

Введение

1. Используйте готовые библиотеки и готовые решения;
2. Используйте проверенные и эффективные паттерны программирования;
3. Используйте контроль версий (TortoiseSVN, GitHub);
4. Используйте сторонние эмуляторы или реальный девайс;
5. Учите английский.

WeatherAppTutorial

1. Правило №1 (правило велосипеда): поискать готовое решение.
 - Простое;
 - Минималистичное;
 - Без лишних функций;
 - Не запрашивает лишних данных;
 - Работает на любом аппарате.

Готовое решение:

1. Собрать и переработать под себя (проверенные и эффективные паттерны программирования);
2. Проверить на эмуляторе и реальном девайсе;
3. Добавить комментарии и логи для пояснений (Log.d, Log.e, Log.i, Log.v, Log.w, Log.wtf);

работы

1. IDE (в моем случае — это Эклипс, хотя есть IDEA и AndroidStudio);
2. Библиотека «Android support v7 appcompat» (ссылка на последней странице). Библиотеки совместимости — ваш лучший друг;
3. OpenWeatherMap API Key для запроса погодных данных. Потребуется бесплатная регистрация для получения «ключа» (ссылка на последней странице);
4. Иконки для отображения погоды (ссылки на последней странице);
5. Крайне желательно: эмулятор (в моем случае — это Genymotion, ссылка на последней странице) или реальный телефон на Андроиде (нужно активировать для разработчика).

Этапы создания проекта (Eclipse)

Создайте проект

Java - WeatherAppTutorial/src/com/example/weatherapptutorial/WeatherFragment.java - Eclipse

File Edit Refactor Source Navigate Search Project Run Window Help

New (Alt+Shift+N) >

- Open File...
- Close (Ctrl+W)
- Close All (Ctrl+Shift+W)
- Save (Ctrl+Shift+S)
- Save As...
- Save All (Ctrl+S)
- Revert
- Move...
- Rename...
- Refresh (F5)
- Convert Line Delimiters To >
- Print... (Ctrl+P)
- Switch Workspace >
- Restart
- Import...
- Export...
- Properties (Alt+Enter)
- 1 WeatherFragment.java [WeatherAppTut...]
- 2 WeatherActivity.java [WeatherAppTut...]
- 3 CityPreference.java [WeatherAppTuto...]
- 4 strings.xml [WeatherAppTutorial/...]
- 5 WeatherData.java [WeatherAppTutoria...]
- Exit

Project...

- Java Project
- Project...
- Package
- Class
- Interface
- Enum
- Annotation
- Source Folder
- Java Working Set
- Folder
- File
- Untitled Text File
- JUnit Test Case
- Task
- Example...
- Other... (Ctrl+N)

```
    } else {  
        Log.i("SimpleWeath  
        switch (id) {  
            case 2:  
                icon = get  
                break;  
            case 3:  
                icon = get  
                break;  
            case 5:  
                icon = get
```

New Project

Select a wizard

Create an Android Application Project

Wizards:

type filter text

- General
- Android
 - Android Application Project
 - Android Project from Existing Code
 - Android Sample Project
 - Android Test Project
- C/C++
- Gradle
- Java
- Maven
- Examples

< Back Next > Finish Cancel

```
private fi  
ivate vo  
Twoche  
try {  
Ha  
bi  
bi  
bi  
bi  
bi  
bi  
Ha  
re  
re  
re  
re  
re  
re  
re  
Au  
St  
Lo  
} catc  
St  
}  
109  
110  
111  
112 @Override
```

Этапы создания проекта (Eclipse)

New Android Application

The prefix 'com.example.' is meant as a placeholder and should not be used

Application Name: WeatherAppTutorial

Project Name: WeatherAppTutorial

Package Name: com.example.weatherapptutorial

Minimum Required SDK: API 11: Android 3.0 (Honeycomb)

Target SDK: API 21: Android 4.X (L Preview)

Compile With: API 24: Android 7.0

Theme: Holo Light with Dark Action Bar

Choose a target API to compile your code against, from your installed SDKs. This is typically the most recent version, or the first version that supports all the APIs you want to directly access without reflection.

