# Notes

## General Software

### Microsoft Word

#### How do you display code snippets in MS Word preserving format and syntax highlighting?

##### Steps

1. Go to "insert" tab, click "object" button (it's on the right)
2. choose "OpenDocument Text" which will open a new embedded word document
3. copy and paste your code from Visual Studio / Eclipse inside this embedded word page
4. save and close

##### References

<http://stackoverflow.com/questions/387453/how-do-you-display-code-snippets-in-ms-word-preserving-format-and-syntax-highlig>

### Miscellaneous

#### How to copy a directory structure with file names without content?

##### Steps

1. Open “Cygwin command prompt” on windows
2. Enter the following command

***find src/ -type d -exec mkdir -p dest/{} \; \***

***-o -type f -exec touch dest/{} \;***

1. Above command means, find directory (-d) under (src/) and create (mkdir -p) them under dest/ or (-o) find files (-f) and touch them under dest/.

##### References

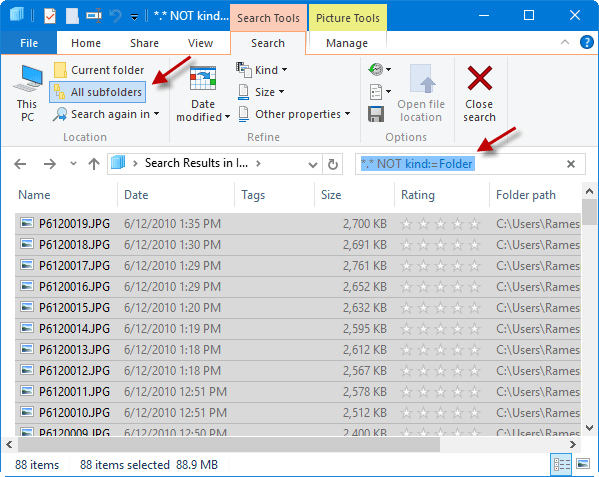
<http://stackoverflow.com/questions/11946465/copy-a-directory-structure-with-file-names-without-content>

#### How to Copy Files from Multiple Sub-Folders to a Single Folder?

##### Steps

1. Open the parent folder (the top-level folder) which contains multiple-sub-folders.
2. In the Search box type the following search query, exactly as below:

***\*.\* NOT kind:=Folder***



1. Just make sure that All subfolders option is enabled. It starts searching for files in all sub-folders and recursively. We use the “NOT” operator to prevent Windows search from listing sub-folders — as we’re only going to copy or move all the files in sub-folders, and not the folders themselves.
2. Wait until the search is completed, and then select all files in the search results, by pressing the key ***CTRL + A***.
3. Paste the copied files to the desired destination.

##### References

<http://www.winhelponline.com/blog/move-copy-files-multiple-sub-folders-single-folder/> - Find the “Method 2: Using Windows Search (GUI)” section

#### How to get Windows Photo Viewer back in Windows 10?

##### Steps

1. Copy the code below into Notepad and save it as a REG file (call it whatever you like, maybe photoviewer.reg).

***Windows Registry Editor Version 5.00***

***[HKEY\_CLASSES\_ROOT\Applications\photoviewer.dll]***

***[HKEY\_CLASSES\_ROOT\Applications\photoviewer.dll\shell]***

***[HKEY\_CLASSES\_ROOT\Applications\photoviewer.dll\shell\open]***

***"MuiVerb"="@photoviewer.dll,-3043"***

***[HKEY\_CLASSES\_ROOT\Applications\photoviewer.dll\shell\open\command]***

***@=hex(2):25,00,53,00,79,00,73,00,74,00,65,00,6d,00,52,00,6f,00,6f,00,74,00,25,\***

***00,5c,00,53,00,79,00,73,00,74,00,65,00,6d,00,33,00,32,00,5c,00,72,00,75,00,\***

***6e,00,64,00,6c,00,6c,00,33,00,32,00,2e,00,65,00,78,00,65,00,20,00,22,00,25,\***

***00,50,00,72,00,6f,00,67,00,72,00,61,00,6d,00,46,00,69,00,6c,00,65,00,73,00,\***

***25,00,5c,00,57,00,69,00,6e,00,64,00,6f,00,77,00,73,00,20,00,50,00,68,00,6f,\***

***00,74,00,6f,00,20,00,56,00,69,00,65,00,77,00,65,00,72,00,5c,00,50,00,68,00,\***

***6f,00,74,00,6f,00,56,00,69,00,65,00,77,00,65,00,72,00,2e,00,64,00,6c,00,6c,\***

***00,22,00,2c,00,20,00,49,00,6d,00,61,00,67,00,65,00,56,00,69,00,65,00,77,00,\***

***5f,00,46,00,75,00,6c,00,6c,00,73,00,63,00,72,00,65,00,65,00,6e,00,20,00,25,\***

***00,31,00,00,00***

***[HKEY\_CLASSES\_ROOT\Applications\photoviewer.dll\shell\open\DropTarget]***

***"Clsid"="{FFE2A43C-56B9-4bf5-9A79-CC6D4285608A}"***

***[HKEY\_CLASSES\_ROOT\Applications\photoviewer.dll\shell\print]***

***[HKEY\_CLASSES\_ROOT\Applications\photoviewer.dll\shell\print\command]***

***@=hex(2):25,00,53,00,79,00,73,00,74,00,65,00,6d,00,52,00,6f,00,6f,00,74,00,25,\***

***00,5c,00,53,00,79,00,73,00,74,00,65,00,6d,00,33,00,32,00,5c,00,72,00,75,00,\***

***6e,00,64,00,6c,00,6c,00,33,00,32,00,2e,00,65,00,78,00,65,00,20,00,22,00,25,\***

***00,50,00,72,00,6f,00,67,00,72,00,61,00,6d,00,46,00,69,00,6c,00,65,00,73,00,\***

***25,00,5c,00,57,00,69,00,6e,00,64,00,6f,00,77,00,73,00,20,00,50,00,68,00,6f,\***

***00,74,00,6f,00,20,00,56,00,69,00,65,00,77,00,65,00,72,00,5c,00,50,00,68,00,\***

***6f,00,74,00,6f,00,56,00,69,00,65,00,77,00,65,00,72,00,2e,00,64,00,6c,00,6c,\***

***00,22,00,2c,00,20,00,49,00,6d,00,61,00,67,00,65,00,56,00,69,00,65,00,77,00,\***

***5f,00,46,00,75,00,6c,00,6c,00,73,00,63,00,72,00,65,00,65,00,6e,00,20,00,25,\***

***00,31,00,00,00***

***[HKEY\_CLASSES\_ROOT\Applications\photoviewer.dll\shell\print\DropTarget]***

***"Clsid"="{60fd46de-f830-4894-a628-6fa81bc0190d}"***

1. Double-click on your new REG file to merge it with your Windows Registry.
2. You should now be able to see the Windows Photo Viewer and set it as the default program for various image files.

##### References

<https://www.cnet.com/how-to/how-to-get-windows-photo-viewer-back-in-windows-10/>

#### How to delete files which gives the error of “Folder Access Denied” when deleting?

1. Install the registry utility of “Take ownership” which is located in following folder
2. “Windows, Linux, Android Software and Drivers\Windows Software\System\OS Enhancements\Take Ownership”.
3. Right click on the folder which cannot be deleted and Click the option “Take ownership”.
4. Try to delete the folder again.
5. Watch the following video tutorial

* <https://www.youtube.com/watch?v=xa6Ru7VKgH4>
* <http://www.howtogeek.com/howto/windows-vista/add-take-ownership-to-explorer-right-click-menu-in-vista/>

#### How to install macOS Sierra on VMWare?

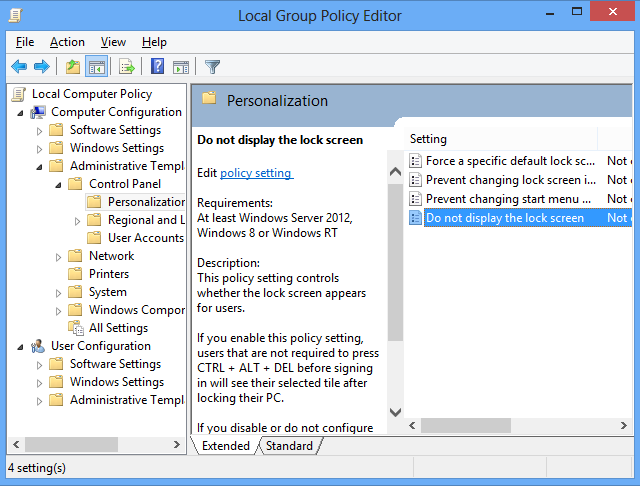
* 
* <https://techsviewer.com/install-macos-sierra-vmware-windows/>

## Operating System

### Windows

#### How to disable the Windows 8 lock screen?

* Hit the Start key, type gpedit.msc, and press Enter. This will open the Local Group Policy Editor.
* Navigate to Computer Configuration > Administrative Templates > Control Panel > Personalization
* Double click “Do not display the lock screen,” and select ‘Enabled’ from the dialog that pops up. Click OK.
* Go ahead and press Win+L and admire your new, minimal lock screen.



