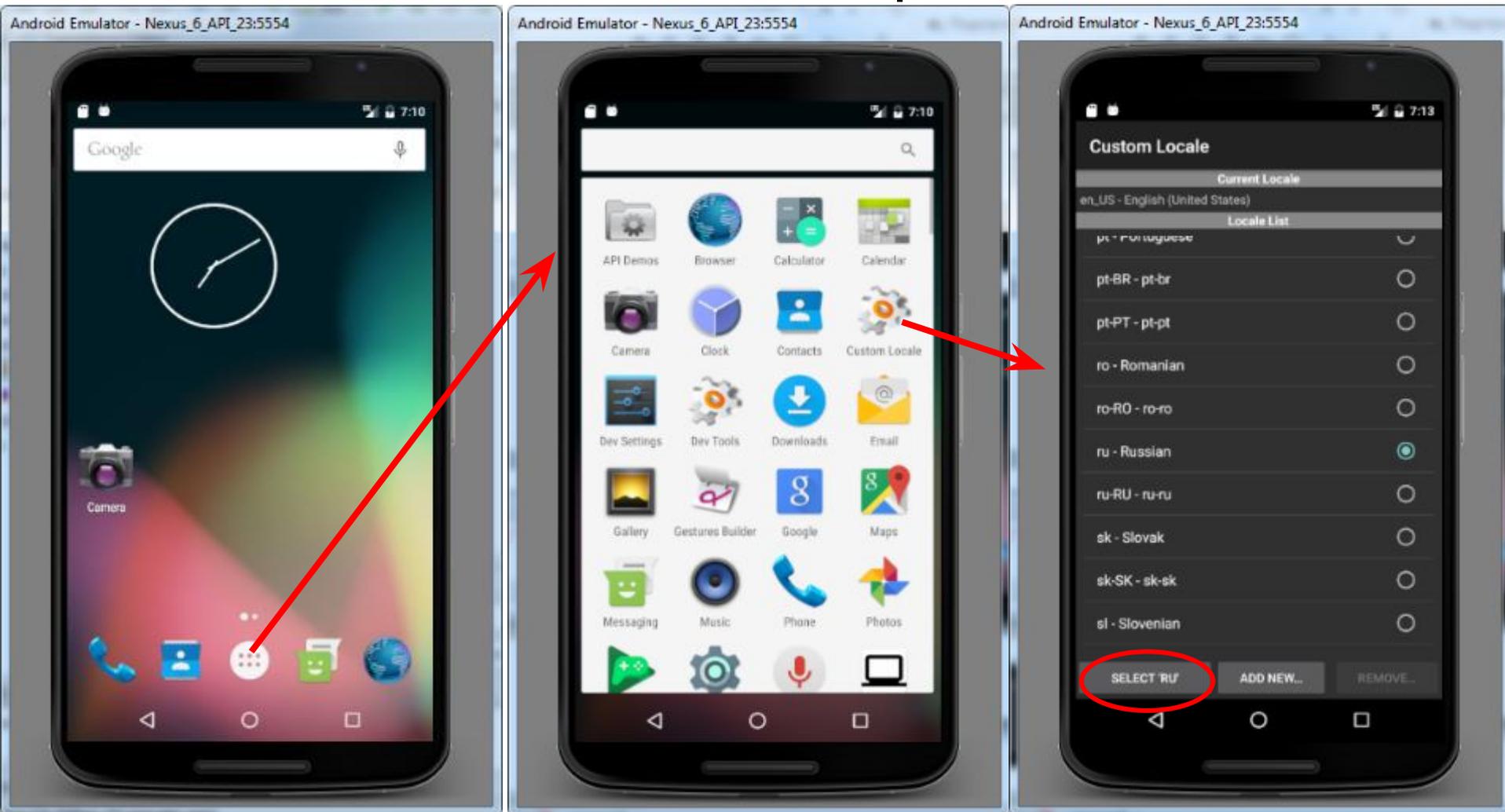
Key	Resource Folder	Untranslatable	Default Value	Russian (ru)
app_name	app\src\main\res	<input checked="" type="checkbox"/>	Welcome	Welcome
welcome	app\src\main\res	<input type="checkbox"/>	Welcome to Android App!	Добро пожаловать в Android-приложении!

Below the table, the details for the selected 'welcome' key are shown:

Key: welcome
Default Value: Welcome to Android App!
Translation: Добро пожаловать в Android-приложении!

The bottom of the screen shows the Build output and Event Log. The Build output indicates a successful build, and the Event Log shows the installation process.

Локализация



Локализация