< Back Next > Finish Cancel

New Android Application

Configure Project

Create custom launcher icon

Create activity

Mark this project as a library

Create Project in Workspace

Location: D:\Documents\workspace\WeatherAppTutorial Browse...

Working sets

Add project to working sets

Working sets: Select...

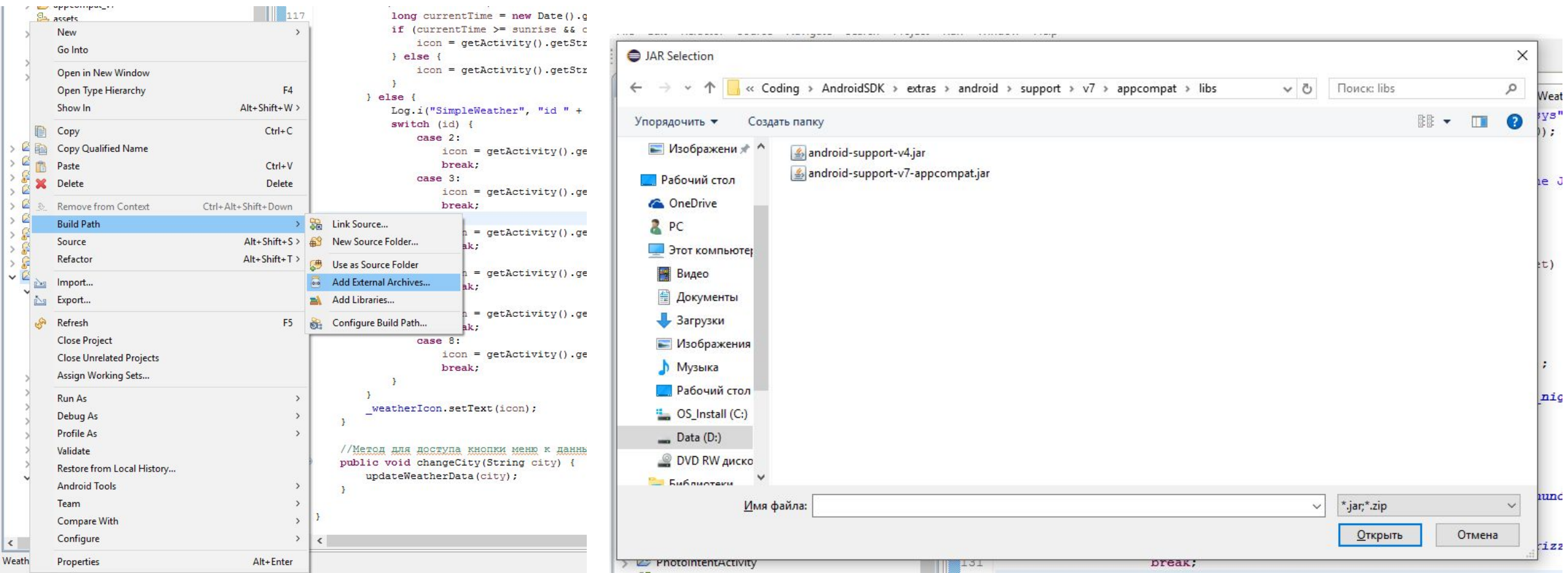
< Back Next > Finish Cancel

Этапы создания проекта (Eclipse)

Пропишите библиотеки обратной совместимости: правой кнопкой на проект для вызова контекстного меню. В окне эксплорера вы должны указать путь к библиотекам.

D:\Program Files\Coding\AndroidSDK\extras\android\support\v7\appcompat\libs

Возможно вам потребуется установить и android-support-v4.



Этапы создания проекта (Eclipse)

Добавьте их в ваш проект через правую кнопку

Context menu for WeatherAppTutorial project:

- New
- Go Into
- Open in New Window
- Open Type Hierarchy F4
- Show In Alt+Shift+W
- Copy Ctrl+C
- Copy Qualified Name
- Paste Ctrl+V
- Delete Delete
- Remove from Context Ctrl+Alt+Shift+Down
- Build Path
- Source Alt+Shift+S
- Refactor Alt+Shift+T
- Import...
- Export...
- Refresh F5
- Close Project
- Close Unrelated Projects
- Assign Working Sets...
- Run As
- Debug As
- Profile As
- Validate
- Restore from Local History...
- Android Tools
- Team
- Compare With
- Configure
- Properties Alt+Enter**