* <http://www.extremetech.com/computing/142482-how-to-disable-the-windows-8-lock-screen>

## Programming Languages

### PHP

#### How to debug PHP web applications using PHPStorm?

* <https://www.youtube.com/watch?v=lOIB0ct-dAU>

## Software Tools

### FileZilla

#### How to Fix 'Failed to retrieve directory listing' in Filezilla?

* <https://www.youtube.com/watch?v=XDxefhxe8wA>
* <http://www.tilcode.com/how-to-fix-failed-to-retrieve-directory-listing-in-filezilla-3-10-with-bluehost/>
* <https://forum.filezilla-project.org/viewtopic.php?f=2&t=34842>

### WebStorm

#### How to configure prettier on webstorm?

1. Configure prettier as an external tool
2. Configure prettier as a file watcher to automatically format when saving

* D:\education\video-tutorials\Nuwan's Technical Recordings\IDEs\WebStorm\ Configure Prettier on Webstorm.mp4
* <https://prettier.io/docs/en/webstorm.html>
* https://medium.com/@jm90mm/adding-prettier-to-webstorm-a218eeec04d2

### Source Tree

#### Source Tree for Windows does not start

1. Delete folder ‘C:\Users\nuwanc\AppData\Local\Atlassian\SourceTree.exe\_Url\_y4l4uc0q4yob55igmhwukw3uqdkk2jgp’
2. Restart ‘Source Tree’

* <https://community.atlassian.com/t5/Sourcetree-questions/Source-Tree-for-Windows-does-not-start/qaq-p/26688>

#### Refresh SourceTree immediately to show remote branches

1. Press `ALT+SHIFT+R` to refresh SourceTree immediately
2. Remote branches would show up in SourceTree

* https://stackoverflow.com/a/33098862

# Redux Saga

## What is it?

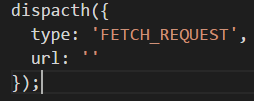
### What is it?

* It is a Redux middleware for handling side effects .
* All the handling inside reducers is synchronous and pure. But in real applications we need also to do things that are asynchronous (may not complete immediately like an AJAX request) and impure (change the state of the outside world, like saving to a database). In functional programming (FP) jargon we commonly refer to those things as side effects . The redux-saga middleware isolates all side effects into software artifacts called sagas so that side effects can be managed.
* The meaning of term saga is actually a process manager basically: "a process that receive events, and may emit new events (sync or async), aiming to orchestrate complex workflows inside your application"
* It's sufficient to know that a saga is a piece of code which runs in the background, watch for dispatched actions, may perform some async calls (or synchronous impure calls like browser storage) and can dispatch other actions to the store.

## Basic Saga sample

### Calling from the UI

* First, UI Components never invoke the tasks themselves, instead they always dispatch plain object actions to notify that something happened in the UI.



* Everything else must be encapsulated inside sagas. To perform the task that will perform the actual fetch, you must create a saga that will watch for the dispatched action FETCH\_REQUEST and *fork* the task whenever we get the desired action:



# JavaScript

## ES2015

### General

#### What are the features introduced in ES2015?

1. Arrows and Lexical This
2. Classes
3. Enhanced Object Literals
4. Template Strings
5. Destructuring
6. Default + Rest + Spread
7. Let + Const
8. Iterators + For..Of
9. Generators
10. Unicode
11. Modules
12. Module Loaders
13. Map + Set + WeakMap + WeakSet
14. Proxies
15. Symbols
16. Subclassable Built-ins
17. Math + Number + String + Object APIs
18. Binary and Octal Literals
19. Promises
20. Reflect API
21. Tail Calls

#### What is ‘Arrows and Lexical This’ feature?

* Arrows are a function shorthand using the => syntax.

*// Expression bodies*var odds = evens.map(v => v + 1);  
var nums = evens.map((v, i) => v + i);

*// Statement bodies*nums.forEach(v => {  
 if (v % 5 === 0) fives.push(v);  
});

*// Lexical this*var bob = {  
 \_name: 'Bob',  
 \_friends: [],  
 printFriends() {  
 this.\_friends.forEach(f => console.log(this.\_name + ' knows ' + f));  
 },  
};

*// Lexical arguments*function square() {  
 let example = () => {  
 let numbers = [];  
 for (let number of arguments) {  
 numbers.push(number \* number);  
 }  
  
 return numbers;  
 };  
  
 return example();  
}  
  
square(2, 4, 7.5, 8, 11.5, 21); *// returns: [4, 16, 56.25, 64, 132.25, 441]*

# React

Introduction

Definition

* React is a front-end JavaScript library developed by Facebook in 2011.
* It follows the component-based approach which helps in building reusable UI components.
* It is used for developing complex and interactive web and mobile UI.
* Even though it was open-sourced only in 2015, it has one of the largest communities supporting it.

Features

* It uses the virtual DOM instead of the real DOM.
* It uses server-side rendering.
* It follows unidirectional data flow or data binding. it.

Advantages

* It increases the application’s performance
* It can be conveniently used on the client as well as server side. This enables improved SEO and performance.
* Because of JSX, code’s readability increases. It is also really easy to see the layout, or how components are plugged/combined with each other.
* React is easy to integrate with other frameworks like Meteor, Angular, etc as it is only a view layer.
* Using React, writing UI test cases become extremely easy.

Limitations

* React is just a library, not a full-blown framework
* Its library is very large and takes time to understand
* Coding gets complex as it uses inline templating and JSX

How React works

* React creates a virtual DOM. When state changes in a component it firstly runs a "diffing" algorithm, which identifies what has changed in the virtual DOM.
* The second step is reconciliation, where it updates the DOM with the results of diff.

Virtual DOM

Introduction

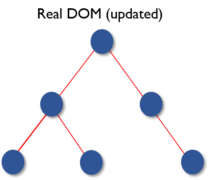
* A virtual DOM is a lightweight JavaScript object which originally is just the copy of the real DOM. It is a node tree that lists the elements, their attributes and content as Objects and their properties.
* React’s render function creates a node tree out of the React components. It then updates this tree in response to the mutations in the data model which is caused by various actions done by the user or by the system.
* This Virtual DOM works in three simple steps.
  1. Whenever any underlying data changes, the entire UI is re-rendered in Virtual DOM representation.



* 1. Then it runs a "diffing" algorithm, which identifies the difference between the previous DOM representation and the new one.



3. Next step is reconciliation which is that the real DOM will be updated with only the things that have actually changed.



React Component Lifecycle

### React Component Lifecycle

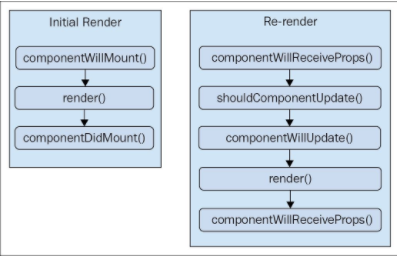
#### Lifecycle methods

* componentWillReceiveProps
* shouldComponentUpdate
* componentWillUpdate
* componentDidMount
* componentWillMount
* componentDidMount

#### Lifecycle methods explanation

* componentWillReceiveProps
  + Only called when the component's properties are updated.
* componentWillMount
  + This is the only lifecycle method that's called when a component is about to be removed
* shouldComponentUpdate
  + Used to determine whether or not the component will render itself when asked to

#### How a component flows through its lifecycle



## Components

### Higher Order Components

#### Higher order function

* A higher order function is a function that takes another function as input, and returns a new function as output.
* This returned function calls the original function in some way. The idea is to compose new behavior out of existing behavior.

#### Higher order React components

* With higher-order React components, you have a function that takes a component as input, and returns a new component as output.
* This is the preferred way to compose new behavior in React applications.
* One obvious use case for a higher-order component is conditional rendering.

### Presentational Components

* Presentational components are concerned with how things look.
* They generally receive data and callbacks exclusively via props.
* They probably only contain a render method and little else logic.
* They do not know how to load or alter the data that they render.
* They are best written as stateless functional components.

### Container Components

* Container components are primarily concerned with how things work.
* They are also often stateful as they serve as data sources.
* They are responsible for providing data and behavior to their children (usually presentational components).
* They call Flux actions and provide these as callbacks to the presentational components.

### Class Components

* Class components allows you to use additional features such as local state and lifecycle hooks.
* Also, to enable your component to have direct access to your store and thus holds state.

### Functional Components

* When your component just receives props and renders them to the page, this is a 'stateless component', for which a pure function can be used.
* These are also called dumb components or presentational components.

### Controlled Components

* In HTML, form elements such as input html element, textarea html element, and select html element typically maintain their own state and update it based on user input. When a user submits a form the values from the aforementioned elements are sent with the form. With React it works differently. The component containing the form will keep track of the value of the input in its state and will re-render the component each time the callback function e.g. onChange is fired as the state will be updated. An input form element whose value is controlled by React in this way is called a "controlled component".

## State

* The state is a data structure that starts with a default value when a Component mounts.
* It may be mutated across time, mostly as a result of user events.

## Props

* Props (short for properties) are a Component's configuration.
* They are received from above and immutable as far as the Component receiving them is concerned.
* A Component cannot change its props, but it is responsible for putting together the props of its child Components.
* Props do not have to just be data - callback functions may be passed in as props.

## JSX

* JSX is a shorthand for JavaScript XML.
* This is a type of file used by React which utilizes the expressiveness of JavaScript along with HTML like template syntax. This makes the HTML file really easy to understand.
* You can embed any JavaScript expression in JSX by wrapping it in curly braces.
* JSX produces React "elements".
* After compilation, JSX expressions become regular JavaScript objects.
* You can use JSX inside of if statements and for loops, assign it to variables, accept it as arguments, and return it from functions.

## Ref

* Refs are used to get reference to a DOM node or an instance of a component in React.
* Good examples of when to use refs are for managing focus/text selection, triggering imperative animations, or integrating with third-party DOM libraries.
* You should avoid using string refs and inline ref callbacks. Callback refs are advised by React.

## Events

## Forms

## Routing

### Declaring routes

### Handling route parameters

### Using link components

### Lazy routing

## Redux

### Introduction

* The basic idea of redux is that the entire application state is kept in a single store.
* The store is simply a Javascript object.
* The only way to change the state is by firing actions from your application and then writing reducers for these actions that modify the state.
* The entire state transition is kept inside reducers and should not have any side-effects.

