



Android Development



— MNNIT Computer Coding Club —

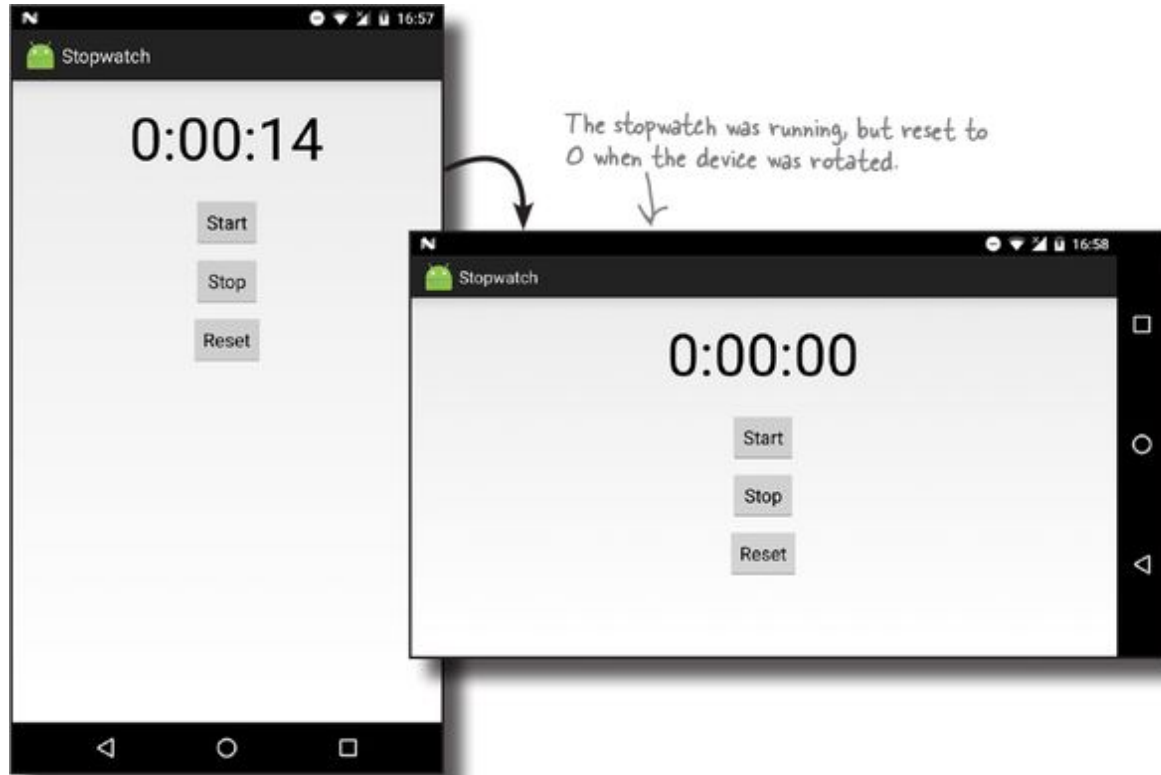
Objective

To make you answer following questions yourselves:

1. What is AndroidManifest.xml?
2. What are permissions?
3. How to dependencies?
4. What is an Adapter?
5. How to display lists?



Previous class Left Off



Handling Screen Rotation

Let's see how we can handle screen rotation.

Remember! This can be used in many places. You will encounter them somewhere.

AndroidManifest.xml

Every app project must have an AndroidManifest.xml file (with precisely that name) at the root of the project source set. The manifest file describes essential information about your app to the Android build tools, the Android operating system, and Google Play.

Among many other things, the manifest file is required to declare the following:

1. The app's package name.
2. The components of the app, which include all activities, services, broadcast receivers, and content providers.
3. The permissions that the app needs in order to access protected parts of the system or other apps.
4. The hardware and software features the app requires, which affects which devices can install the app from Google Play.

AndroidManifest.xml continued....

1. Now, let's see AndroidManifest.xml file of the project we just build. (To be honest, whatever there is written in it is written automatically till now, since we are using Android Studio and it is handling things for us)
2. Let's now see one more AndroidManifest.xml file :
<https://github.com/ashu12chi/RiseUp/blob/master/Android/app/src/main/AndroidManifest.xml>

Permissions

App permissions help support user privacy by protecting access to the following:

1. **Restricted data**, such as system state and a user's contact information.
2. **Restricted actions**, such as connecting to a paired device and recording audio.

