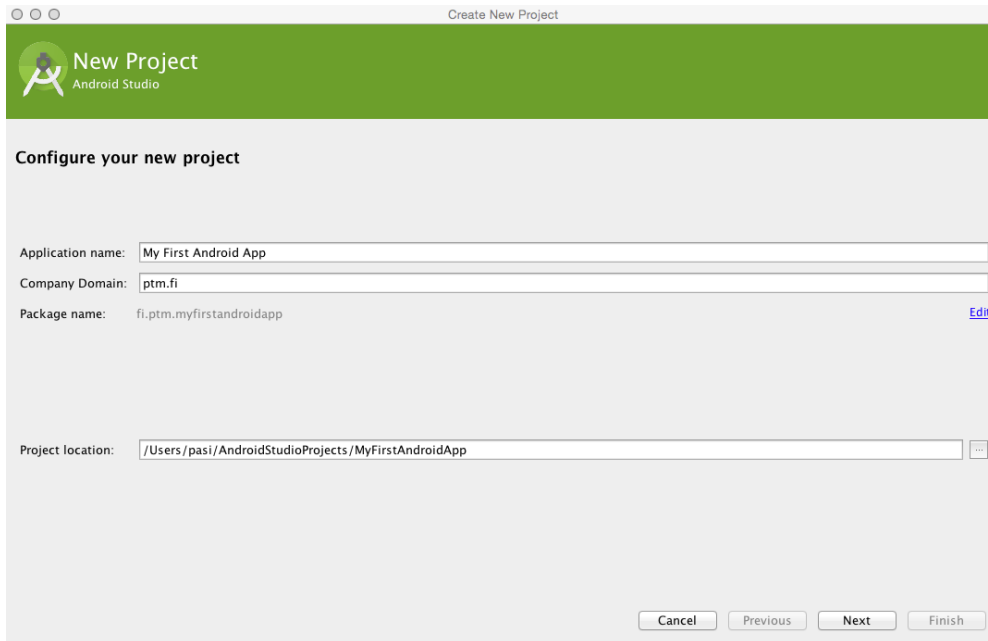


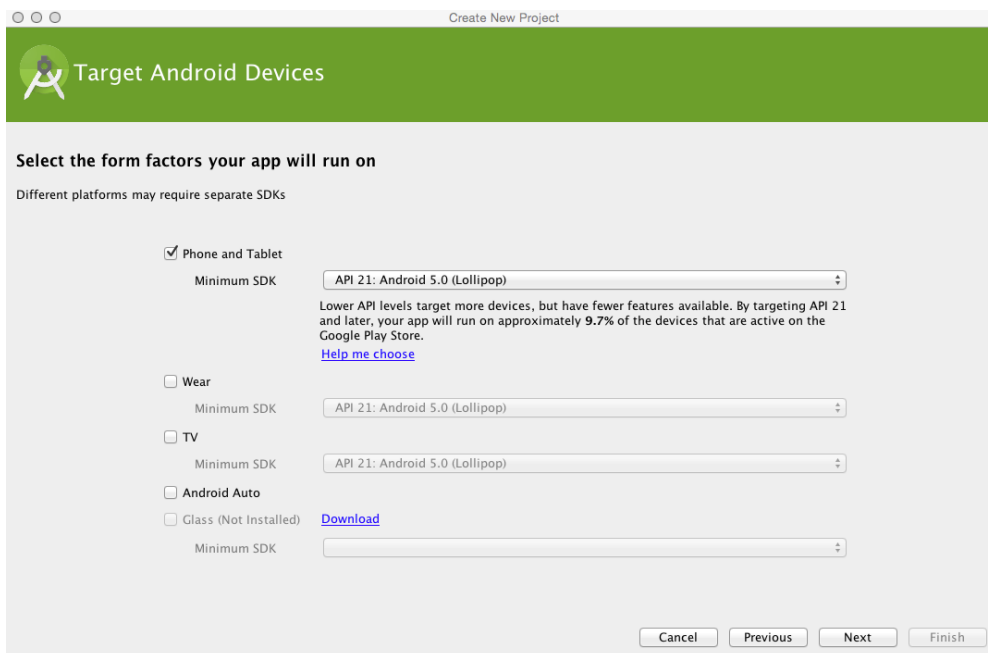
Creating a first Android application

1. Launch Android Studio
2. Start a new Android Studio Project
3. Configure your new project
 - a. Give unique application name
 - b. Use your own company domain name, or for example jamk.fi



