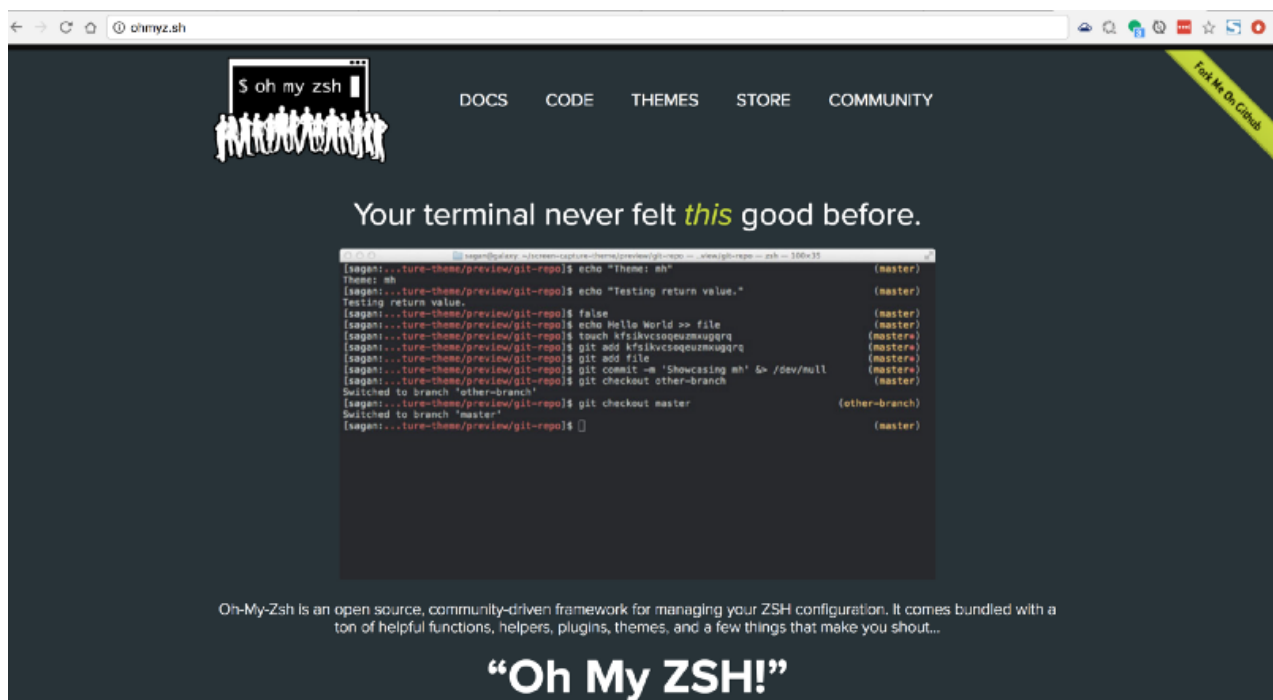


Stay safe, friends. Learn to code from home. Use our free 2,000 hour curriculum.

13 MARCH 2018 / #JAVASCRIPT

Jazz Up Your "ZSH" Terminal In Seven Steps — A Visual Guide



by rajaraodv

In this blog I'll cover installing ITerm2, ZSH shell, "oh my ZSH", Themes, ITerm2 color schemes, "oh my ZSH" plugins and enable "ligature" support to help create a beautiful and powerful Terminal.

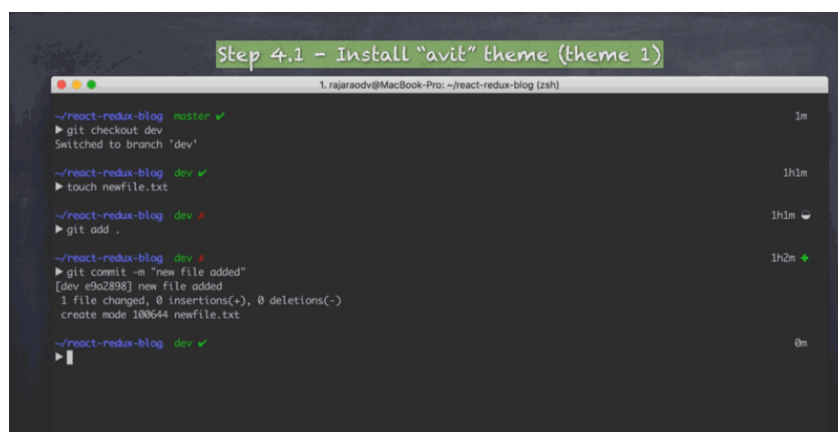
If you want to just make your regular Bash Terminal powerful, take a look at my previous blog: "[Jazz Up Your Bash Terminal](#)". But ZSH explained in this blog is more powerful stuff.

Summary:

We'll be covering a lot of things. This may be confusing, so here is the summary of what we will be doing.

1. Install ITerm2 — This is a better alternative to the default Terminal
2. Install latest ZSH shell — This is more powerful than the regular bash shell. We will switch ITerm2 to use ZSH shell.

