Advanced Layers Theme Installer

By Niklas Schnettler

Table of contents

Advanced Layers Theme Installer

By Niklas Schnettler

About the Installer

I.   Features:

    a.   Material Design

    b.   The user can choose the Overlays he wants to install – not more, not less

    c.   Include your Overlays in different colors and let the user decide which he/she likes most

    d.   Include additional/optional Overlays for each Overlay to choose from

    e.   Include different versions of your Overlay (for example two different NavBars)

    f.   Much more…

Permission to use it / License

I.   The Installer App is licensed under the MIT License. This means that you are allowed to use the app for your commercial or non commercial projects. You need just to include the copyright notice into your project.

II.   If you want to contribute to the development to this app, you can add commits to the Github and I will surely merge it.

III. If you want to use the app, you can change as much as you like, but you **have to keep the card with my name, picture and link** at the **Top of the About page.** It would also be great if you mention me in your posts about your Theme.

How to use the App?

Requirements:

I.   Latest version of Android Studio (currently 1.2):

    https://developer.android.com/sdk/index.html

II.   Latest Java SE Development Kit

III. Android 5.0 & 5.1 SDK

IV.   Latest version of Android Support Library

V.   App Source Downloaded from Github:

    https://github.com/Sh4dowSoul/AdvancedLayersThemeInstaller/releases
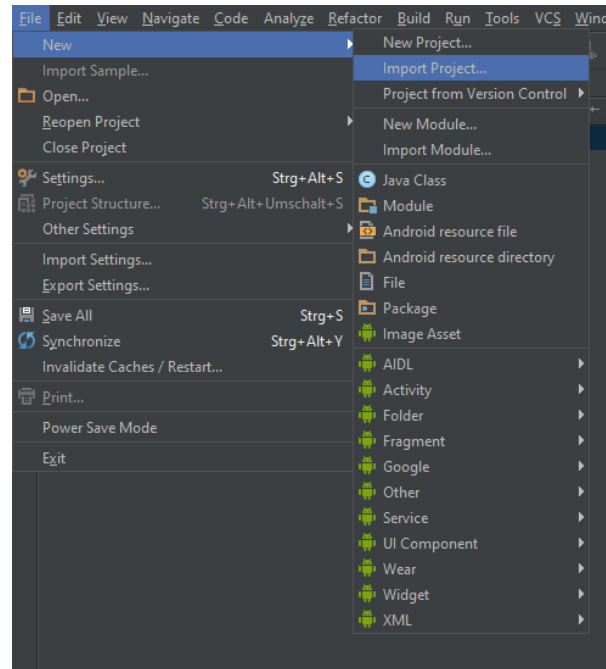
Advanced Layers Theme Installer

By Niklas Schnettler

Getting Started

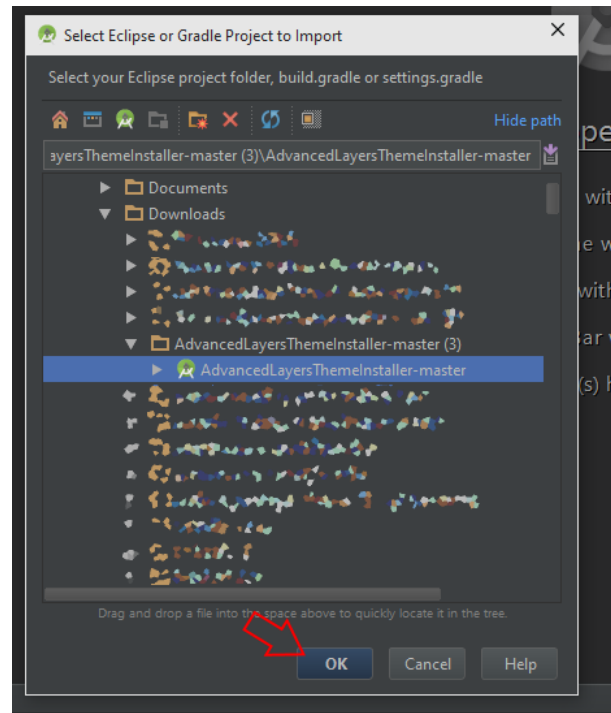I. Download and extract the source of my app.