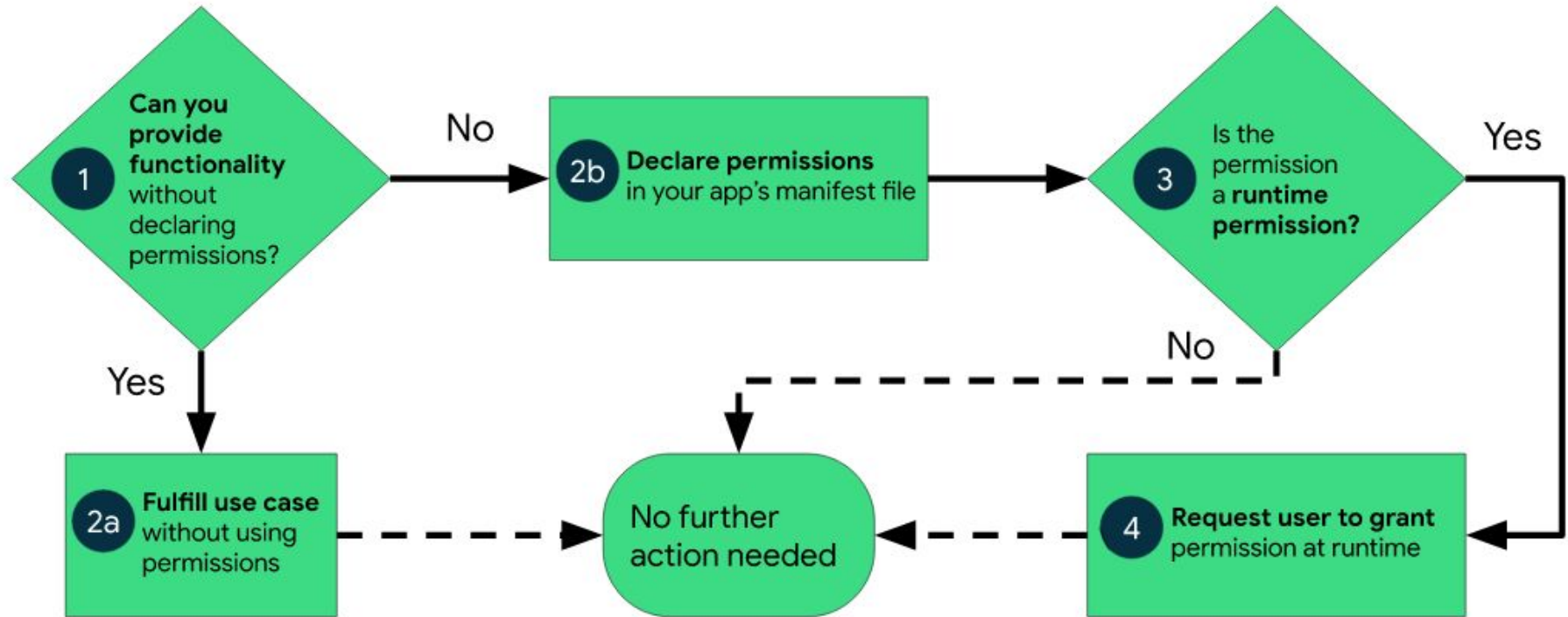
If your app offers functionality that might require access to restricted data or restricted actions, determine whether you can get the information or perform the actions without needing to declare permissions.

Permissions Continued...

You can fulfill many use cases in your app, such as taking photos, pausing media playback, and displaying relevant ads, without needing to declare any permissions.

If you decide that your app must access restricted data or perform restricted actions to fulfill a use case, declare the appropriate permissions. Some permissions, known as **install-time permissions**, are automatically granted when your app is installed. Other permissions, known as **runtime permissions**, require your app to go a step further and request the permission at runtime.

Workflow for using Permissions



Install time Permissions

Install-time permissions give your app limited access to restricted data, and they allow your app to perform restricted actions that minimally affect the system or other apps. When you declare install-time permissions in your app, the system automatically grants your app the permissions when the user installs your app. An app store presents an install-time permission notice to the user when they view an app's details page.

Android includes several sub-types of install-time permissions, including normal permissions and signature permissions.

Install time Permissions Continued...