3. Install "Oh My ZSH"— This is a CLI tool to easily configure ZSH and add themes and plugins to ZSH
4. Add two types of Themes using "Oh My ZSH"— some themes need extra steps so we will cover both
5. Install different ITerm2 Schemes — These are just color schemes for the UI
6. Add two different Plugins using "Oh My ZSH" to improve productivity
7. Enable "ligature" support so when you write an arrow =>;, it appears like a real arrow →



```
Step 4.1 - Install "avit" theme (theme 1)
1. rajarsodv@MacBook-Pro: ~/react-redux-blog (zsh)

~/react-redux-blog master ✓
└─▶ git checkout dev
Switched to branch 'dev'

~/react-redux-blog dev ✓
└─▶ touch newFile.txt

~/react-redux-blog dev ✓
└─▶ git add .

~/react-redux-blog dev ✓
└─▶ git commit -m "new file added"
[dev e9a2898] new file added
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 newFile.txt

~/react-redux-blog dev ✓
└─▶
```

Step 1— Install ITerm2

A lot of programmers like ITerm2 instead of the default Terminal. It is similar to the Terminal, but has lots of features of its own. It of course can run ZSH, Bash, and other shells inside it.

The following video shows some of the new features of Item 2 (v3).

For this blog we'll use ITerm2. When I mention "Terminal", I mean ITerm2. Although the steps are the same for both Terminal or ITerm2.

Step 2— Change Shell To ZSH

The following video shows why ZSH is better than just a bash shell.

Option 1 — Use Mac's own ZSH:

Mac comes with a ZSH out-of-the-box,

so we don't need to install it. However, sometimes it's an older version of ZSH. Typically it's located at `/bin/zsh`. To use it, all we need to do is to change shell (chsh).

1. Open the Terminal (or iTerm2) and type the following command.

```
$ chsh -s $(which zsh)
```

2. Enter the password and it will change the shell, upon logout and login.

3. Logout and re-login

4. To test, open the Terminal and type the following, and it should say zsh.

```
$ echo $0  
zsh //should return zsh
```

Option 2— Install Homebrew and Install latest ZSH via Homebrew

This option is pretty common among users, because some of the plugins only work with the latest ZSH.

Homebrew, simply said, is a command line installer for all sorts of software. Let's install that first.

1. Install Homebrew by running the following command.

```
ruby -e "$(curl -fsSL https://raw.githubusercontent.com
```

2. If you get Command Line Tools for Xcode error, it means you haven't installed the CLI tools for Xcode. *If you don't get the error, you can skip this step, because you already have it installed.*

XCode Developer CLI tools are used by various apps that manipulate core OSX features. So make sure to install the Xcode CLI tools by running the following command.

```
$ xcode-select --install
```

Note: The above command opens up Mac's installer and installs the XCode Developer CLI tools. If it doesn't work, try `xcode-select -r` to reset.

3. Install ZSH via Homebrew

Run the following command to install

🔍 Search 6,000+ tutorials

ZSH. It gets installed at `/usr/local/bin/zsh`. PS: Mac's default ZSH is at `/bin/zsh`

```
brew install zsh
```

4. Use the Homebrew version of ZSH

Run the following command. You will be prompted to enter Mac's password.

```
chsh -s /usr/local/bin/zsh
```

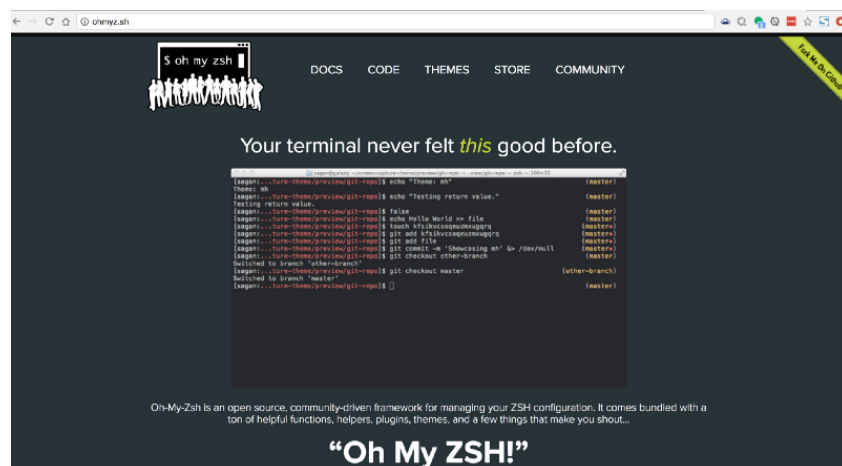
5. Logout and log back in.