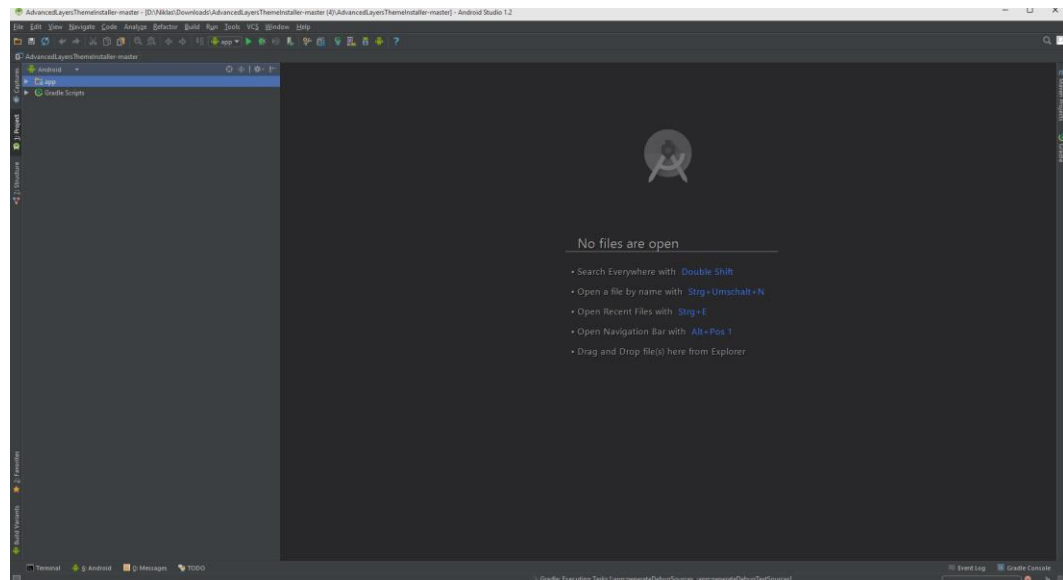## Changes in Android Studio

a. File, New, Import Project

Advanced Layers Theme Installer

By Niklas Schnettler

b. In the opened dialog, navigate to the location where the downloaded file is. In the folder you downloaded there is another one. Just one. Select this and click OK.
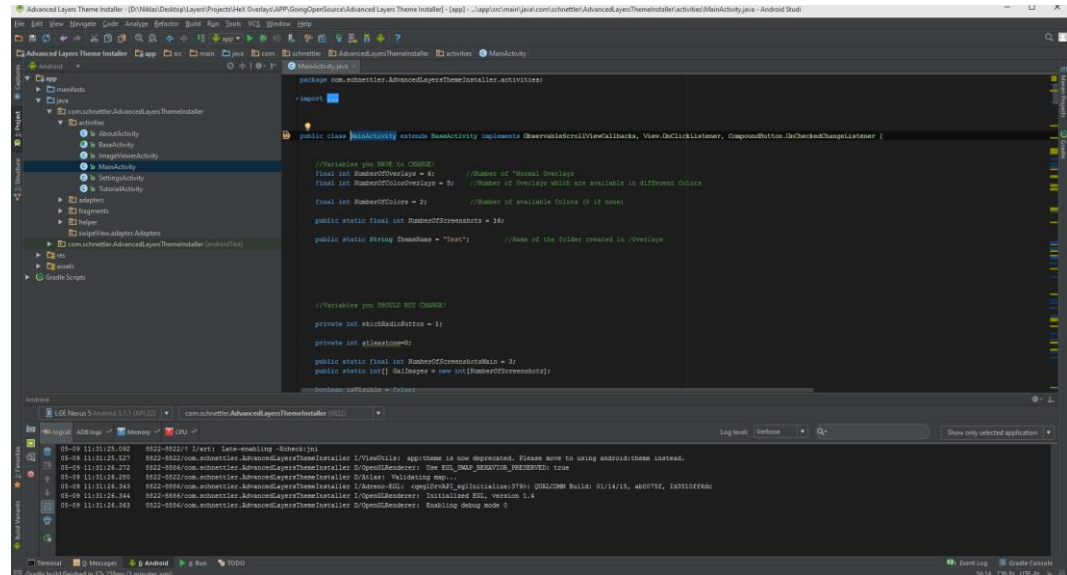

c. Android Studio should now show this screen:

At the left side, expand **app**, then **Java, com.schnettler.AdvancedLayersThemeInstaller, activities** and open **MainActivity.**

Advanced Layers Theme Installer

By Niklas Schnettler

You should now see a similar screen to this:

d. Changes to **MainActivity**:

    i. At the very top you will see a few variables, which have to be changed

        1. **NumberOf Overlays**: Add the Number of Overlays you want to include here

        2. **NumberOfColorOverlays**: Add Number of color specific Overlays here (0 if you don't use any)

        3. **NumberOfColors**: Add number of Colors your theme is available in here (0 if you only have one color)

        4. **NumberOfScreenshots**: Add the Number of Screenshots you provide here

        5. **ThemeName**: Add the Name of your theme here. This Name has to be used for the Overlay apk naming too (more about that later) Don´t use spaces in your Theme Name, it won´t work.

    ii. Scroll down until you see the **ONCREATE** method.