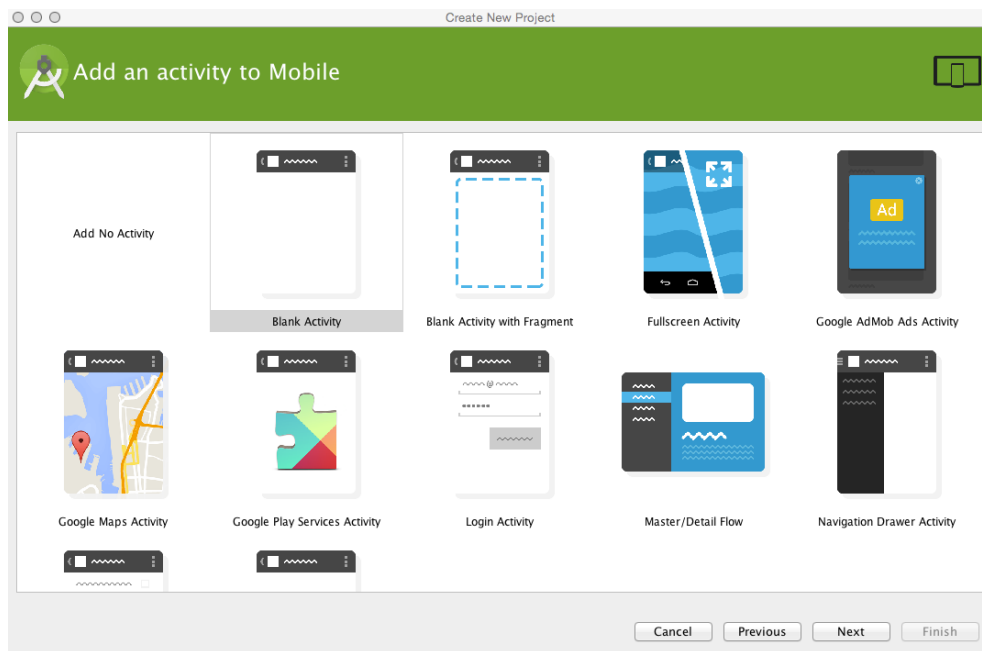
The screenshot shows the 'New Project' dialog in Android Studio. The title bar says 'Create New Project'. The header has the Android Studio logo and the text 'New Project Android Studio'. The main section is titled 'Configure your new project'. It contains four input fields: 'Application name' with the value 'My First Android App', 'Company Domain' with the value 'ptm.fi', 'Package name' with the value 'fi.ptm.myfirstandroidapp' and an 'Edit' link, and 'Project location' with the value '/Users/pasi/AndroidStudioProjects/MyFirstAndroidApp' and a file explorer icon. At the bottom, there are four buttons: 'Cancel', 'Previous', 'Next', and 'Finish'.