6. Test if we are using ZSH and the correct ZSH

```
$ echo $0  
zsh    //correct
```

```
$ which zsh  
/usr/local/bin/zsh    //correct
```

Step 3— "Oh My ZSH"



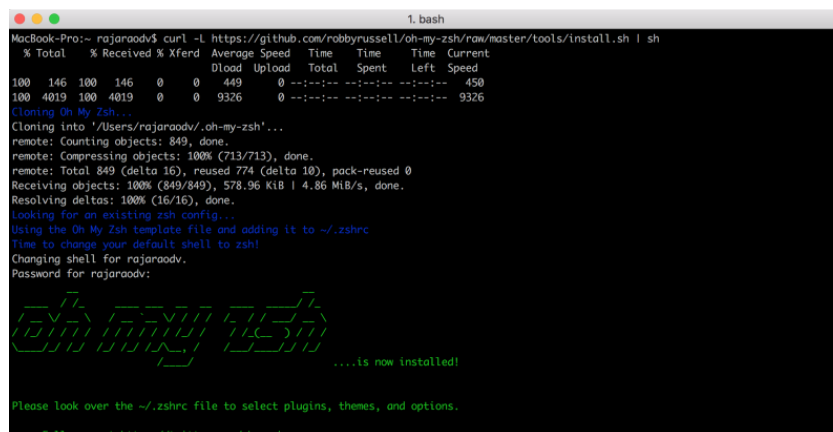
"Oh My ZSH" is a plugin that runs on top of ZSH. It provides default config for ZSH (~/.zshrc file) and also provides themes and more features.

From what I know, most power users who use ZSH also use "Oh My ZSH".

1. Install "Oh My ZSH"

Run the following command to install "oh My ZSH".

```
sh -c "$(curl -fsSL https://raw.githubusercontent.com/robbyrussell/oh-my-zsh/master/tools/install.sh)"
```



```
MacBook-Pro:~ rajaraodv$ curl -L https://github.com/robbyrussell/oh-my-zsh/raw/master/tools/install.sh | sh
% Total % Received % Xferd Average Speed Time Time Time Current
Dload Upload Total Spent Left Speed
100 146 100 146 0 0 449 0 --:--:-- --:--:-- --:--:-- 450
100 4019 100 4019 0 0 9326 0 --:--:-- --:--:-- --:--:-- 9326
Cloning Oh My Zsh...
Cloning into '/Users/rajaraodv/.oh-my-zsh'...
remote: Counting objects: 849, done.
remote: Compressing objects: 100% (713/713), done.
remote: Total 849 (delta 16), reused 774 (delta 10), pack-reused 0
Receiving objects: 100% (849/849), 578.96 KiB | 4.86 MiB/s, done.
Resolving deltas: 100% (16/16), done.
Looking for an existing zsh config...
Using the Oh My Zsh template file and adding it to ~/.zshrc
Time to change your default shell to zsh!
Changing shell for rajaraodv.
Password for rajaraodv:
...is now installed!
Please look over the ~/.zshrc file to select plugins, themes, and options.
... Follow up at https://github.com/ohmyzsh
```

Oh My ZSH is installed

2. Close and quit ITerm2 and reopen it.

It should look something like below.

Notice that the prompt has changed and the theme is a bit different — That’s “Oh My ZSH” in action for you.



Initial Oh My ZSH Theme

Step 4— Change Themes And Install Fonts

In this step, we’ll add two different “Oh My ZSH” Themes. “Oh My ZSH” comes with tons of themes. *PS: But some Themes need extra steps like installing specific fonts and so on.*

To set a Theme, simply open ~/.zshrc file (created by “Oh My ZSH”) and change the theme as shown below.

PS: .zshrc is the config file for ZSH shell. People who don’t use “Oh My ZSH” will have to manually

create this file and add any configs themselves. "Oh My ZSH" automatically creates this file if it doesn't exist and then adds its own set of configs into this file.

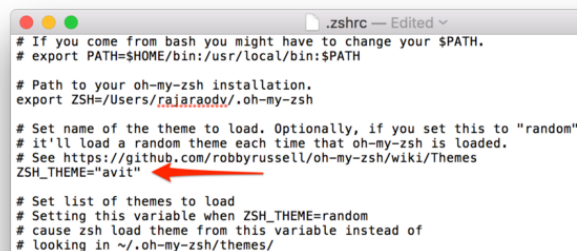
Theme 1 — Let's add a Theme called "Avit"