        1. **OverlayName[NumberOfApp]**: Add the name of the App the Overlays are for. The number of this has to be the same as

NumberOfOverlays defined before. Spaces are allowed here. After you entered all App Names, create another OverlayName directly after the others and save "Nothing" in it.

2. **OPTIONAL IF YOU USE DIFFERENT COLORS:**

   a. **ColorOverlayName[NumberOfApp]:** Add the names of your color specific Overlays here. Number has to be the same as defined as NumberOfColorOverlays. Spaces are allowed here too.

   b. **Color[NumberOfColor]:** Add the Name of your Colors here. Please be aware that there need to be as many as you defined before as NumberOfColors.

3. **OPTIONAL IF YOU USE ADDITIONAL/OPTIONAL OVERLAYS:**

   a. **InstallAdditionalOverlays[NumberOfAppWhichHasAdditionalOverlays][0] = 1**; : Add this if a app has additionalOverlays

   b. **NumberOfAdditionalOverlays[NumberOfAppWhichHasAdditionalOverlays**] : Add the Number of additional Overlays this app has.

   c. **ShowAdditionalOverlaysDialog[NumberOfAppWhichHasAdditionalOverlays] = true** : Add this if you want to show a dialog where the user can select the optional Overlays he wants to use.

   d. **AdditionalDialogMode[NumberOfAppWhichHasAdditionalOverlays] = 1**: Add this when you want to use a radio button dialog instead one with checkboxes.

   e. **AdditionalOverlayName[NumberOfAppWhichHasAdditionalOverlays][NumberOfAdditionalOverlayForThisApp]:** Add the Names of the additionalOverlays for this app here. You have to define as much names as you defined in NumberOfAdditionalOverlays[NumberOfAppWhichHasAdditionalOverlays].

e. Changes to **ThemeSpecificStrings:**

    i. At the left panel navigate to: **app/res/values/ThemeSpecificStrings.xml**

    ii. Change the strings to be compatible with your theme.

    iii. AppName : This time you can include spaces in your App Name.

    iv. Please don´t change strings stored in: DontChangeStrings.xml(if you change them, the app wont work) and ManagerAppStrings (You can change them, but suggest you to not.

Making your theme files ready for the use with my app.

f. How you have to **name your apks:**

ThemeName_NameOfTheApp.apk

Example: HeX_PlayStore.apk

    i. ThemeName: The same as defined in the MainActivity of the App. (nospaces)

    ii. NameOfTheApp: Name of the Overlay defined in the oncreate method of the Main Activity of the installer App. (only without the spaces used there)

g. How to **name the zip Files** of your Theme.

    i. Zip with normal Overlays: ThemeName_General.zip

    Example: HeX_General.zip

    ii. Zip with color Overlays: ThemeName_NameOfTheColor.zip

    Example: HeX_Blue.zip , HeX_red.zip

        1. NameOfTheColor: Color defined in the oncreate method of MainActivity

        2. You have to create a zip like that for every color available.

h. How to **name the screenshots** of your theme and where to put them.

    i. Screenshot1, screenshot2, screenshot3,….

    ii. Include as many as you defined in the MainActivity of the app.

    iii. You have to copy them to: /app/src/main/res/drawable-xxxhdpi

    iv. Use a size similar to 1400 * 2367

    i.   change the **images** shown at the **tutorial** on first startup.

        i.   Files are located at: \app\src\main\res\drawable-xxhdpi

        ii.   Use screenshot of a similar size like 700 * 939

        iii.   Name of the images: tutorial2-png – tutorial7.png

    j.   change the **theme developer image** displayed at the about section of the app.

        i.   File located at /app/src/main/res/drawable-xxhdpi

        ii.   Name of the file: themedeveloper.png

        iii.   Use a round image with a size of about 200 * 200

    k.   Change **hero image.**

        i.   File located at/app/src/main/res/drawable-xxhdpi

        ii.   File name: heroimage.png

        iii.   Use a image with a size of 1598 * 1052 to get the best possible result.

    l.   Change **launcher icon.**

        i.   File located in all drawable folders

        ii.   Name of the file: ic_launcher.png

Include your theme files into the app.

    m. Copy the Overlay apks into the zip file they belong to.