4. Target your project to Phone and Tablet
 - a. Use Android 5.0 Lollipop (for example)

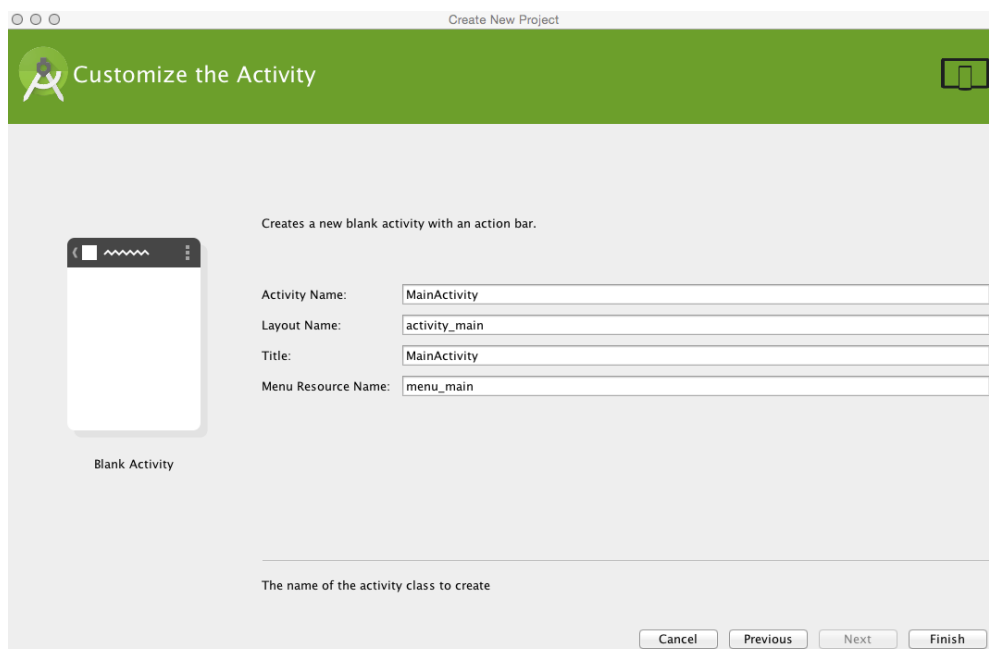


The screenshot shows the 'Target Android Devices' dialog in Android Studio. The title bar says 'Create New Project'. The header has the Android Studio logo and the text 'Target Android Devices'. The main section is titled 'Select the form factors your app will run on'. Below this, it says 'Different platforms may require separate SDKs'. There are five options, each with a checkbox and a 'Minimum SDK' dropdown menu. The first option, 'Phone and Tablet', is checked. Its dropdown menu shows 'API 21: Android 5.0 (Lollipop)'. Below this dropdown, there is a paragraph of text: 'Lower API levels target more devices, but have fewer features available. By targeting API 21 and later, your app will run on approximately 9.7% of the devices that are active on the Google Play Store.' and a link 'Help me choose'. The other four options are 'Wear', 'TV', 'Android Auto', and 'Glass (Not Installed)', all of which are unchecked. At the bottom, there are four buttons: 'Cancel', 'Previous', 'Next', and 'Finish'.

5. Add Activity for your application (first screen)
 - a. Select Blank Activity



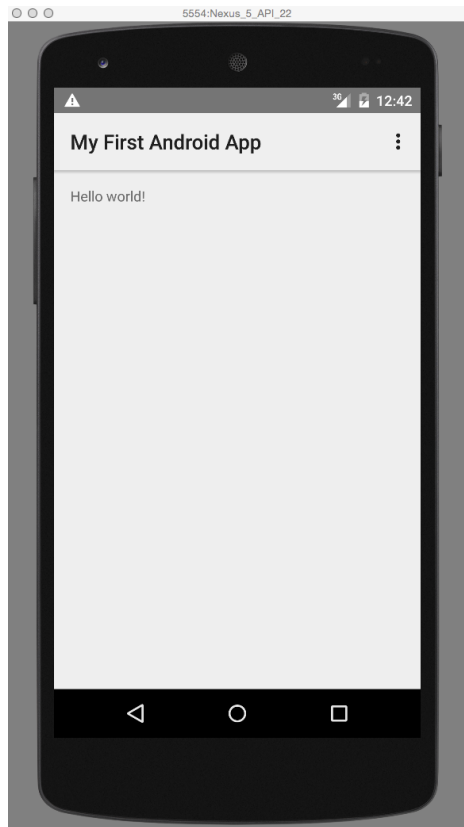
6. Customize your app
 - a. For now, you can just use default naming



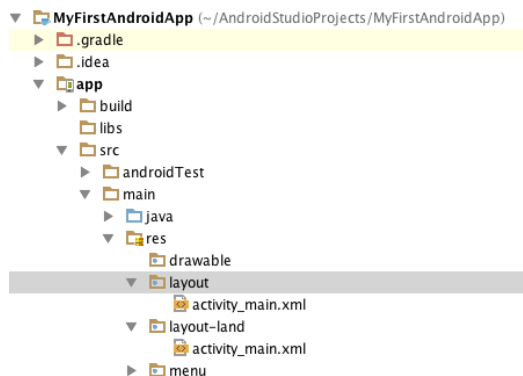
7. Click Finish and Android Studio will create project for you.

Application and needed settings are now generated and it can be run. Before that you should create one Android Virtual Device or use your own phone with USB cable.

8. Creating a Android Virtual Device
 - a. Select Tools > Android > AVD Manager
 - b. Create a new Virtual Device (if needed)
 - i. First select a hardware (for example Nexus 5)
 - ii. Then select system image (for example Lollipop, 22, with google)
 - iii. And finally accept default configuration and create AVD
9. Run and test your application in Android Virtual Device



10. Add a new Activity for Landscape use.
 - a. Create a new folder in res and name it layout-land
 - b. Copy activity_main.xml from layout to layout-land



11. Add a new String to Strings.xml in values folder

```
<string name="hello_world_land">Hello landscape world!</string>
```