1. Open .zshrc

```
$ open ~/.zshrc
```

2. Change the Theme to "Avit"

You can browse all the "Oh My ZSH" Themes [here](#). To change the Theme, simply change the ZSH_THEME value in ~/.zshrc file from **robbyrussell** to **Avit**.



```
.zshrc — Edited
# If you come from bash you might have to change your $PATH.
# export PATH=$HOME/bin:/usr/local/bin:$PATH

# Path to your oh-my-zsh installation.
export ZSH=/Users/rajaarady/.oh-my-zsh

# Set name of the theme to load. Optionally, if you set this to "random"
# it'll load a random theme each time that oh-my-zsh is loaded.
# See https://github.com/robbyrussell/oh-my-zsh/wiki/Themes
ZSH_THEME="avit"

# Set list of themes to load
# Setting this variable when ZSH_THEME=random
# cause zsh load theme from this variable instead of
# looking in ~/.oh-my-zsh/themes/
```

```
# An empty array have no effect
# ZSH_THEME_RANDOM_CANDIDATES=( "robbyrussell" "agnoster" )

# Uncomment the following line to use case-sensitive completion.
# CASE_SENSITIVE="true"

# Uncomment the following line to use hyphen-insensitive completion. Case
# sensitive completion must be off. _ and - will be interchangeable.
# HYPHEN_INSENSITIVE="true"

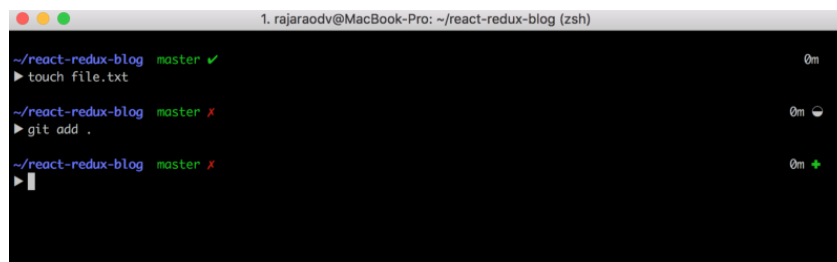
# Uncomment the following line to disable bi-weekly auto-update checks.
# DISABLE_AUTO_UPDATE="true"

# Uncomment the following line to change how often to auto-update (in days).
# export UPDATE_ZSH_DAYS=13
```

3. Update ZSH config

Run the following command to update the config.

```
$ source ~/.zshrc
```

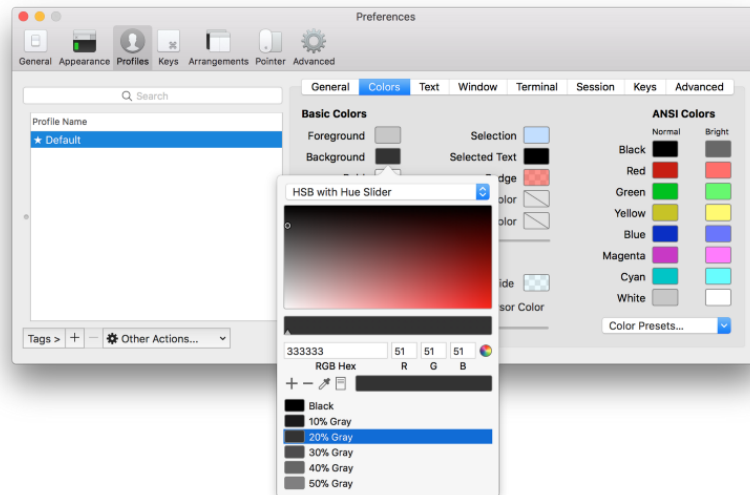


Your command prompt in Avit Theme

4. Change the background color and font size

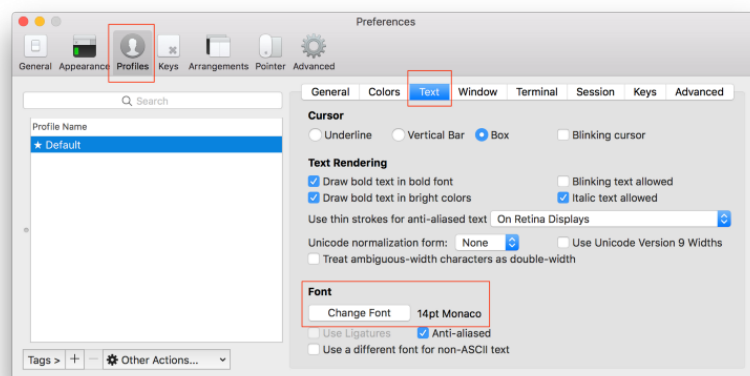
Open iTerm2 > Preferences > Profiles >

Colors and change the background black color to use 20% gray as shown below.




Use 20% Gray background

Then open Text > Change Font and change the size to 14pt.



Change font to 14pt



```
1.rajarao@MacBook-Pro: ~/react-redux-blog (zsh)

~/react-redux-blog master ✓
└─▶ git checkout dev
    Switched to branch 'dev'

~/react-redux-blog dev ✓
└─▶ touch newfile.txt

~/react-redux-blog dev ✗
└─▶ git add .

~/react-redux-blog dev ✗
└─▶ git commit -m "new file added"
    [dev e9a2898] new file added
    1 file changed, 0 insertions(+), 0 deletions(-)
    create mode 100644 newfile.txt

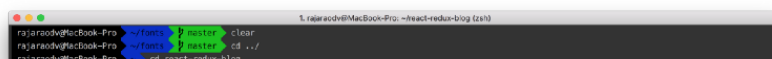
~/react-redux-blog dev ✓
└─▶
```

A clean and beautiful Iterm2 with ZSH!

OK, Let's install a different Theme that needs fonts.