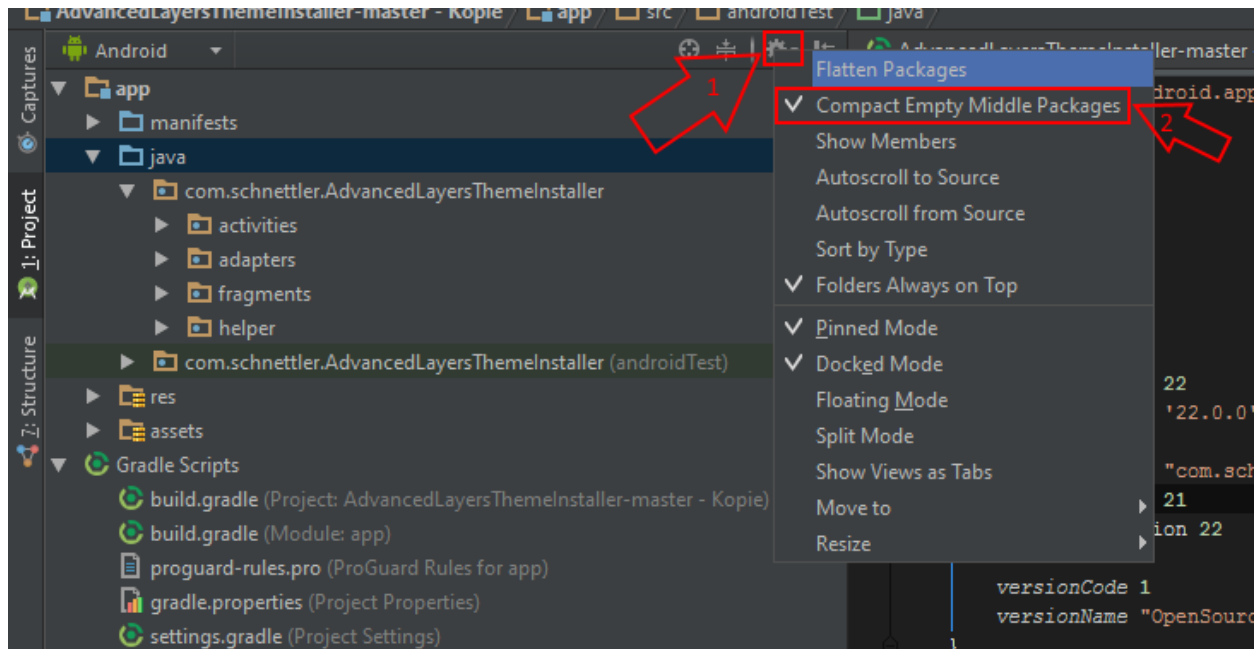   **n.** Copy the zip files to: \app\src\main\assets\Files

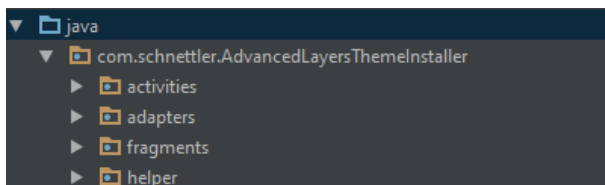# Advanced Layers Theme Installer

By Niklas Schnettler

Change Package Name in Android Studio

    o. Look at the project panel in android studio (the one on the left). In the **top right corner** you will see a little **gear icon**. Click on it. You will see the following:
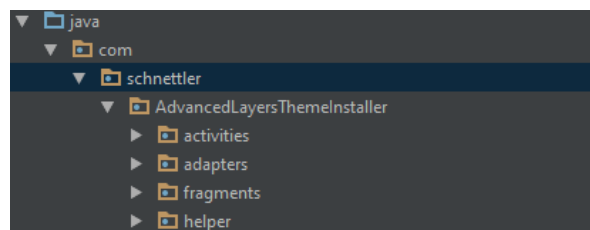