Edit activity_main.xml from layout-land to use above string

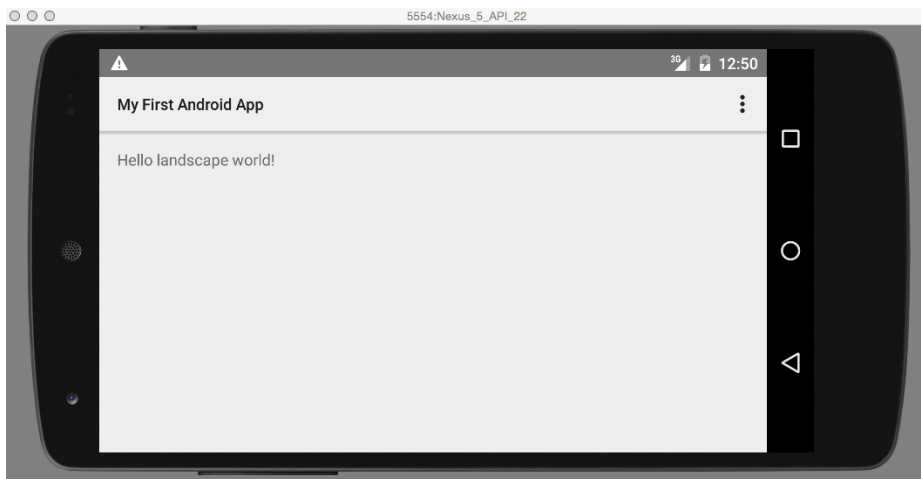
```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="@string/hello_world_land" />
```

12. Run application again and switch emulator from portrait to landscape

In mac fn-ctrl-f11

In pc left-ctrl-f11

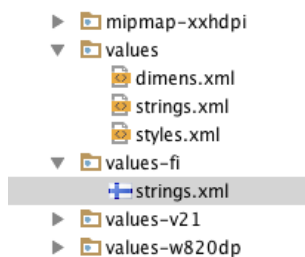
In linux ctrl-f11



Note! Android switches layouts automatically!

13. Supporting different languages

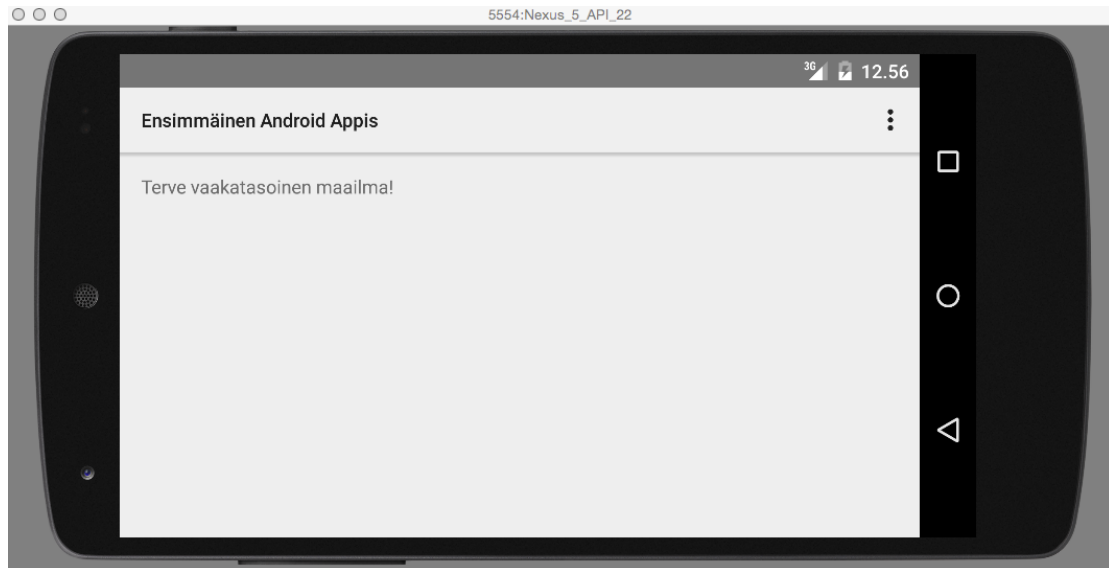
- Create a new folder values-fi (or other language) in res folder
- Copy Strings.xml from values folder to values-fi folder
- Edit both hello_world lines to contains finnish (or other language) words



```
<string name="hello_world">Terve Maailma!</string>
```

```
<string name="hello_world_land">Terve vaakatasoinen maailma!</string>
```

14. Switch your Emulator to use Finnish (Suomi) as a default language (or other language), install and run your application again



So these application resources are very important in Android application development.

15. Add a few more TextViews and colors

Add a new colors.xml file to your res/values folder.
Define a few colors:

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="blue">#0000FF</color>
    <color name="red">#FF0000</color>
</resources>
```

Add a new String to strings.xml file to contain some sample text.

```
<string name="another_string">Here is a another sample
text.</string>
```

Edit activity_main.xml (layout folder) and add a new TextView to contain above new text and use for example blue text color. Use designer or text view to add a new TextView.

```
<TextView
    android:id="@+id/textView1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@id/textView"
```

```
android:layout_below="@id/textView"  
android:layout_marginTop="16dp"  
android:text="@string/another_string"  
    android:textColor="@color/blue" />
```

Test!