Theme 2 — Installing "agnoster" Oh My ZSH theme

This is a popular theme because it emulates the Powerline Python app that enhances the terminal. The following picture shows how it looks. But this theme also needs us to install Powerline themes.



```
1.rajarao@MacBook-Pro: ~/react-redux-blog (zsh)

rajarao@MacBook-Pro: ~/react-redux-blog (zsh)
rajarao@MacBook-Pro: ~/react-redux-blog (zsh)
rajarao@MacBook-Pro: ~/react-redux-blog (zsh)
```



```
rajarad@MacBook-Pro: ~/react-redux-blog $ git checkout master
Switched to branch 'master'
Your branch is ahead of 'origin/master' by 3 commits.
(see 'git push' to publish your local commits)
rajarad@MacBook-Pro: ~/react-redux-blog $ ls -al
total 80
drwxr-xr-x 21 rajarad staff 672 Mar 3 06:04 .
drwxr-xr-x 20 rajarad staff 608 Mar 3 06:05 ..
-rw-r--r-- 1 rajarad staff 48 Mar 3 04:49 babelrc
drwxr-xr-x 13 rajarad staff 416 Mar 3 06:04 .git
-rw-r--r-- 1 rajarad staff 854 Mar 3 04:49 .gitignore
-rw-r--r-- 1 rajarad staff 14 Mar 3 04:49 Profile
-rw-r--r-- 1 rajarad staff 5558 Mar 3 04:49 README.md
-rw-r--r-- 1 rajarad staff 335 Mar 3 04:49 app.js
-rw-r--r-- 1 rajarad staff 279 Mar 3 04:49 app.json
drwxr-xr-x 3 rajarad staff 96 Mar 3 04:40 bin
-rw-r--r-- 1 rajarad staff 0 Mar 3 06:04 file.txt
```

agnoster Oh My ZSH theme

1. Install Powerline fonts

```
$ git clone https://github.com/powerline/font
$ cd fonts
$ ./install.sh
```

2. Change the Theme to “agnoster”

```
$ open ~/.zshrc
Set ZSH_THEME="agnoster" and save the file
```

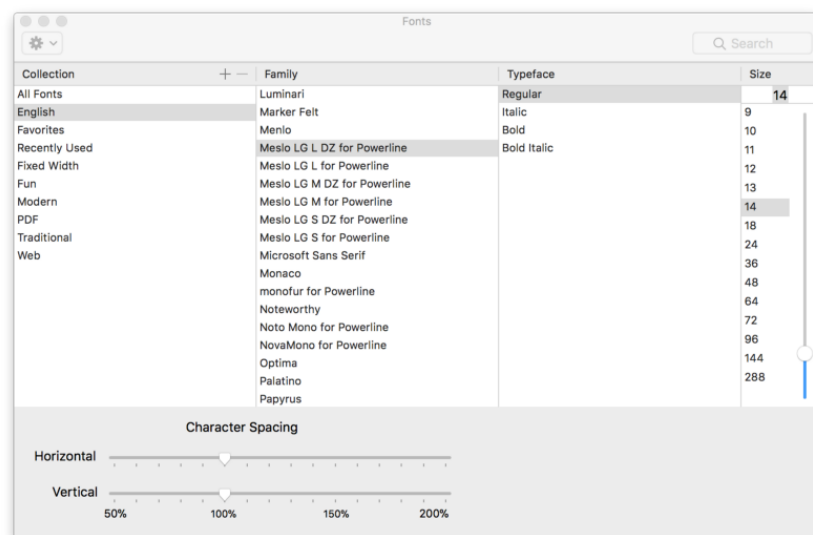
3. Quit iTerm2 and reopen it.

4. Set Powerline font

You can set any Powerline patched font you like. All the fonts end with “**for**”

Powerline".

Open ITerm2 > Preferences > Profiles > Text > Change Font and set it to something that has "for Powerline". I'm choosing **"Meslo LG DZ for Powerline"** font.



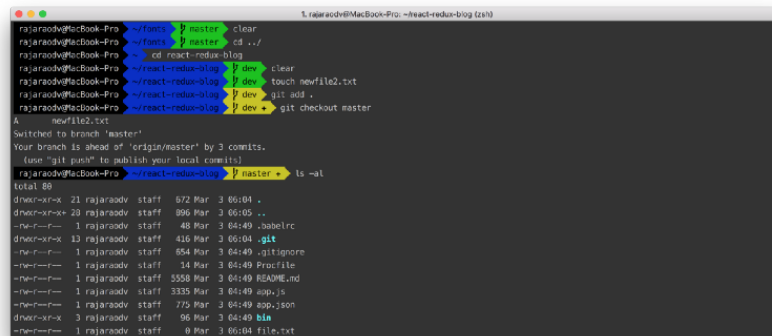
Meslo LG DZ for Powerline Iterm2 font

Note — If you are confused about the fonts and Themes: the Themes are for "Oh My ZSH" and ZSH shell and the fonts are for the Iterm2 itself.