### Store

* The store is a Javascript object that holds application state.
* Along with this it also has the following responsibilities:
  + Allows access to state via getState()
  + Allows state to be updated via dispatch(action)
  + Registers listeners via subscribe(listener)
  + Handles unregistering of listeners via the function returned by subscribe(listener).

### Action

* Actions are plain javascript objects.
* They must have a type indicating the type of action being performed.
* In essence, actions are payloads of information that send data from your application to your store.

### Reducer

* A reducer is simply a pure function that takes the previous state and an action, and returns the next state.

# Next.js

## Documentation

<https://nextjs.org/docs/>

## Using Shared Components

Adding a common Header component to the app inside a Layout - <https://nextjs.org/learn/basics/using-shared-components>

# React Native

## Samples

### Premium React Native templates

* ReactAZ - <https://reactaz.com/>

### Free React Native samples

* F8 App 2017 - <https://github.com/fbsamples/f8app>
* https://react.rocks/tag/ReactNative

## Common issues

### Running project

#### How to fix ‘Failed to crunch file’ error?

1. Following error comes when running the project using ‘react-native run-android’

:app:mergeDebugResources

Error: Failed to crunch file D:\projects\reference-applications\react-native-reference-application\authorization\gh\_NerdySaiyan\_redux-oidc-example-react-native\ReactNativeApp\android\app\build\intermediates\exploded-aar\com.android.support\appcompat-v7\23.0.1\res\drawable-xhdpi\abc\_list\_selector\_disabled\_holo\_light.9.png into D:\projects\reference-applications\react-native-reference-application\authorization\gh\_NerdySaiyan\_redux-oidc-example-react-native\ReactNativeApp\android\app\build\intermediates\res\merged\debug\drawable-xhdpi\abc\_list\_selector\_disabled\_holo\_light.9.png

:app:mergeDebugResources FAILED

FAILURE: Build failed with an exception.

1. It was due to long path of project directory. Move the project to a directory with shorter path.

* Video - D:\education\video-tutorials\Nuwan's Technical Recordings\React Native\Common Issues\Running project
* <https://stackoverflow.com/questions/30764604/execution-failed-for-task-appmergedebugresources-crunching-cruncher-png-fa>

### Polyfills

#### How to fix ‘Symbol is not a function’ error?

1. 
2. When debugging the app in Android, it does not give an error. Above error occurs when running the app without the debug mode.
3. Fix for this is to use a polyfill.
4. Import the polyfill as follows in ‘App.android.js’ file’ after doing redux imports

import 'core-js/es6/symbol';  
import 'core-js/fn/symbol/iterator';

1. Import following lines as the very first lines in ‘App.android.js’ file’

* <https://github.com/facebook/immutable-js/issues/1305#issuecomment-341308925>
* <https://github.com/facebookincubator/create-react-app/issues/2765#issuecomment-314526097>
* <https://github.com/facebook/react-native/issues/4676#issuecomment-266740462>
* <http://babeljs.io/docs/usage/polyfill/>
* <http://facebook.github.io/react-native/docs/javascript-environment.html#content>
* <https://medium.com/@housecor/babel-6-cheat-sheet-7344f7936f2d>

1. Make sure to do the polyfill imports after doing redux imports. Otherwise following error would occur.
2. 
3. So import the polyfill after doing redux imports in ‘App.android.js’ file.

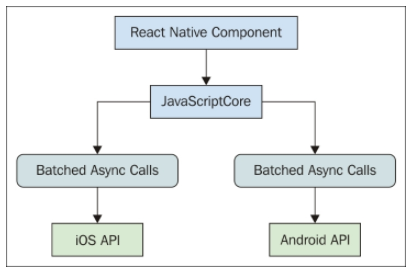
import { Provider } from 'react-redux';  
import store from './Redux/index';  
import 'core-js/es6/symbol';  
import 'core-js/fn/symbol/iterator';

* http://discuss.babeljs.io/t/babel-polyfill-react-redux-and-old-browsers/1075

## What is it?

### What is it?

* React Native uses a technique that makes asynchronous calls to the underlying mobile OS, which calls the native widget APIs.
* There's a JavaScript engine, and the React API is mostly the same as React for the Web. The difference is mostly with the target; instead of a DOM, there are asynchronous API calls.
* The concept is visualized here.



* The same React library that's used on the Web is used by React Native and runs in JavaScriptCore.
* Messages that are sent to native platform APIs are asynchronous and batched for performance purposes.
* React Native ships with components implemented for mobile platforms, instead of components that are HTML elements.
* As the mobile platform is updated, you want the components of your app to stay updated too. This isn't a problem with React Native because they're using actual components from the platform.

## Kickstarting React Native Projects

### Using the React Native command-line tool

1. Install the *react-native* command onto your system



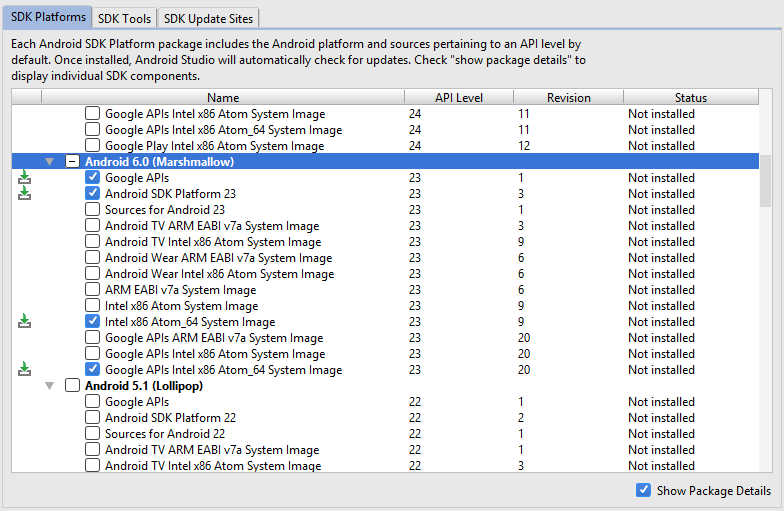
### Android development environment

#### 1. Install Android Studio

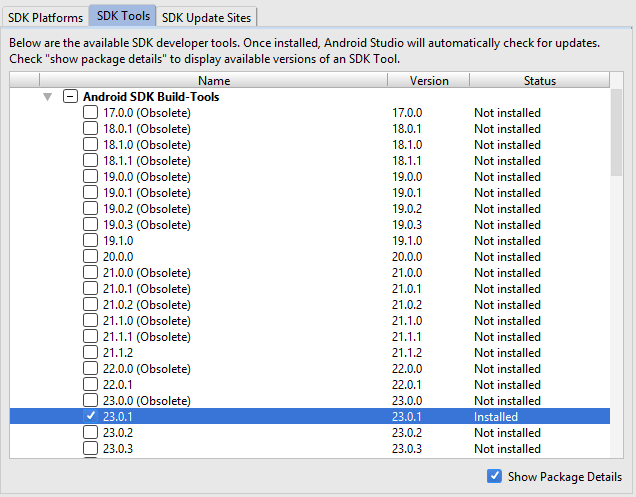
* Download and install Android Studio.
* Choose a "Custom" setup when prompted to select an installation type. Make sure the boxes next to all of the following are checked
  + Android SDK
  + Android SDK Platform
  + Performance (Intel ® HAXM)
  + Android Virtual Device

#### 2. Install the Android SDK

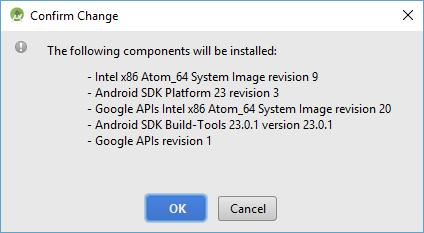
* Building a React Native app with native code, however, requires the Android 6.0 (Marshmallow) SDK in particular.
* The SDK Manager can be found within the Android Studio "Preferences" dialog, underAppearance & Behavior → System Settings → Android SDK.
* Select the "SDK Platforms" tab from within the SDK Manager, then check the box next to "Show Package Details" in the bottom right corner. Look for and expand the Android 6.0 (Marshmallow) entry, then make sure the following items are all checked:
  + Google APIs
  + Android SDK Platform 23
  + Intel x86 Atom\_64 System Image
  + Google APIs Intel x86 Atom\_64 System Image



* Next, select the "SDK Tools" tab and check the box next to "Show Package Details" here as well. Look for and expand the "Android SDK Build-Tools" entry, then make sure that 23.0.1 is selected.

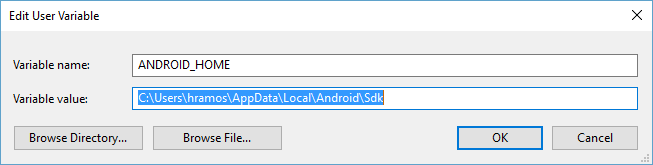


* Finally, click "Apply" to download and install the Android SDK and related build tools.



#### 3. Configure the ANDROID\_HOME environment variable

* Open the System pane under System and Security in the Control Panel, then click onChange settings.... Open the Advanced tab and click on Environment Variables.... Click on New... to create a new ANDROID\_HOME system variable that points to the path to your Android SDK:



* Add following entries to the “Path” in envrironment variables

*%ANDROID\_HOME%\tools*

*%ANDROID\_HOME%\platform-tools*

* You can find the actual location of the SDK in the Android Studio "Preferences" dialog, underAppearance & Behavior → System Settings → Android SDK.
* Open a new Command Prompt window to ensure the new environment variable is loaded before proceeding to the next step.

### Preparing the Android device

* You will need an Android device to run your React Native Android app. This can be either a physical Android device, or more commonly, you can use an Android Virtual Device which allows you to emulate an Android device on your computer.