Properties for WeatherAppTutorial - Android

Target Name	Vendor	Platform	API ...
<input type="checkbox"/> Android 2.2	Android Open Source Project	2.2	8
<input type="checkbox"/> Google APIs	Google Inc.	2.2	8
<input type="checkbox"/> Android 2.3.3	Android Open Source Project	2.3.3	10
<input type="checkbox"/> Google APIs	Google Inc.	2.3.3	10
<input type="checkbox"/> Android 4.0.3	Android Open Source Project	4.0.3	15
<input type="checkbox"/> Google APIs	Google Inc.	4.0.3	15
<input type="checkbox"/> Android 4.1.2	Android Open Source Project	4.1.2	16
<input type="checkbox"/> Google APIs	Google Inc.	4.1.2	16
<input type="checkbox"/> Android 4.2.2	Android Open Source Project	4.2.2	17
<input type="checkbox"/> Google APIs	Google Inc.	4.2.2	17
<input type="checkbox"/> Android 4.3.1	Android Open Source Project	4.3.1	18
<input type="checkbox"/> Google APIs	Google Inc.	4.3.1	18
<input type="checkbox"/> Android 4.4.2	Android Open Source Project	4.4.2	19
<input type="checkbox"/> Glass Development...	Google Inc.	4.4.2	19
<input type="checkbox"/> Google APIs	Google Inc.	4.4.2	19
<input type="checkbox"/> Google APIs (x86 S...	Google Inc.	4.4.2	19
<input type="checkbox"/> Android 4.4W.2	Android Open Source Project	4.4W.2	20
<input type="checkbox"/> Android 5.0.1	Android Open Source Project	5.0.1	21
<input type="checkbox"/> Google APIs	Google Inc.	5.0.1	21
<input type="checkbox"/> Android 5.1.1	Android Open Source Project	5.1.1	22
<input type="checkbox"/> Google APIs	Google Inc.	5.1.1	22
<input type="checkbox"/> Android 6.0	Android Open Source Project	6.0	23
<input type="checkbox"/> Google APIs	Google Inc.	6.0	23
<input type="checkbox"/> Android N (Preview)	Android Open Source Project	N	N
<input checked="" type="checkbox"/> Android 7.0	Android Open Source Project	7.0	24