5 All Done

5. All Done

At this point your Terminal should look like below:



```
rajaraodv@MacBook-Pro: ~/react-redux-blog (zsh)
rajaraodv@MacBook-Pro: ~/react-redux-blog (zsh) $ clear
rajaraodv@MacBook-Pro: ~/react-redux-blog (zsh) $ cd ../
rajaraodv@MacBook-Pro: ~/react-redux-blog (zsh) $ cd react-redux-blog
rajaraodv@MacBook-Pro: ~/react-redux-blog (zsh) $ clear
rajaraodv@MacBook-Pro: ~/react-redux-blog (zsh) $ touch newfile2.txt
rajaraodv@MacBook-Pro: ~/react-redux-blog (zsh) $ git add .
rajaraodv@MacBook-Pro: ~/react-redux-blog (zsh) $ git checkout master
A
 newfile2.txt
Switched to branch 'master'
Your branch is ahead of 'origin/master' by 3 commits.
To push
  git push
rajaraodv@MacBook-Pro: ~/react-redux-blog (zsh) $ git push
rajaraodv@MacBook-Pro: ~/react-redux-blog (zsh) $ cd ..
rajaraodv@MacBook-Pro: ~/react-redux-blog (zsh) $ ls -la
total 88
drwxr-xr-x  21 rajaraodv  staff   672 Mar 3  06:04 .
drwxr-xr-x  20 rajaraodv  staff   640 Mar 3  06:05 ..
-rw-r--r--   1 rajaraodv  staff    48 Mar 3  06:49 .babelrc
drwxr-xr-x  12 rajaraodv  staff   416 Mar 3  06:04 .git
-rw-r--r--   1 rajaraodv  staff   654 Mar 3  06:49 .gitignore
-rw-r--r--   1 rajaraodv  staff   144 Mar 3  06:49 Procfile
-rw-r--r--   1 rajaraodv  staff  5558 Mar 3  06:49 README.md
-rw-r--r--   1 rajaraodv  staff   335 Mar 3  06:49 app.js
-rw-r--r--   1 rajaraodv  staff   776 Mar 3  06:49 app.json
drwxr-xr-x   3 rajaraodv  staff   96 Mar 3  06:49 bin
-rw-r--r--   1 rajaraodv  staff    8 Mar 3  06:04 file.txt
```

Step 5 — Install iTerm2 "color schemes" (ITerm2 Themes)

There are plenty of gorgeous color schemes for iTerm2. These schemes change the foreground color, background color, cursor color, and so on. You can find them at [iTerm2-color-schemes Github repo](https://github.com/mbadolato/iTerm2-color-schemes).

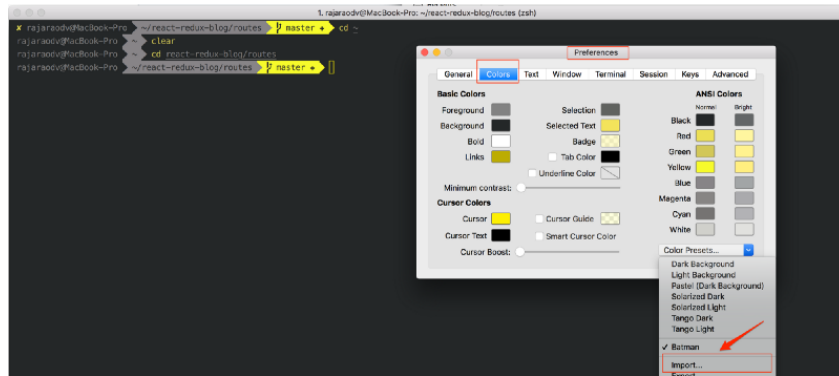
Note: These are just color schemes of the iTerm2 UI and don't deal with the command prompt's look and feel like "Oh My ZSH"'s themes (other than

just changing colors).

Follow these steps to install them.

1. Download the iTerm2-color-schemes as a zip file and extract it
2. The "Schemes" folder contains all the color scheme files — they end with `.itermcolors`
3. Open iTerm2 > Preferences > Profile > Colors > Color Presets > Import
4. In the import window, navigate to the "Schemes" folder (from step 2)
5. Select all the files so you can import all the color schemes at once
6. Simply select whichever color scheme you like.

My favorites are **Batman** and **Argonaut**



Batman Iterm2 Theme

The Argonaut color scheme looks like below:



Argonaut color scheme

Step 6— Install Plugins

Plugins add more functionalities to your workflow. By default “Oh My ZSH” already has the “git” plugin! and that’s why you were able to see all those Git statuses in the prompts in earlier

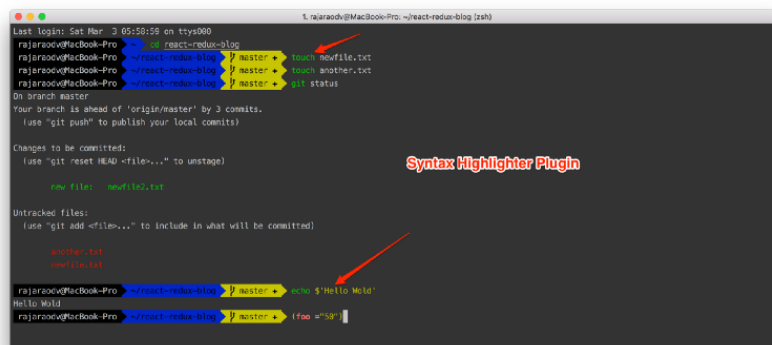
status in the prompts in earlier

screenshots. Let's add another one to see how it works.

Note: In this section, we'll install two different plugins to show how they work.