#### Using a physical device

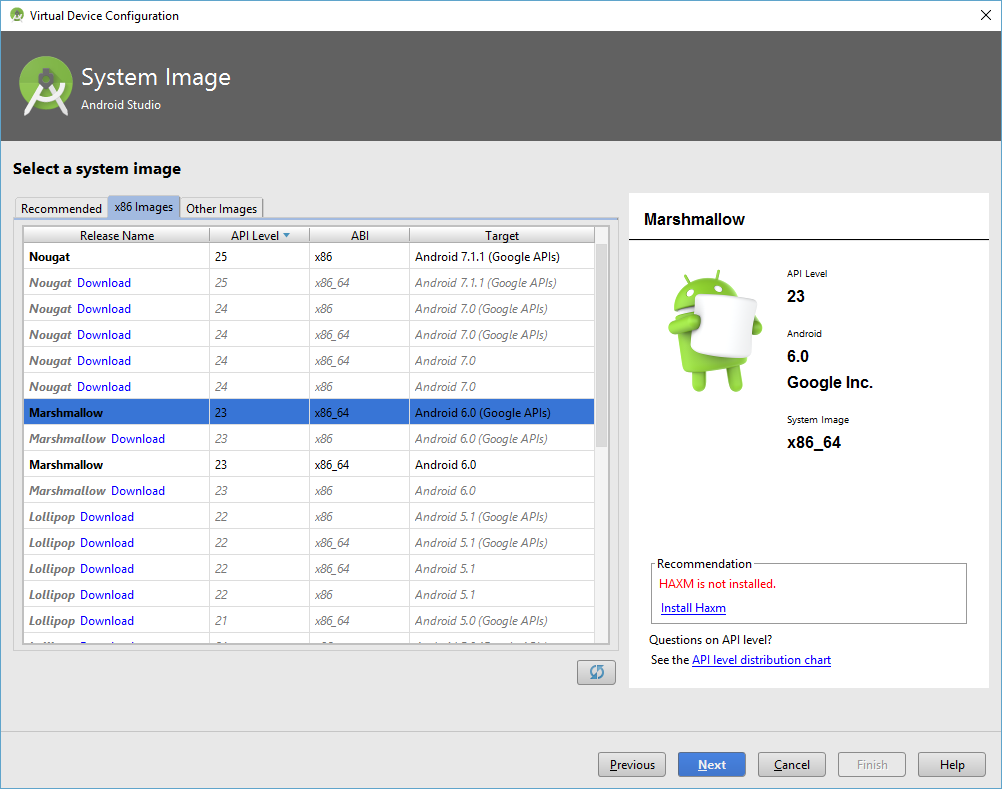
* If you have a physical Android device, you can use it for development in place of an AVD by plugging it in to your computer using a USB cable and following the instructions [here](https://facebook.github.io/react-native/docs/running-on-device.html).

#### Using a virtual device

* You can see the list of available Android Virtual Devices (AVDs) by opening the "AVD Manager" from within Android Studio.
* If you have just installed Android Studio, you will likely need to [create a new AVD](https://developer.android.com/studio/run/managing-avds.html). Select "Create Virtual Device...", then pick any Phone from the list and click "Next".



* Select the "x86 Images" tab, then look for the Marshmallow API Level 23, x86\_64 ABI image with a Android 6.0 (Google APIs) target.



* If you don't have HAXM installed, click on "Install HAXM", then go back to the AVD Manager.

AVD List

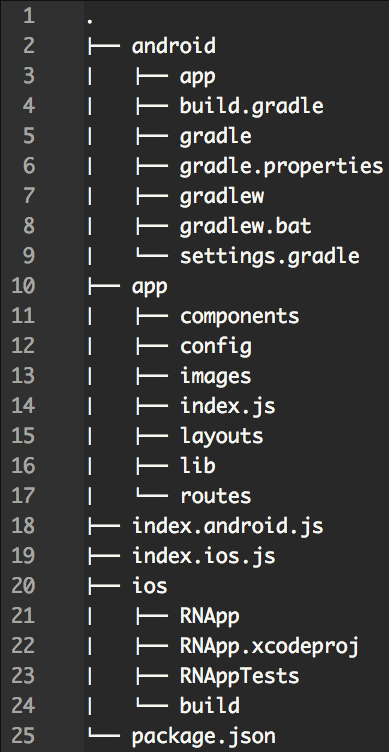
* Click "Next" then "Finish" to create your AVD. At this point you should be able to click on the green triangle button next to your AVD to launch it.

### Creating a new application

1. Create a new *MyProject* directory with several files and directories within.



1. Default folder structure



* + **package.json** - The typical npm package definition. It lists react and react-native as dependencies
  + **index.android.js** - The entry point used by react-native when building Android apps
  + **index.ios.js** - The entry point used by react-native when building iOS apps
  + **android** - The project files used to build Android apps
  + **ios** - The project files used to build iOS apps
  + **node\_modules** - The react-native package and all its dependencies

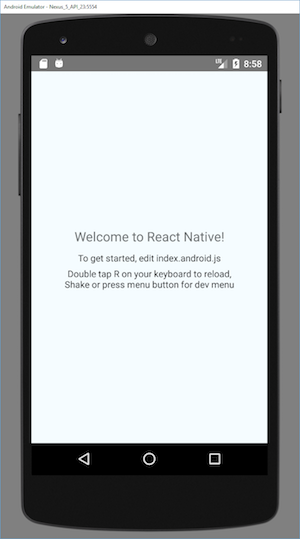
### Running your React Native application

* Run react-native run-android inside your React Native project folder:

cd AwesomeProject

react-native run-android

* If everything is set up correctly, you should see your new app running in your Android emulator shortly.



* react-native run-android is just one way to run your app - you can also run it directly from within Android Studio or [Nuclide](https://nuclide.io/).

### Modifying your app

* Open index.android.js in your text editor of choice and edit some lines.
* Select “Reload” from the Developer Menu (Ctrl + M) to see your changes!

### Debugging your app

* Select “Debug JS Remotely” from the Developer Menu (Ctrl + M) to debug the application

## Components

### Checkboxes/ Radios

#### How to fix the issue of making small checkboxes touched with large fingers?

1. Use ‘hitSlop’ property of ‘TouchableWithoutFeedback’
2. Define how far the touch can start away from the button using ‘hitSlop’ property.

<CheckBox  
 onPress={() => {  
 this.props.OnItem\_checkBoxPress();  
 }}  
 hitSlop={{ top: 25, left: 25, bottom: 25, right: 25 }}  
/>

* [https://facebook.github.io/react-native/docs/touchablewithoutfeedback.html#hitslop](https://facebook.github.io/react-native/docs/touchablewithoutfeedback.html%23hitslop)

## Navigation

### General

#### What are the popular libraries available for navigation in React native?

1. React Navigation (https://reactnavigation.org/)
2. Wix – react native navigation (https://wix.github.io/react-native-navigation/)

### React Navigation

#### What are the built-in navigators available in ‘React Navigation’?

1. StackNavigator
2. TabNavigator
3. DrawerNavigator

#### What is a ‘StackNavigator’?

* Renders one screen at a time and provides transitions between screens. When a new screen is opened it is placed on top of the stack.

#### What is a ‘TabNavigator’?

* Renders a tab bar that lets the user switch between several screens

#### What is a ‘DrawerNavigator’?

* Provides a drawer that slides in from the left of the screen

#### Give an example of a ‘StackNavigator’?

import React from 'react';  
import {  
 AppRegistry,  
 Text,  
} from 'react-native';  
import { StackNavigator } from 'react-navigation';  
  
class HomeScreen extends React.Component {  
 static *navigationOptions* = {  
 title: 'Welcome',  
 };  
 render() {  
 return <Text>**Hello, Navigation!**</Text>;  
 }  
}  
  
export const SimpleApp = StackNavigator({  
 Home: { screen: HomeScreen },  
});  
  
AppRegistry.registerComponent('SimpleApp', () => SimpleApp);

## Boilerplate

### How to create a boilerplate using React Native boilerplate with Ignite CLI

## Unit tests for redux thunk action creators

* <https://github.com/facebook/jest/issues/2071>
* <https://github.com/jefflau/jest-fetch-mock>
* <https://github.com/jefflau/jest-fetch-mock/issues/13>

### Configuring jest-fetch-mock

*// setupJest.js*import 'isomorphic-fetch';  
  
global.fetch = require('jest-fetch-mock');

*//package.json*"devDependencies": {  
 "isomorphic-fetch": "^2.2.1",  
 "jest-fetch-mock": "^1.2.1"  
}

*//package.json*"jest": {  
 "setupFiles": [  
 "./jest/setup.js"  
 ]  
}

*// actions.test.js  
// Writing the test*import React from 'react';  
import "react-native";  
import configureMockStore from 'redux-mock-store'  
import thunk from 'redux-thunk'  
import \* as DiscoverActions from '../../../src/Redux/Modules/Discover/actions';  
  
const middlewares = [thunk];  
const mockStore = configureMockStore(middlewares);  
  
describe('DiscoverBuckets ->', () => {  
 it('loadDiscoverBuckets API call', () => {  
 const store = mockStore({ });  
 const auth = { token: "134644644493544" };  
 const successCallback = () => {};  
 const errorCallback = () => {};  
 const responseObj = {  
 buckets: [],  
 allocatedAmount: 1000  
 };  
  
 fetch.mockResponse(*JSON*.stringify(responseObj));  
 return store.dispatch(DiscoverActions.loadDiscoverBuckets(auth, errorCallback));  
 })  
});

## How to mock a react native library function

Following is the application code.

import { Navigation } from 'react-native-navigation';  
  
Navigation.startSingleScreenApp({  
 screen: {  
 screen: 'BlueSquirrel.Authentication',  
 navigatorStyle: {  
 navBarHidden: true,  
 statusBarColor: Colors.StatusbarColor  
 }  
 },  
 passProps: {  
 isTokenExpired: true,  
 errorMessage: "Token expired. Please relogin again."  
 },  
 animationType: 'none'  
});

Following is how to mock the above react-native-navigation library’s Navigation object.