Library

Is Library

Reference	Project
-----------	---------

Buttons: Add..., Remove, Up, Down, Restore Defaults, Apply, OK, Cancel

Properties for WeatherAppTutorial - Android

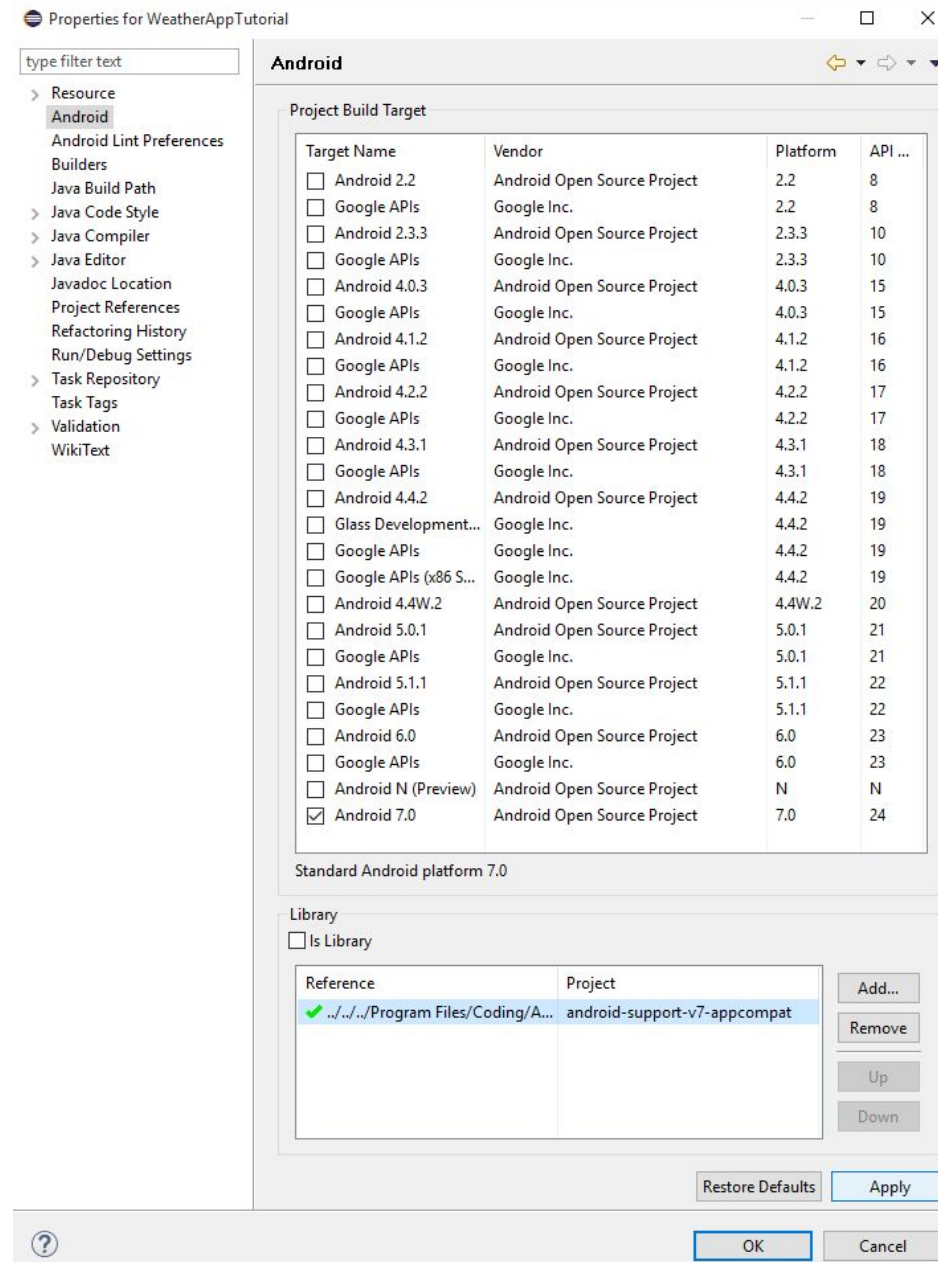
Project Selection dialog:

Please select a library project

- android-support-v7-appcompat
- design-support-library
- Meragana_-_Android_Extensions_Resources
- Meragana_-_Karaoke_Android_Resources
- recyclerview
- simple-crop-image-lib

Buttons: OK, Cancel

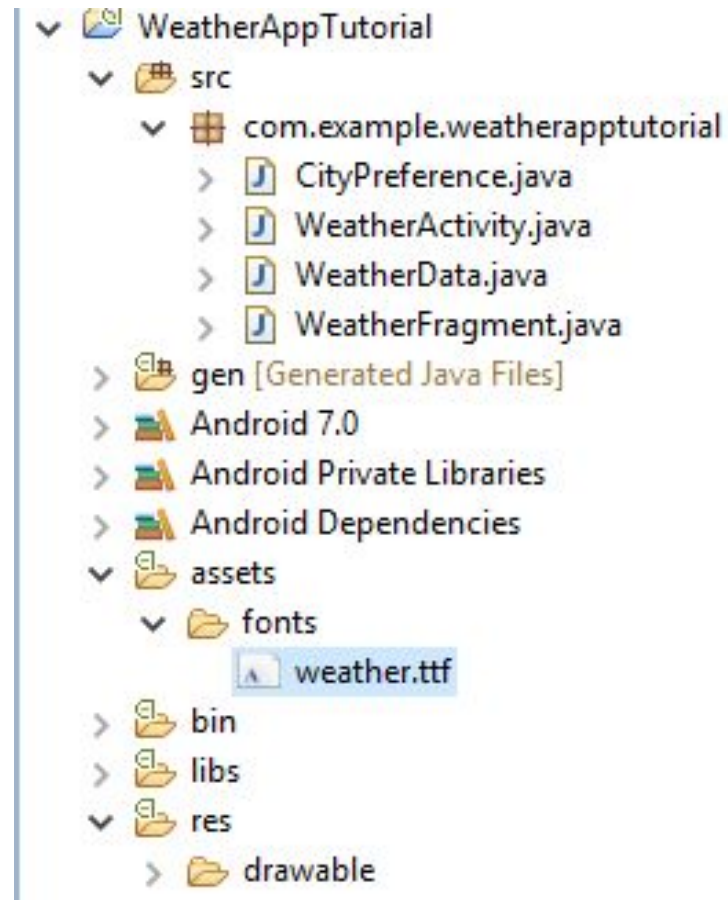
Этапы создания проекта (Eclipse)