Plugin 1 — Add Syntax Highlighting Plugin

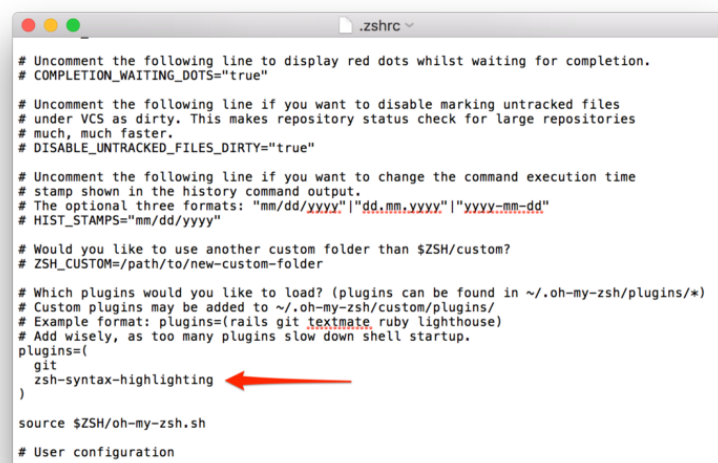
The Syntax Highlighting plugin adds beautiful colors to the commands you are typing as shown below.

A screenshot of a terminal window titled 'rajaraodv@MacBook-Pro: ~/react-redux-blog (zsh)'. The terminal shows a series of commands and their outputs, with syntax highlighting applied to the commands. For example, 'cd react-redux-blog' has 'cd' in blue and 'react-redux-blog' in green. 'touch newfile.txt' has 'touch' in yellow and 'newfile.txt' in green. 'git status' has 'git' in blue and 'status' in green. The prompt 'rajaraodv@MacBook-Pro' is in blue. A red arrow points to the text 'Syntax Highlighter Plugin' in the middle of the terminal output. Another red arrow points to the command 'echo \$'Hello World'' at the bottom of the terminal.

1. Clone the zsh-syntax-highlighting plugin's repo and copy it to the "Oh My ZSH" plugins directory.

```
git clone https://github.com/zsh-users/zsh-syntax-highlighting
```

2. Activate the plugin in `~/.zshrc` by adding `zsh-syntax-highlighting` to the Plugins section as shown below.



```
# Uncomment the following line to display red dots whilst waiting for completion.
# COMPLETION_WAITING_DOTS="true"

# Uncomment the following line if you want to disable marking untracked files
# under VCS as dirty. This makes repository status check for large repositories
# much, much faster.
# DISABLE_UNTRACKED_FILES_DIRTY="true"

# Uncomment the following line if you want to change the command execution time
# stamp shown in the history command output.
# The optional three formats: "mm/dd/yyyy"|"dd.mm.yyyy"|"yyyy-mm-dd"
# HIST_STAMPS="mm/dd/yyyy"

# Would you like to use another custom folder than $ZSH/custom?
# ZSH_CUSTOM=/path/to/new-custom-folder

# Which plugins would you like to load? (plugins can be found in ~/.oh-my-zsh/plugins/*)
# Custom plugins may be added to ~/.oh-my-zsh/custom/plugins/
# Example format: plugins=(rails git textmate ruby lighthouse)
# Add wisely, as too many plugins slow down shell startup.
plugins=(
  git
  zsh-syntax-highlighting
)

source $ZSH/oh-my-zsh.sh

# User configuration
```

Add a new plugin in a new line inside plugins section

3. Re-read zshrc configuration

```
source ~/.zshrc
```

Plugin 2 — Add ZSH-AutoSuggestion Plugin

This plugin auto suggests any of the previous commands. Pretty handy! To select the completion, simply press → key.



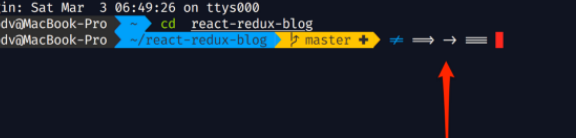
1. Install the plugin

```
git clone https://github.com/zsh-users/zsh-autosuggestions
```

PS: ZSH_CUSTOM points to
~/oh-my-zsh/custom

2. Open ~/.zshrc and add zsh-autosuggestions

Step 7 — Use Ligature Support



A terminal window titled "1. rajaraodva@MacBook-Pro: ~/react-redux-blog (zsh)" displays the following text:

```
Last login: Sat Mar 3 06:49:26 on ttys000
rajaraodva@MacBook-Pro ~ % cd react-redux-blog
rajaraodva@MacBook-Pro ~/react-redux-blog % master ➤ => -> ==
```

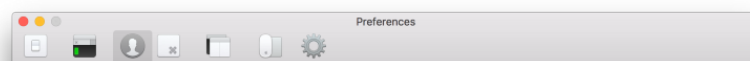
An orange arrow points from the text below to the ligature symbols in the terminal output.