*//~/jest/setup.js  
// Mock for { Navigation } in react-native-navigation*jest.mock('react-native-navigation', () => {  
 return {  
 Navigation: {  
 startSingleScreenApp: ()=>{}  
 }  
 }  
});

## Libraries

### react-native-navigation

#### Adding badges for top tabs

Currently adding badges for the top tabs is **NOT** supported. Only the bottom tabs which were created using Navigation.startTabBasedApp() are eligible to add badges. (<https://github.com/wix/react-native-navigation/issues/1435>). There is a pull request that handles adding badges to the title bar buttons (not the top tab buttons) which is still not merged. (<https://github.com/wix/react-native-navigation/pull/619/>).

### react-native-fbsdk

#### How to fix the issue of ‘Error: No resource found that matches the given name: attr 'android:keyboardNavigationCluster'

1. Following error comes when running the Android application in react-native-fbsdk

D:\projects\office\mitrai\bluesquirrel\blue-squirrel\BlueSquirrelMobile\node\_modules\react-native-fbsdk\android\build\intermediates\res\merged\release\values-v26\values-v26.xml:15: error: Error: No resource found that matches the given name: attr 'android:keyboardNavigationCluster'.

1. Compile ‘react-native-fbsdk’ as follows in ‘android/app/build.gradle’ file.

compile project(':react-native-fbsdk') {  
 configurations.all {  
 resolutionStrategy {  
 force 'com.facebook.android:facebook-android-sdk:4.28.0'  
 }  
 }  
}

* Video - D:\education\video-tutorials\Nuwan's Technical Recordings\React Native\Libraries\react-native-fbsdk\ Fix the issue of Error No resource found that matches the given name attr androidkeyboardNavigationCluster in react-native-fbsdk
* <https://stackoverflow.com/questions/44190829/facebook-sdk-android-error-building/44205756#44205756>
* <https://developers.facebook.com/docs/android/change-log-4x>
* <https://stackoverflow.com/questions/44202072/error-retrieving-parent-for-item-no-resource-found-that-matches-the-given-name>
* <https://www.codesd.com/item/error-retrieving-parent-for-item-no-resource-found-matching-the-given-name-android-textappearance-material-widget-button-colored.html>

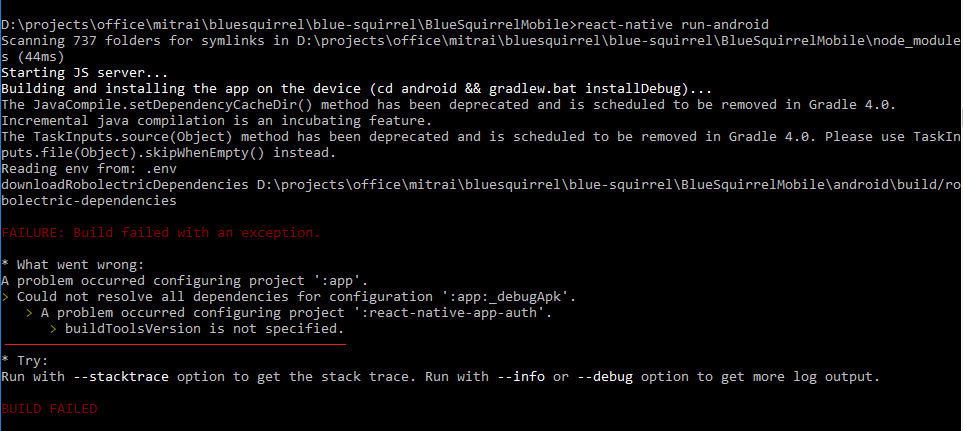
### react-native-appp-auth

#### How to run react-native-app-auth library with gradle v2

1. Add *‘google’* repository to the buildscript repositories section in ‘android/build.gradle’ file in your project.

buildscript {  
 repositories {  
 maven {  
 url "https://maven.google.com"  
 }  
 }  
 dependencies {  
 ...  
 }  
}

1. Following error would occur stating that the *‘buildToolsVersion is not specified’*



1. Add ‘buildToolsVersion’ and ‘targetSdkVersion’ in ‘node\_modules/react-native-app-auth/android/build.gradle’ file as follows.

android {  
 compileSdkVersion 23  
 buildToolsVersion "23.0.1"  
  
 defaultConfig {  
 minSdkVersion 16  
 targetSdkVersion 23  
 versionCode 1  
 versionName "1.0"  
 manifestPlaceholders = [  
 'appAuthRedirectScheme': 'please.override.me'  
 ]  
 }  
 lintOptions {  
 abortOnError false  
 }  
}

## Android app permissions in React native

In React Native, even though the permissions like SYSTEM\_ALERT\_WINDOW and READ\_PHONE\_STATE are removed in AndroidManifest.xml they seems not to be removed in the release build. This is because the react-native library itself add these permissions in it’s internal AndroidManifest.xml file for purposes like running the app in dev mode. So to remove these permissions completely in the App in release build, AndroidManifest.xml in the app has to add a special attribute with declaring this permission to indicate to remove the permissions in the internal libraries too. The syntax is follows.

<uses-permission  
 android:name="android.permission.READ\_PHONE\_STATE"  
 tools:node="remove"/>  
<uses-permission  
 android:name="android.permission.SYSTEM\_ALERT\_WINDOW"  
 tools:node="remove"/>

THIS SHOULD NOT BE PUT WHILE DEBUGGING. Othewise the application would resist to come to display in debug mode without the SYSTEM\_ALERT\_WINDOW permission. So comment these lines while debugging and add them when creating a release build.

Reference –

<https://github.com/facebook/react-native/issues/5886>

https://medium.com/@applification/fixing-react-native-android-permissions-9e78996e9865

In order to find what permissions the app has without putting it to the plastore, we could copy a release apk to the phone and try to install it from the phone as opposed to install using adb. It would display a list of permissions the app requires. Another way to find permissions the app has would be to install an application like ‘Permissions Explorer’ (<https://play.google.com/store/apps/details?id=com.carlocriniti.android.permission_explorer>) which would be capable of listing the permissions of apps which are installed in the computer.

## Crash reporting in React Native

### Resources

<https://github.com/facebook/react-native/issues/5378>

Following can be used for crash reporting in React Native

* react-native-fabric
  + <https://github.com/corymsmith/react-native-fabric>
  + https://github.com/mikelambert/react-native-fabric-crashlytics - what traps errors and sends them in to crashlytics
* bugsnag
  + <https://docs.bugsnag.com/platforms/react-native/>
* Sentry
  + https://github.com/getsentry/react-native-sentry

### Pricing

Crashlytics – Free

Bugsnag – Partly free

Sentry – Partly free (<https://sentry.io/pricing/>)

### Crashlytics

#### Integrate Crashlytics to the project

Resources

<https://fabric.io/kits/android/crashlytics/install>

<https://github.com/corymsmith/react-native-fabric#no_android_studio>

https://www.spritle.com/blogs/2016/02/15/integrate-fabricio-crashlytics-to-a-react-native-android-app/

<https://streetsmartdev.com/react-native-analytics-fabric/>

https://blog.mojotech.com/fabric-integration-for-react-native-on-android/

https://medium.com/komenco/beta-testing-your-react-native-android-application-with-crashlytics-483c7e66a423

* android /build.gradle

+ buildscript {

+ repositories {

+ maven { url 'https://maven.fabric.io/public' }

+ }

+ dependencies {

+ // The Fabric Gradle plugin uses an open ended version to react

+ // quickly to Android tooling updates

+ classpath 'io.fabric.tools:gradle:1.+'

+ }

+ }

* android/app/build.gradle

apply plugin: "com.android.application"

+ apply plugin: 'io.fabric'

[...]

dependencies {

[...]

+ compile('com.crashlytics.sdk.android:crashlytics:2.7.0@aar') {

+ transitive = true;

+ }

}

* android/app/src/main/AndroidManifest.xml

<manifest xmlns:android="http://schemas.android.com/apk/res/android"

[...]

<application

[...]

+ <meta-data

+ android:name="io.fabric.ApiKey"

+ android:value=[YOUR API KEY]

+ />

</application>

+ <uses-permission android:name="android.permission.INTERNET" />

</manifest>

* android/app/src/main/java/com/bluesquirrel/MainApplication.java

+ import com.crashlytics.android.Crashlytics;

+ import io.fabric.sdk.android.Fabric;

public class MainApplication extends Application implements ReactApplication {

+ @Override

+ public void onCreate() {

+ super.onCreate();

+ Fabric.with(this, new Crashlytics());

+ }

[...]