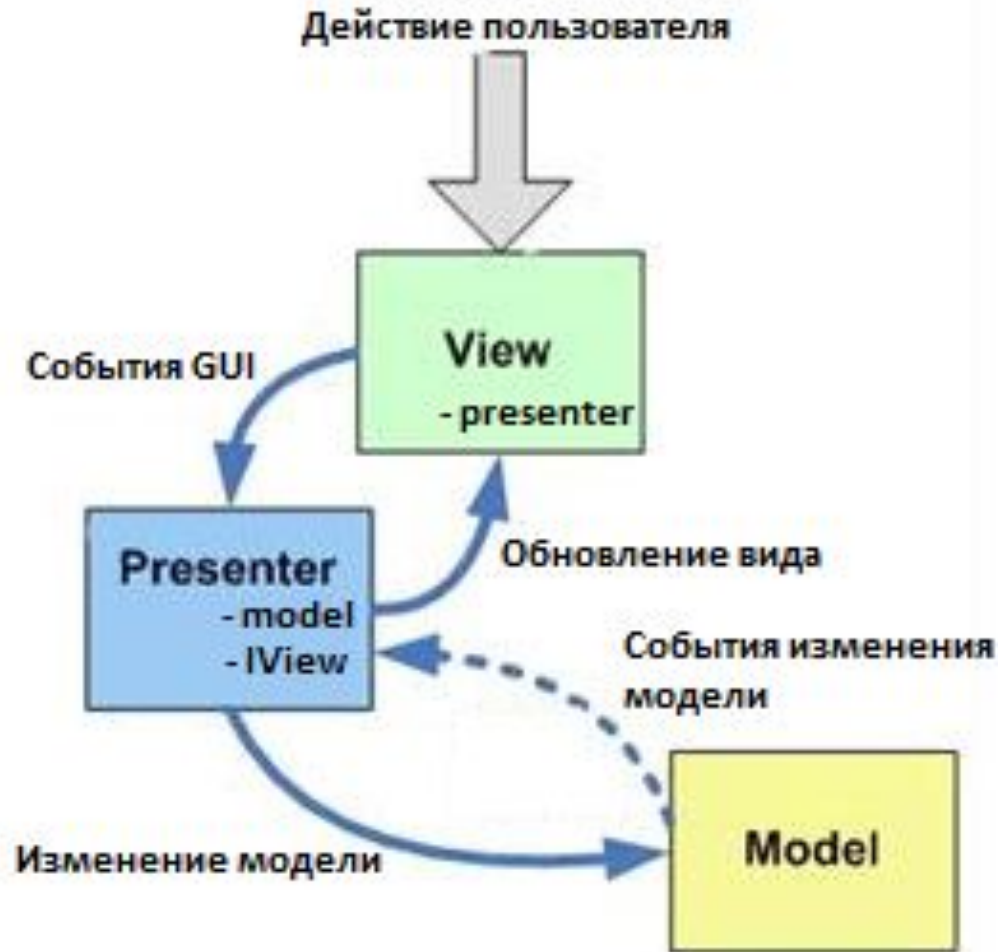
Этапы создания проекта (Eclipse)

Добавьте шрифт.

Для этого просто откройте папку workspace, найдите ваш проект, создайте там папку assets, в ней — fonts, и положите туда шрифт weather.ttf. Кликните правой кнопкой на проект в Эклипсе и Обновите. Папки и шрифт появятся в проекте.