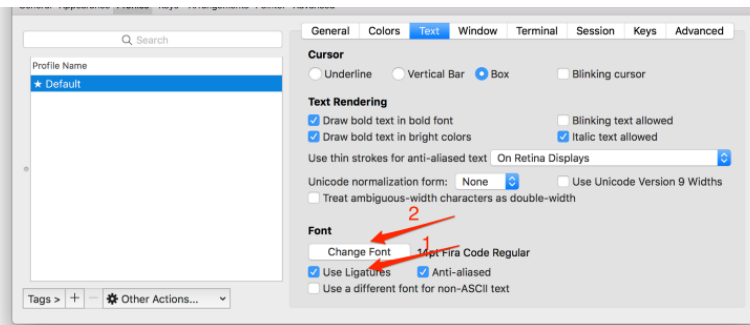
With FiraCode font and ligature support enabled, !=, ==>, ->, == symbols look like above



To use this, we need fonts that support ligatures. We also need to enable it in ITerm2. FiraCode is one such font. Follow the steps to install and enable ligatures.

1. Download the FiraCode repo and extract the zip file (or clone it)
2. Open the `dstr > ttf` folder and double click on all the `*.ttf` files and select the "Install font" button to install each of the font variations.
3. Navigate to ITerm2 | Preferences | Profiles | Text
4. **Select Use Ligatures checkbox**
5. Click on Change Font and select Fira Code Regular font





Summary

We have covered a lot in this blog starting from installing latest ZSH via Homebrew, Oh My ZSH, Plugins, Themes, enable “ligatures” for FiraCode font.

?? Thank you!

If this was useful, please click the clap ? button down below a few times to show your support! ↓ ↓ ↓ ??

My Other Posts

<https://medium.com/@rajaraodv/latest>

ECMAScript 2015+

1. *Check out these useful ECMAScript*

2015 (ES6) tips and tricks

2. 5 JavaScript "Bad" Parts That Are Fixed In ES6
3. Is "Class" In ES6 The New "Bad" Part?

Terminal Improvements

1. How to Jazz Up Your Terminal — A Step By Step Guide With Pictures
2. Jazz Up Your "ZSH" Terminal In Seven Steps — A Visual Guide

WWW

1. A Fascinating And Messy History Of The Web And JavaScript

Virtual DOM

1. Inner Workings Of The Virtual DOM

React Performance

1. Two Quick Ways To Reduce React

App's Size In Production

2. Using Preact Instead Of React

Functional Programming

1. JavaScript Is Turing Complete — Explained
2. Functional Programming In JS — With Practical Examples (Part 1)
3. Functional Programming In JS — With Practical Examples (Part 2)
4. Why Redux Need Reducers To Be "Pure Functions"

Webpack

1. Webpack — The Confusing Parts
2. Webpack & Hot Module Replacement [HMR] (under-the-hood)
3. Webpack's HMR And React-Hot-Loader — The Missing Manual

Draft is

[Draft.js](#)

1. [Why Draft.js And Why You Should](#)

[Contribute](#)

2. [How Draft.js Represents Rich Text Data](#)

React And Redux :

1. [Step by Step Guide To Building React Redux Apps](#)
2. [A Guide For Building A React Redux CRUD App \(3-page app\)](#)
3. [Using Middlewares In React Redux Apps](#)
4. [Adding A Robust Form Validation To React Redux Apps](#)
5. [Securing React Redux Apps With JWT Tokens](#)
6. [Handling Transactional Emails In React Redux Apps](#)
7. [The Anatomy Of A React Redux App](#)
8. [Why Redux Need Reducers To Be "Pure Functions"](#)
9. [Two Quick Ways To Reduce React](#)

7. TWO QUICK WAYS TO REDUCE REDUCED App's Size In Production

**If this was useful, please click
the clap ? button below a few
times to show your support!**
↓↓↓ ??

If you have questions, please feel free to
ask me on Twitter:

<https://twitter.com/rajaraodv>

If this article was helpful, [tweet it.](#)

Learn to code for free. freeCodeCamp's
open source curriculum has helped more
than 40,000 people get jobs as
developers. [Get started](#)

Continue reading about

LoveScript

JavaScript

The Principles of Functional Programming

How to Add TypeScript to a JavaScript Project

How JavaScript Works: Under the Hood of the V8 Engine

See all 1685 posts →



#REDUX

How to create your first Redux middleware with ease

2 YEARS AGO



#PROGRAMMING

Why coding and kindness go hand in hand, and how I learned to be kind to myself

2 YEARS AGO

freeCodeCamp is a donor-supported tax-exempt 501(c)(3) nonprofit organization (United States Federal Tax Identification Number: 82-0779546)

07 / 340,

Our mission: to help people learn to code for free. We accomplish this by creating thousands of videos, articles, and interactive coding

lessons - all freely available to the public. We also have thousands of freeCodeCamp study groups around the world.

Donations to freeCodeCamp go toward our education initiatives, and help pay for servers, services, and staff.

You can [make a tax-deductible donation here](#).

JavaScript Closure

CSS Box Shadow

Python List Append

JavaScript Array Sort

Symlink in Linux

Linux Grep Command

What is DNS?

Primary Key SQL

SQL Update Statement

Screenshot on PC

About Alumni Network Open Source Shop Support Sponsors Academic Honesty Code of Conduct