}

#### Generating source maps for the React native application

Resources

<https://docs.bugsnag.com/platforms/react-native/showing-full-stacktraces/#generating-source-maps>

<https://github.com/facebook/react-native/issues/7393>

<https://github.com/corymsmith/react-native-fabric/issues/2>

<https://stackoverflow.com/questions/34715106/how-to-add-sourcemap-in-react-native-for-production/34733906#34733906>

react-native bundle --platform android --dev false --entry-file index.android.js --bundle-output android-release.bundle --sourcemap-output android-release.bundle.map

#### Mapping minified error with generated source map

Resources

https://github.com/SoftwareMansion/stack-beautifier

Steps

1. Install stack-beautifier globally

npm install -g stack-beautifier

1. Save the minified stack trace from ‘fabric dashboard’ in a file named ‘mytrace.txt’
2. Save the generated source map in a file named ‘app.js.map’
3. Run the following command

stack-beautifier app.js.map -t mytrace.txt

## App store submission

### Replacing Your App with a New Version

https://developer.apple.com/library/content/documentation/LanguagesUtilities/Conceptual/iTunesConnect\_Guide/Chapters/ReplacingYourAppWithANewVersion.html

#### Creating the Next App Version in iTunes Connect

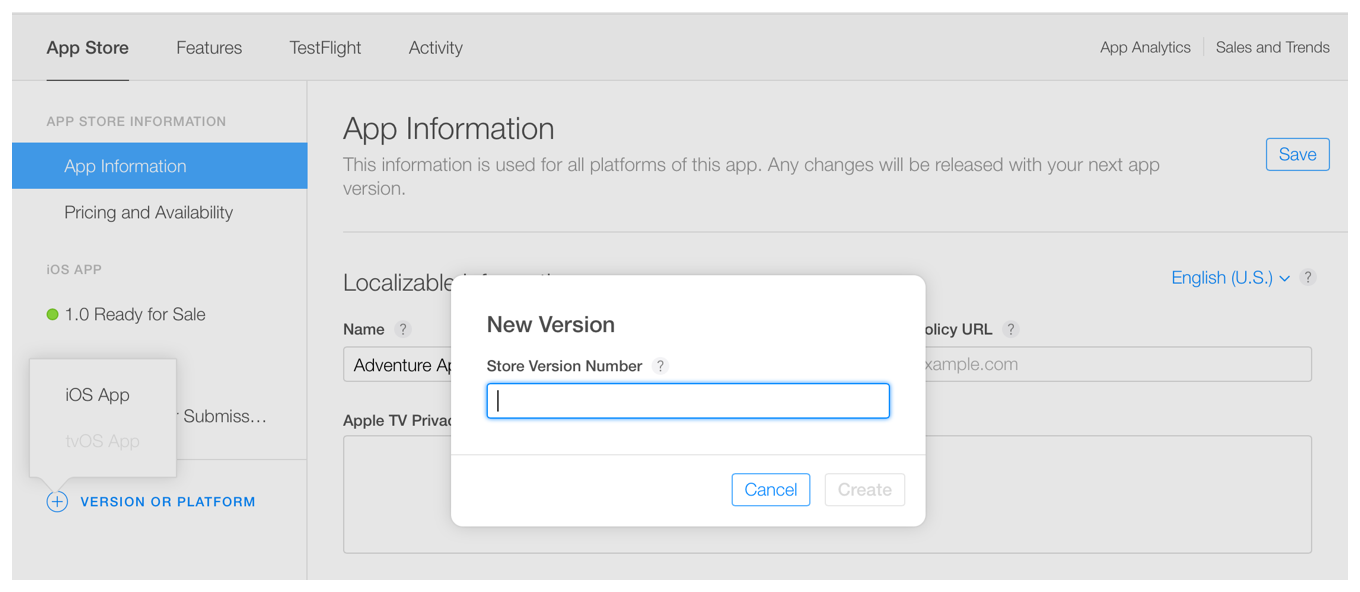
* You can add a new version of an app when the current version’s status is Ready for Sale or Developer Removed from Sale.
* Only the team agent or users with an Admin, Technical, or App Manager role can add a new app version.

##### To add an updated version of an app

1. As an iTunes Connect user with the appropriate role, sign in to iTunes Connect.
2. On the iTunes Connect homepage, click My Apps.



1. Select the app. If you don’t find the app in the list, see Searching for Apps.
2. Click + Version or Platform from the left pane.



* You will only be able to add a new version for a new platform or a platform that has a Ready for Sale or Developer Removed from Sale version. If a platform version has not completed the app review process—it has a status such as Prepare For Submission or Waiting For Review—you can replace the build in the existing version instead of adding a new version.

1. In the New Version dialog, enter the new version number, and click Create. (This will be the version number shown in the App Store.)

A new version for that platform is added with existing metadata from the current version automatically transferred.

1. Review and update the information to describe the new version.

* What’s New in This Version. (Required for each new version.)
* App Preview and Screenshots; see App Preview Specifications and Screenshot Properties.
* Description
* Keywords
* Support URL
* Marketing URL

General App Information (see Platform Version Information), which consists of platform-specific metadata that applies to all localizations, including:

* App icon
* Version number
* Rating
* Copyright
* Trade Representative Contact Information
* Routing app coverage file

Make sure to select a build, as explained in Choosing a Build.

1. Fix any errors indicated for the new version.

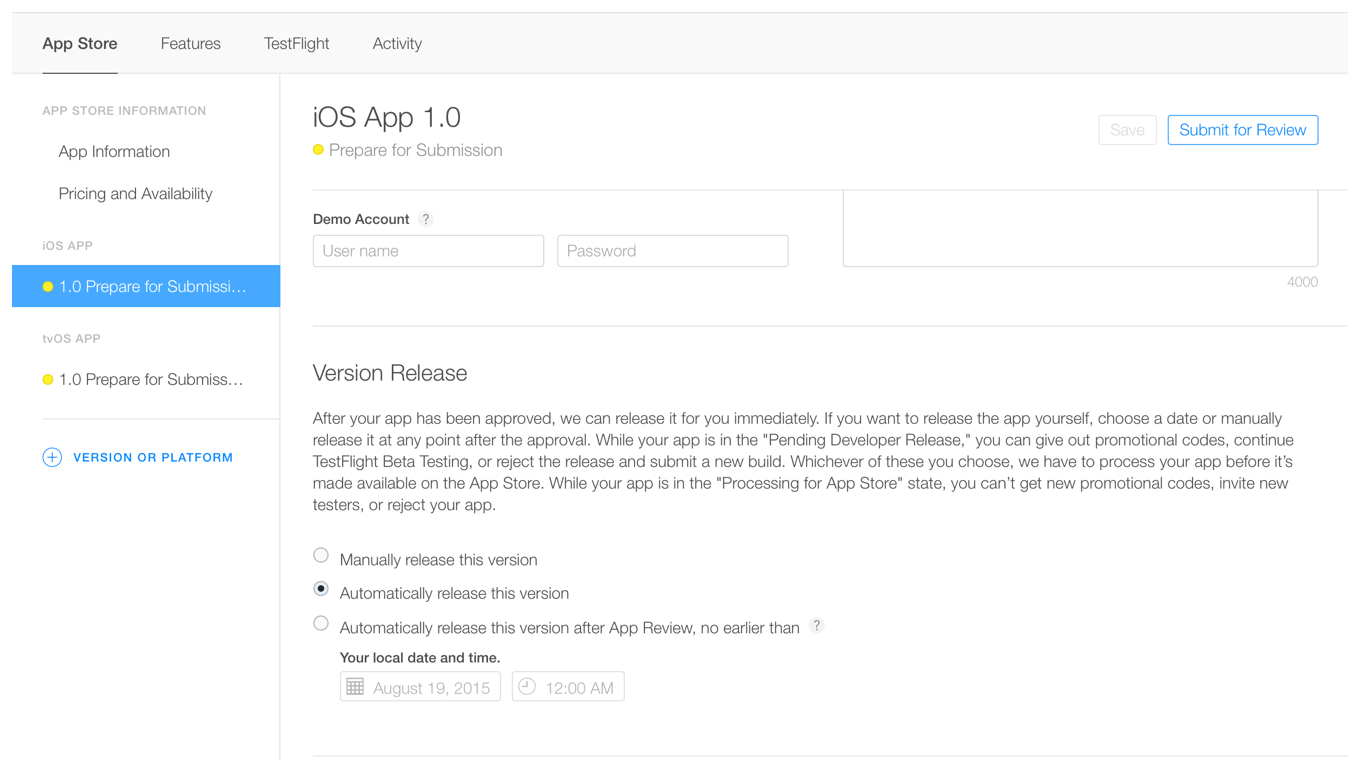
When all errors are resolved, and a build has been added to the new version, the Submit for Review button will be visible.

##### Uploading the New Build

* After you’ve prepared your iTunes Connect record for a new app version, you’re ready to upload the updated build for the new version
* After the build is uploaded, the App Details page shows the version currently for sale with the status Ready for Sale and the version just added with the status Prepare for Submission. The page will show at most two app versions for each platform.

##### Specifying When App Versions Are Released

* You can specify when you want to release an app version to the store by using iTunes Connect version release control.
* For each platform version, you are shown the three options for version release control: manual; automatic; or automatic, no earlier than. The options appear in the Version Release section at the bottom of the page.



# Angular 2

## Architecture

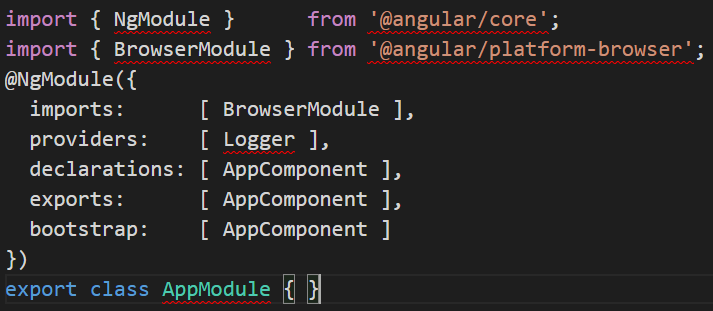


## Main building blocks of an Angular application

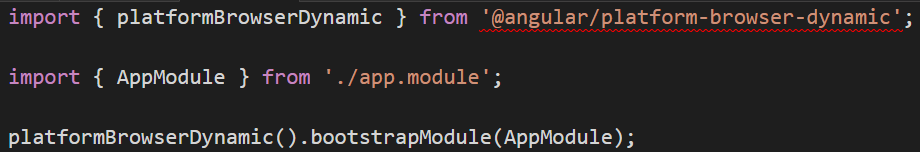
1. Modules
2. Components
3. Templates
4. Metadata
5. Data binding
6. Directives
7. Services
8. Dependency injection

## Modules

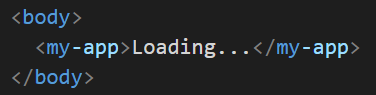
* Simple root module (src/app/app.module.ts)
* **imports** — the BrowserModule that this and every application needs to run in a browser.
* **declarations** — the application's lone component.
* **bootstrap** — the root component that Angular creates and inserts into the index.html host web page.