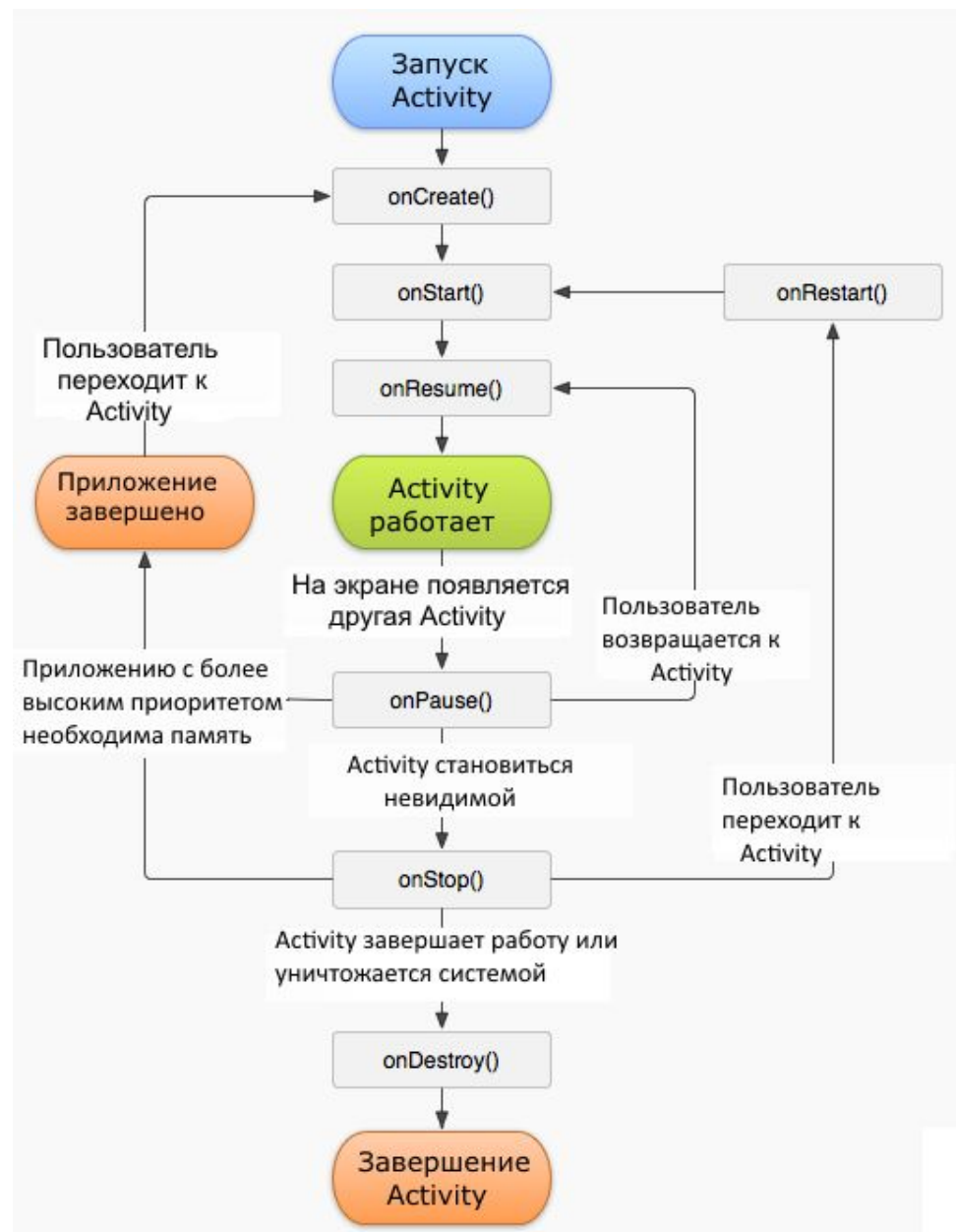


Паттерн MVP (Model-View-Presenter)