Version 1.234.5 may request access to



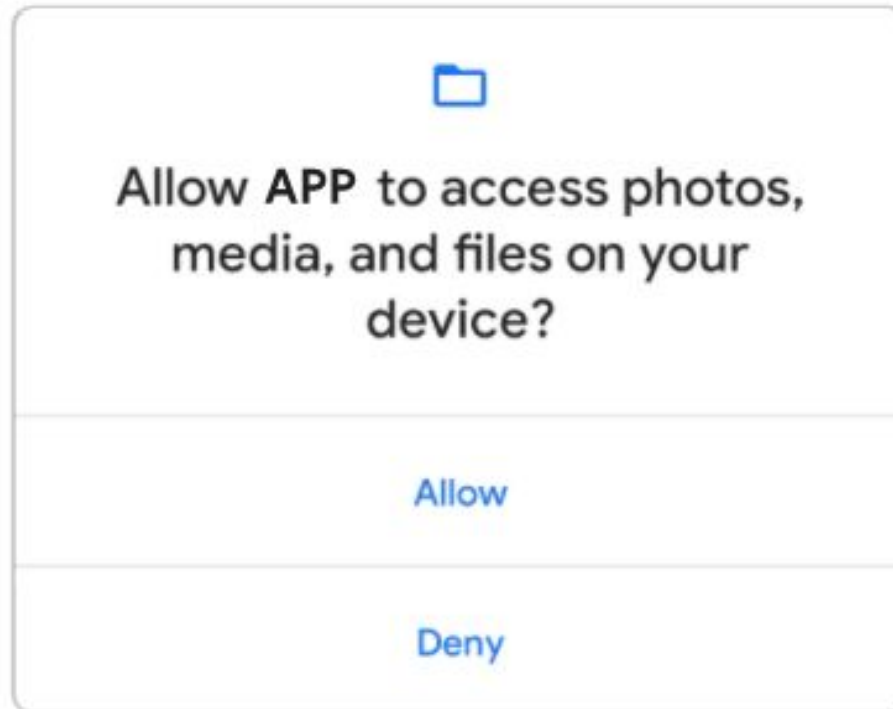
Other

- have full network access
- view network connections
- prevent phone from sleeping
- Play Install Referrer API
- view Wi-Fi connections
- run at startup
- receive data from Internet

Runtime Permissions

Runtime permissions, also known as dangerous permissions, give your app additional access to restricted data, and they allow your app to perform restricted actions that more substantially affect the system and other apps. Therefore, you need to request runtime permissions in your app before you can access the restricted data or perform restricted actions. When your app requests a runtime permission, the system presents a runtime permission prompt.

Runtime Permissions Continued...



Home Assignment

Which Android version started runtime permissions?

Dependencies

The Gradle build system in Android Studio makes it easy to include external binaries or other library modules to your build as dependencies. The dependencies can be located on your machine or in a remote repository, and any transitive dependencies they declare are automatically included as well.

To add a dependency to your project, specify a dependency configuration such as implementation in the dependencies block of your **build.gradle** file.

Dependencies Continued...

```
apply plugin: 'com.android.application'

android { ... }

dependencies {
    // Dependency on a local library module
    implementation project(":mylibrary")

    // Dependency on local binaries
    implementation fileTree(dir: 'libs', include: ['*.jar'])

    // Dependency on a remote binary
    implementation 'com.example.android:app-magic:12.3'
}
```


Dependencies Continued...

Inside the dependencies block, you can declare a library dependency using one of several different *dependency configurations* (such as implementation shown above). Each dependency configuration provides Gradle with different instructions about how to use the dependency.

Examples - implementation, compileOnly, runtimeOnly.

These are not important for you at this stage. If you refer to some old code you will see word 'compile' instead of implementation. This word 'compile' is now deprecated and can be replaced with implementation for your use case.

Home Assignment

Read what is the difference between Implementation and Compile dependencies.

Dependencies Continued...

Let's move to our own project and add a dependency to it.

Adapter

An **AdapterView** is a view whose children are determined by an **Adapter**. See ListView, GridView, Spinner and Gallery for commonly used subclasses of AdapterView.

An Adapter object acts as a bridge between an AdapterView and the underlying data for that view. The Adapter provides access to the data items. The Adapter is also responsible for making a View for each item in the data set.

ListView

Displays a vertically-scrollable collection of views, where each view is positioned immediately below the previous view in the list. For a more modern, flexible, and performant approach to displaying lists, use RecyclerView.

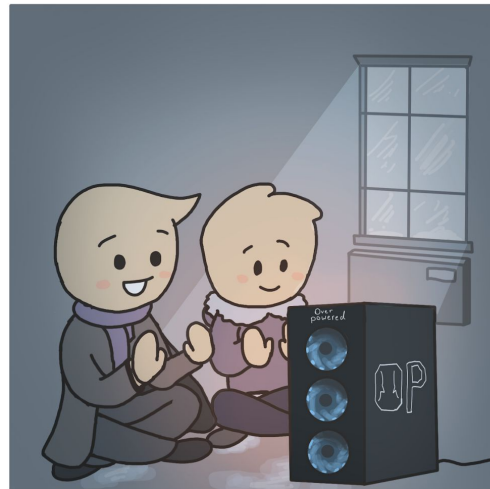
Home Assignment

Explore following topics:-

1. SharedPreferences.
2. RecyclerView.

Home Project

<https://github.com/rjankit/MultilanguageApp>





Smart phones are re-inventing the
connections between companies
and their customers.

— *Rich Miner* —

AZ QUOTES

Thank You.