* Launching an application by bootstrapping its root module (src/main.ts)



* The AppComponent selector (index.html)



## Components

## Angular Forms

## Types of Angular Forms

1. Template driven forms

--- customer.component.html ---

<form (ngSubmit)="save()" #signupForm="ngForm">

<fieldset>

<div [ngClass]="{'has-error': firstNameVar.touched && !firstNameVar.valid}">

<label for="firstNameId">First Name</label>

<input id="firstNameId" type="text"

placeholder="First Name (required)"

required

minlength="3"

[(ngModel)]="customer.firstName"

name="firstName"

#firstNameVar="ngModel" />

<span \*ngIf="firstNameVar.touched && firstNameVar.errors">

Please enter your first name.

</span>

</div>

...

<button type="submit">Save</button>

</fieldset>

</form>

1. Reactive Forms

--- customer.component.html ---

<form (ngSubmit)="save()" [formGroup]="signupForm">

<fieldset>

<div [ngClass]="{'has-error': formError.firstName}">

<label for="firstNameId">First Name</label>

<input id="firstNameId" type="text"

placeholder="First Name (required)"

formControlName="firstName"/>

<span \*ngIf="formError.firstName">

Please enter your first name.

</span>

</div>

...

<button type="submit">Save</button>

</fieldset>

</form>

## Feature comparison of Angular Forms

|  |  |
| --- | --- |
| **Template driven forms** | **Reactive Forms** |
| * we write HTML in our template for the form element, each input element, data binding, validation rules using attributes, and validation error messages. * we first import the FormsModule to bring in the appropriate set of directives. This includes ngForm to access the form model Angular generates for us, ngModel for two-way binding, and to access the input element state defined in the generated form model, and ngModelGroup for grouping * When we add a form element to our template, Angular automatically assigns the ngForm directive to that form. | * We define the form model by creating the instances of the FormGroup and FormControl building blocks in our component class. * To use Reactive forms, we first input the ReactiveFormsModule to bring in its appropriate set of directives. This includes formGroup, formControl, formControlName, formGroupName, formArrayName directives. * With the Reactive forms approach, Angular does not create a form model for us, rather we create it ourselves in our component class. * If you need more functionality, more flexibility, or more control of your forms and their associated data, selecting the Reactive forms approach may be the better choice. |

## Essential form classes

|  |  |  |  |
| --- | --- | --- | --- |
| **AbstractControl** | **FormControl** | **FormGroup** | **FormArrray** |
| * The abstract base class for the three concrete form control classes: FormControl, FormGroup, and FormArray. * It provides their common behaviors and properties, some of which are observable. | * Tracks the value and validity status of an individual form control. * It corresponds to an HTML form control such as an input box or selector. | * Tracks the value and validity state of a group of AbstractControl instances. The group's properties include its child controls. * The top-level form in your component is a FormGroup. | * Tracks the value and validity state of a numerically indexed array of AbstractControl instances. |

## Angular Form and inputs available states

1. Value Changed
   * pristine
   * dirty
2. Validity
   * valid
   * errors
3. Visited
   * touched
   * untouched

# RxJS

## Reactive Programming

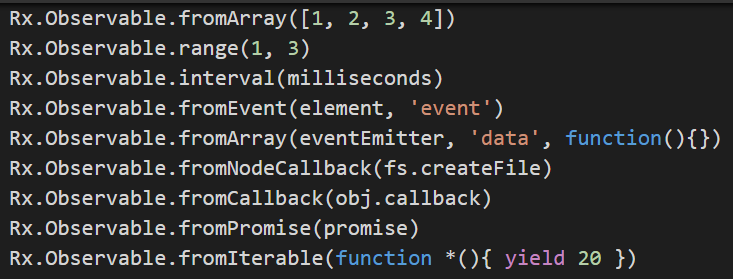
#### What is it?

* Focuses on propagating changes **without** us having to explicitly specify how the propagation happens
* Ideal for processing sequences that can contain any amount of values in any amount of time
* programming with asynchronous data streams.
* you are given an amazing toolbox of functions to combine, create and filter any of those streams.

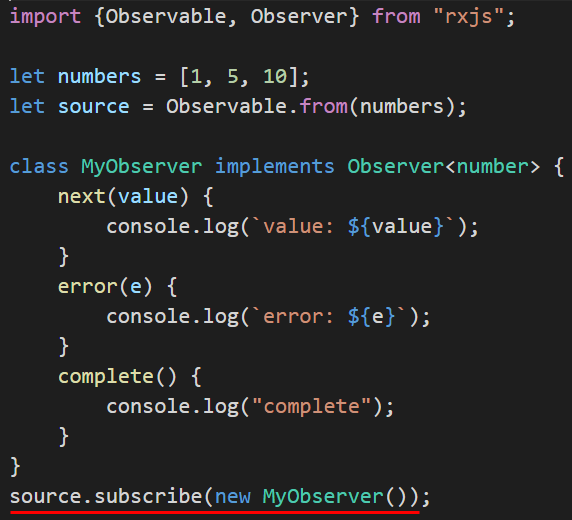
## Asynchronous Background Operations

## Observables

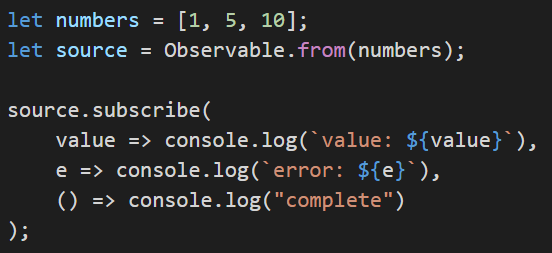
* Creating an observable



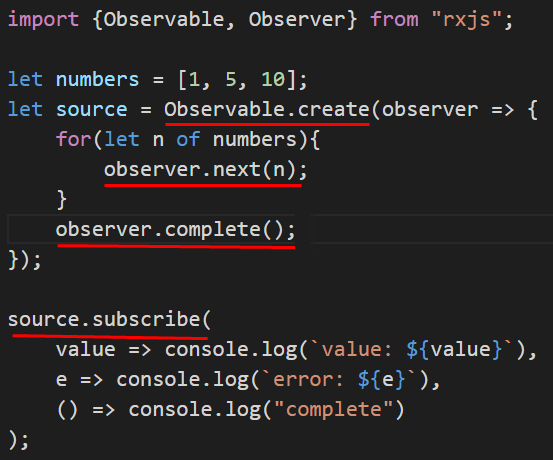
* Creating a simple Observable and a Observer