    p. In this dialog you have to click on **Compact Empty Middle Packages**

    q. The structure of the folders in the project panel will change a bit:
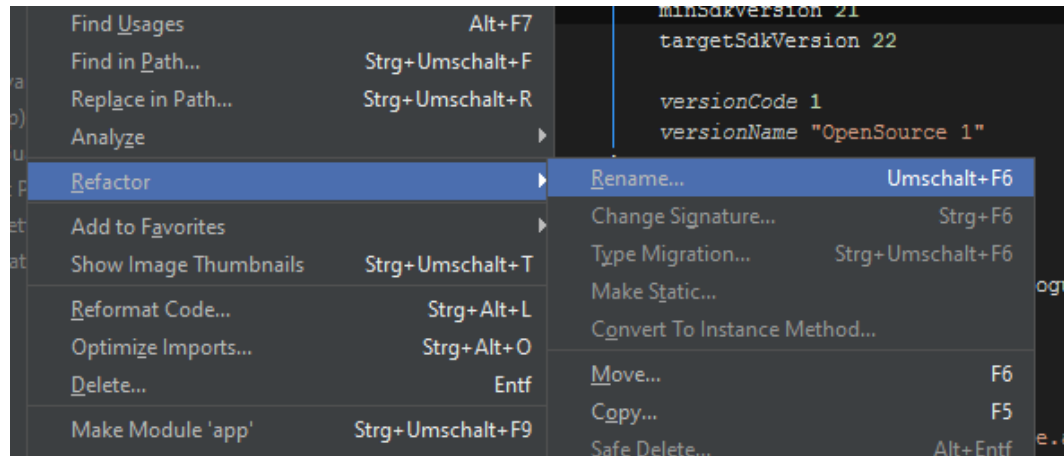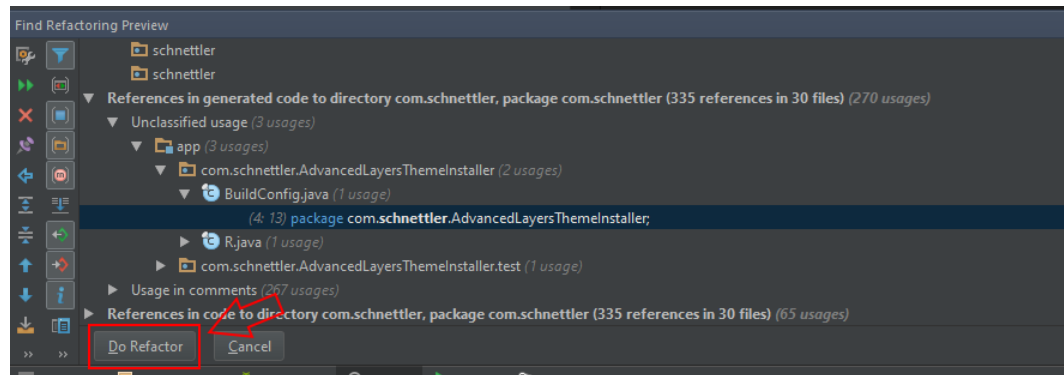
Before:

After

    r.   Select the schnettler folder and rightclick it. Choose refactor and then rename.

    s.   A warning window will pop up. Just click on Rename package.

    t.   Enter the desired first part of your package name and click refactor.

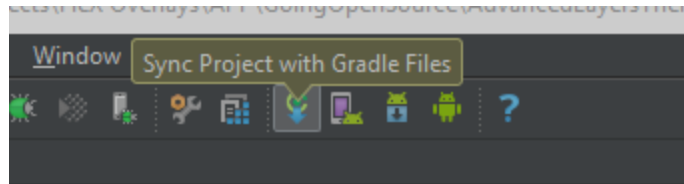   u.   At the bottom of the screen a refactoring preview will appear. Just click on DO REFACTOR at the left corner.

    v.   Do the same again with the AdvancedLayersThemeInstallerFolder.
          ←

    w.   At the left project panel click on Gradle Scripts and then select build.gradle(Module: app).

    x.   After you opened the file, search for the applicationId. It should still be: "com.schnettler.AdvancedLayersThemeInstaller"

y. Change it to:

"com.FirstPartOfTheNewPackageName.SecondPartOfTheNemPackageName"

    i. FirstPartOfTheNewPackageName is that what you replaced "Schnettler" with at the steps before

    ii. SecondPartOfTheNemPackageName is what you replaced AdvancedLayersThemeInstaller with at the steps before.

z. Optional: change versionCode and versionName to your desired value.

aa. Make a gradle sync

bb. SUCCESS. You should be ready now to publish your theme. To try it on your device press run/run app at the top.

To build it for publishing it at the playstore, click on build/generate Signed Apk.

Further Support

I. I hope this tutorial / documentation helped you to work with my app. If you still have questions or anything which doesn't work, message me on hangouts.

II. If you think I forgot to mention sth. In this tutorial, please message me also on Hangouts.

Thank you for considering my app. Reagrds,

Niklas