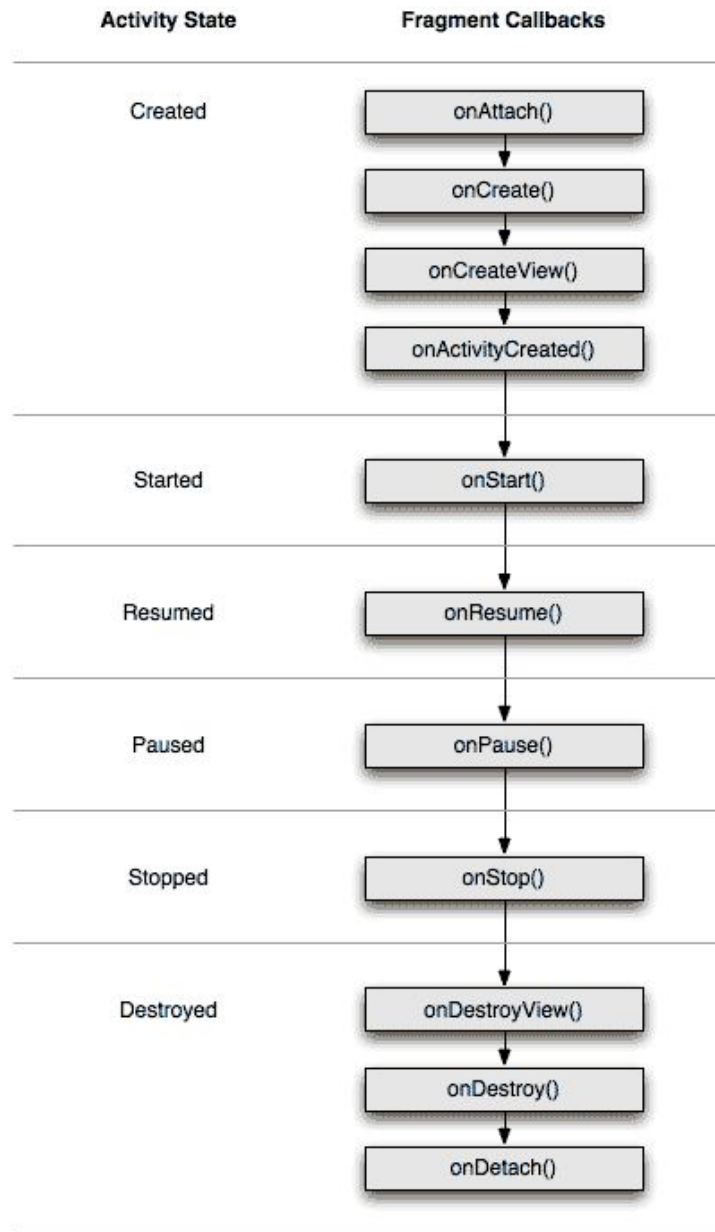


Model-View-Presenter

Жизненный цикл Activity

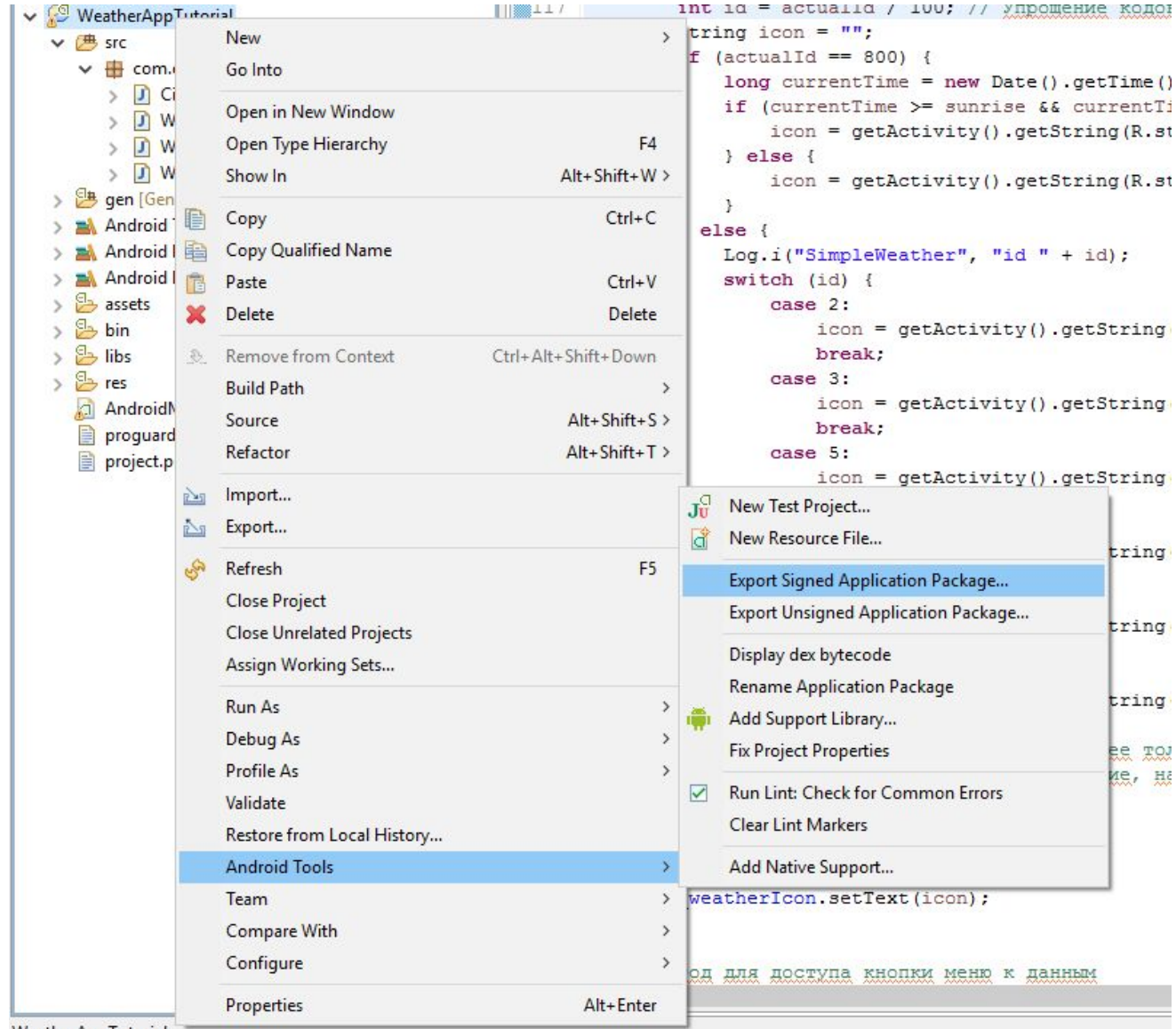


Жизненный цикл Fragment



Сохраняем проект в файле

Подготовка .apk: Android Tools – Export Signed Application Package.



Выводы

Да, все очень просто (любой может разобраться за день).

На самом деле — нет:

- Куча плохого кода;
- Антипаттерны;
- Низкая культура и уровень разработки.

Это именно то, что никому не нужно, в том числе и рынку.

С другой стороны — вам не нужно вникать во все подряд. Самое важное — понимать основы языка, как что работает и что с чем связано. Для всего остального есть Google, Stack Overflow, GitHub и Android developers.

То есть вам не обязательно понимать в деталях, например, как работает `BufferedReader` в классе `WeatherData`. Достаточно понимать принцип работы, чтобы имплементировать все, что нужно.

Пользуйтесь преимуществами ООП: больше классов для бога классов + переиспользование.

Пользуйтесь логами, всегда!

Конечно выводов и советов (на стр. 1) гораздо больше, но с этих можно начать)

Использованные ресурсы

1. Погодный сайт (нужна бесплатная регистрация для получения «ключа»)
<http://openweathermap.org/>
2. Крупнейший сайт с бесплатными иконками <https://www.iconfinder.com/>
3. Шрифт с погодными иконками <https://github.com/erikflowers/weather-icons>
4. Бесплатный эмулятор <https://www.genymotion.com/>
5. Лучшие друзья Android-разработчика:
 - Android support v7 appcompat:
<https://developer.android.com/topic/libraries/support-library/features.html>
 - StackOverflow: <http://stackoverflow.com/>
 - Android developers: <https://developer.android.com/index.html>
6. Код приложения <https://github.com/Sandor13/WeatherAppTutorial/tree/master>
(хотя правильнее будет самим создать проект, необходимые классы и скопировать последовательно весь код, разбирая его по ходу)
7. Шрифт и презентация <https://yadi.sk/d/zyvIwEBWxLogN>