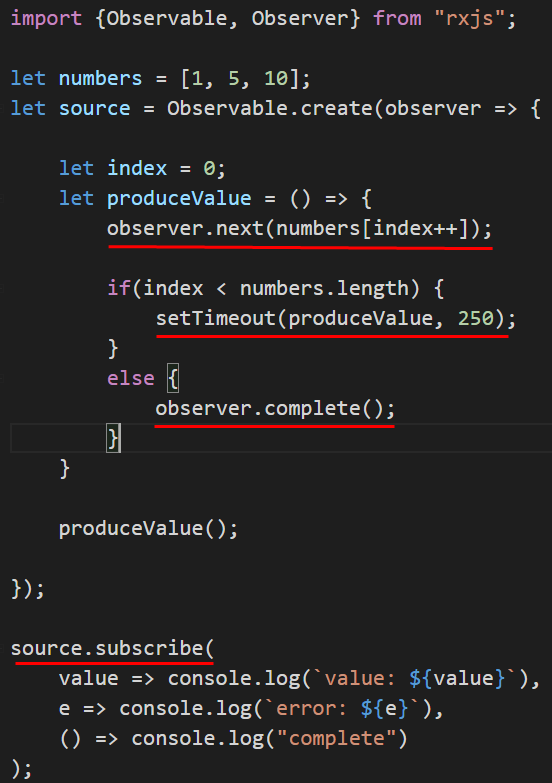
* Looping an array



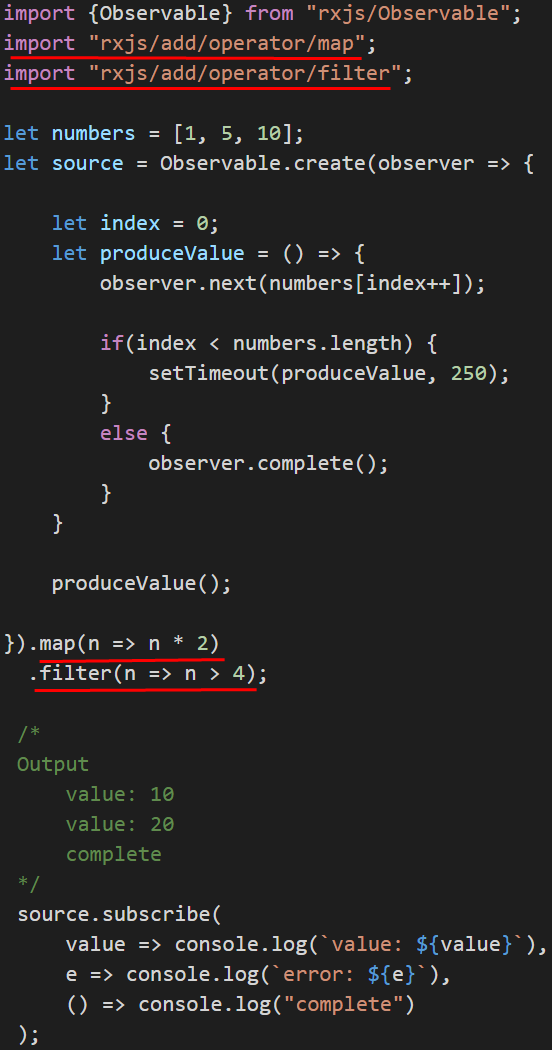
* Using Observable.create



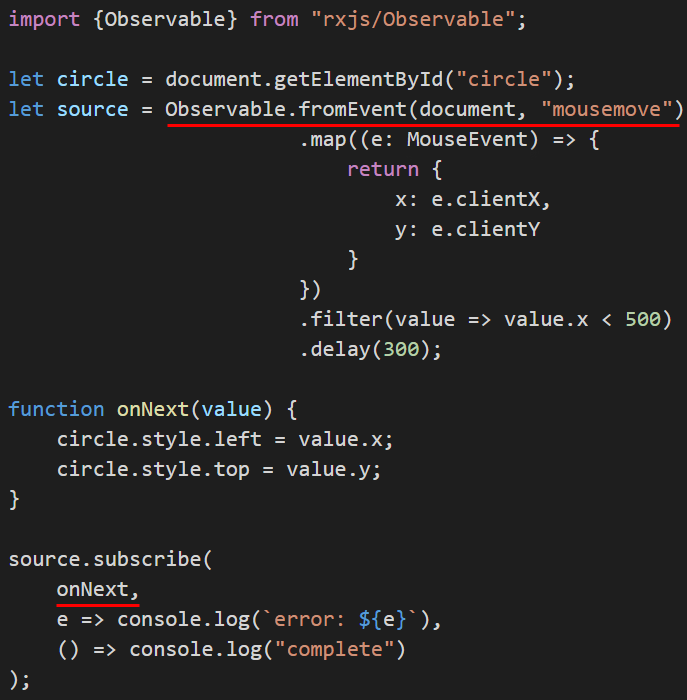
* Asynchronous Observable with setTimeout



* Using RxJS operators with Observables

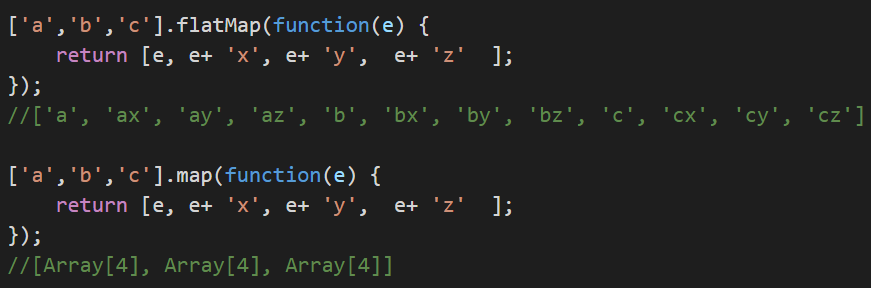


* Processing mouse events



#### flatMap

* Use flatMap when you have an Observable whose results are more Observables.



## Operators

## Observation Operators

## Restriction Operators

## Projection Operators

## Grouping

## Error Handling

## Time-Related Operators

## Window and Joins

## Range

## Generate

## Combination Operators

# iOS

## Common issues

### Building project

#### How to fix ‘No such module 'RestKit' with cocoapods and swift’ error?

1. Error message similar as follow comes when iOS project is opened and when building.

*No such module 'RestKit'*

1. Make sure cocoapods is installed.

sudo gem install cocoapods

1. Navigate to project directory in terminal and execute following command.

pod install

1. Make sure to open the project using the .xcworkspace, not the .xcodeproj

* https://stackoverflow.com/questions/29994331/no-such-module-restkit-with-cocoapods-and-swift

# Git

## Important Git commands

|  |  |  |
| --- | --- | --- |
| Tell Git who you are | Configure the author name and email address to be used with your commits.  Note that Git [strips some characters](http://stackoverflow.com/questions/26159274/is-it-possible-to-have-a-trailing-period-in-user-name-in-git/26219423#26219423) (for example trailing periods) from user.name. | git config --global user.name "Sam Smith"  git config –global user.email sam@example.com |
| Create a new local repository |  | git init |
| Checkout a repository | Create a working copy of a local repository: | git clone /path/to/repository |
| For a remote server, use: | git clone username@host:/path/to/repository |
| Add files | Add one or more files to staging (index): | git add <filename>  git add \* |
|  | Commit changes to head (but not yet to the remote repository): | git commit -m "Commit message" |
| Commit any files you've added with git add, and also commit any files you've changed since then: | git commit -a |
| Push | Send changes to the master branch of your remote repository: | git push origin master |
| Push with credentials | Send changes to the remote repository with explicit credentials | git push https://username:password@github.com/kdnc/angular-reference-application.git --all |
| Status | List the files you've changed and those you still need to add or commit: | git status |
| Connect to a remote repository | If you haven't connected your local repository to a remote server, add the server to be able to push to it: | git remote add origin <server> |
| List all currently configured remote repositories: | git remote -v |
| Branch | Create a new branch and switch to it: | git checkout -b <branchname> |
| Switch from one branch to another: | git checkout <branchname> |
| List all the branches in your repo, and also tell you what branch you're currently in: | git branch |
| Delete the feature branch: | git branch -d <branchname> |
| Push the branch to your remote repository, so others can use it: | git push origin <branchname> |
| Push all branches to your remote repository: | git push --all origin |
| Delete a branch on your remote repository: | git push origin :<branchname> |
| [Update from the remote repository](https://www.atlassian.com/git/tutorials/syncing) | Fetch and merge changes on the remote server to your working directory: | git pull |
| To merge a different branch into your active branch: | git merge <branchname> |
| View all the merge conflicts:  View the conflicts against the base file:  Preview changes, before merging: | git diff  git diff --base <filename>  git diff <sourcebranch> <targetbranch> |
| After you have manually resolved any conflicts, you mark the changed file: | git add <filename> |
| Tags | You can use tagging to mark a significant changeset, such as a release: | git tag 1.0.0 <commitID> |
| CommitId is the leading characters of the changeset ID, up to 10, but must be unique. Get the ID using: | git log |
| Push all tags to remote repository: | git push --tags origin |
| [Undo local changes](https://www.atlassian.com/git/tutorials/undoing-changes) | If you mess up, you can replace the changes in your working tree with the last content in head:  Changes already added to the index, as well as new files, will be kept. | git checkout -- <filename> |
| Instead, to drop all your local changes and commits, fetch the latest history from the server and point your local master branch at it, do this: | git fetch origin  git reset --hard origin/master |
| Search | Search the working directory for foo(): | git grep "foo()" |

* <https://confluence.atlassian.com/bitbucketserver/basic-git-commands-776639767.html>
* <https://orga.cat/posts/most-useful-git-commands>
* <https://stackoverflow.com/a/29776651>

# Docker

## Important Docker commands

* Build a custom image
  + Go to the project root directory where docker file exists
  + Build the custom image

docker build -t nuwan/node-docker-test .

* Pull an image or a repository from a registry

docker pull nginx

* List all the images in my machine

docker images

* Remove docker image

List docker images and find the image id

docker images -a

Remove docker image

docker rmi <image\_id>

* Remove all images

docker rmi $(docker images -a -q)

* Run a custom container

docker run -d -p 49260:8080 nuwan/node-docker-test

* Run a command in a new container in daemon mode

docker run -d -p 49260:80 nginx

* Show containers

docker ps -a

* Stop a container

List docker containers and find the container id

docker ps -a

Stop the container

docker stop <container\_id>

* Remove a container

List docker containers and find the container id

docker ps -a

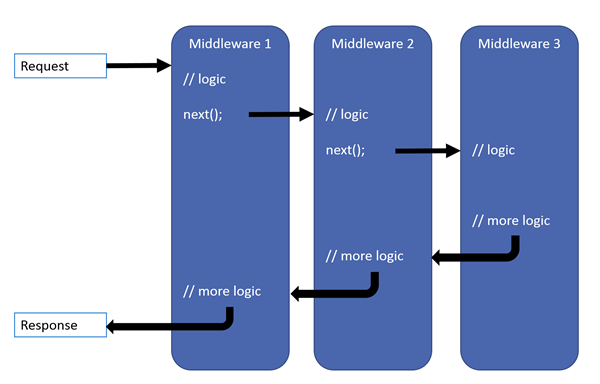
Remove the container

docker rm <container\_id>

# ASP.NET Core

## **Middleware**

* Middleware are software components that are assembled into an application pipeline to handle requests and responses.
* Each component chooses whether to pass the request on to the next component in the pipeline, and can perform certain actions before and after the next component is invoked in the pipeline.
* Request delegates are used to build the request pipeline. The request delegates handle each HTTP request.
* Request delegates are configured using Run, Map, and Use extension methods on the IApplicationBuilder type that is passed into the Configure method in the Startup class.
* An individual request delegate can be specified in-line as an anonymous method, or it can be defined in a reusable class. These reusable classes are middleware, or middleware components.
* Each middleware component in the request pipeline is responsible for invoking the next component in the pipeline, or short-circuiting the chain if appropriate. For example, an authorization middleware might only call the next delegate if the request is authenticated; otherwise it could short-circuit the pipeline and return a “Not Authorized” response.
* Exception handling delegates need to be called early on in the pipeline, so they are able to catch exceptions that occur in deeper calls within the pipeline.
* The ASP.NET request pipeline consists of a sequence of request delegates, called one after the next, as this diagram shows (the thread of execution follows the black arrows)



# Security

## Content Security Policy