

# 為 MAC 的 Terminal 上色 - 透過 iTerm 2 和 Oh My Zsh 高亮你的終端機

in [其他資源分享](#), [網頁設計資源分享](#) with [沒有留言](#)

```
~ ➤ cd testproject
~/testproject ➤ master ➤ gco detached-head-state -q
~/testproject ➤ fdffaf6 ➤ touch dirty-working-directory
~/testproject ➤ fdffaf6± ➤ cd
~ ➤ ssh milly
Welcome to Ubuntu 11.04 (GNU/Linux 2.6.18-308.8.2.el5.028stab101.1 x86_64)
Last login: Wed Sep 26 03:42:49 2012 from 71-215-222-90.mpls.qwest.net
agnoster@milly ➤ ~ ➤
Connection to milly.agnoster.net closed.
~ ➤ sudo -s
Password:
~ ➤ root@Arya ➤ ~ ➤ top &
[1] 34523
[1] + 34523 suspended (tty output) top
~ ➤ root@Arya ➤ ~ ➤ rm no-such-file
rm: no-such-file: No such file or directory
~ ➤ root@Arya ➤ ~ ➤ kill %1
[1] + 34523 terminated top
~ ➤ root@Arya ➤ ~ ➤
```

奇怪，為什麼你的終端機畫面可以長的這麼好看阿？我的換來換去卻還是長的像是個 Hacker ...

```
pjchender — exec bash — exec — bash — 80x24
Last login: Sat Feb 18 19:03:22 on ttys000
~ ➤ exec bash
bash-3.2$ echo "Hello Everyone"
Hello Everyone
bash-3.2$ cd Projects/
bash-3.2$ ls
CompleteProject TreeHouse cope-tech rails_blog
F2EProjects Wavinfo depot slider
JavaScript30 bundler rails50
bash-3.2$ cd JavaScript30/
bash-3.2$ ls
01 - JavaScript Drum Kit
02 - JS + CSS Clock
03 - CSS Variables
04 - Array Cardio Day 1
05 - Flex Panel Gallery
06 - Type Ahead
07 - Array Cardio Day 2
08 - Fun with HTML5 Canvas
09 - Dev Tools Domination
10 - Hold Shift and Check Checkboxes
11 - Custom Video Player
12 - Key Sequence Detection
13 - Slide in on Scroll
```

今天，就來讓我們看一下可以怎麼做把 MAC 的終端機畫面用的更有人性吧。

讓 MAC 終端機高亮的過程雖然看起來很複雜又冗長，但是如果你是常會使用終端機的使用者，又希望有一個比較好看的視覺介面，只要按照下面一步步跟著做就可以完成了。

在下面的終端機的指令中，\$ 只是表示是終端機指令的開頭，不需要複製輸入到終端機中。

## 安裝 Homebrew

[Homebrew](#) 是 Mac OSX 上的套件管理工具，是方便安裝管理 OSX 裡需要用到但預設沒安裝的套件。Homebrew 的安裝只需要打開終端機，輸入：

```
$ /usr/bin/ruby -e "$(curl -fsSL  
https://raw.githubusercontent.com/Homebrew/install/master/install)"
```

就可以了。接著可以輸入以下指令確認是否有安裝成功：

```
$ brew --version
```

## 安裝 iTerm2

[iTerm2](#) 是一個可以用來取代 MAC 原生終端機的工具，擁有許多[更方便的功能和特色](#)，只需到[官網下載](#)安裝就可以了。

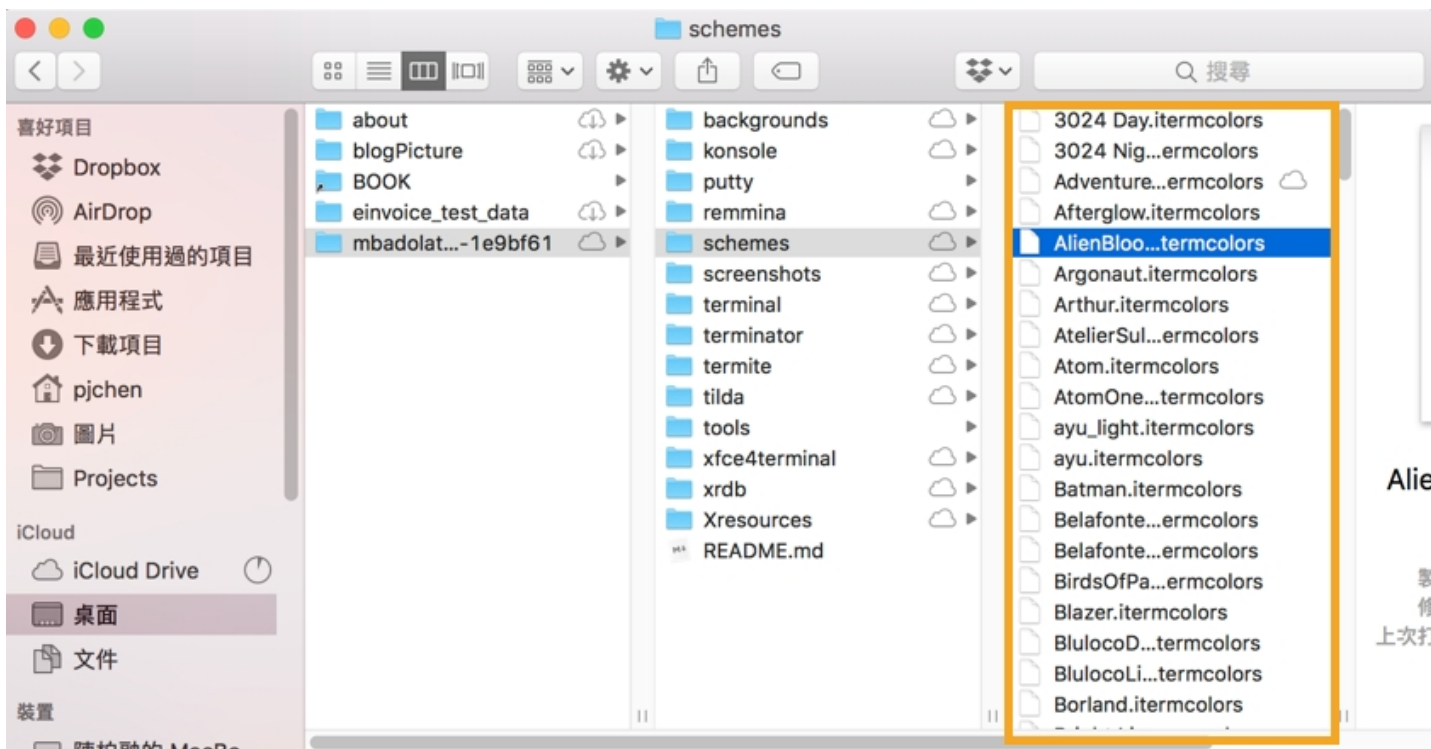
或者也可以在 terminal 中輸入下述的指令，直接透過 homebrew 安裝：

```
$ brew cask install iterm2
```

## iTerm2 配色主題下載

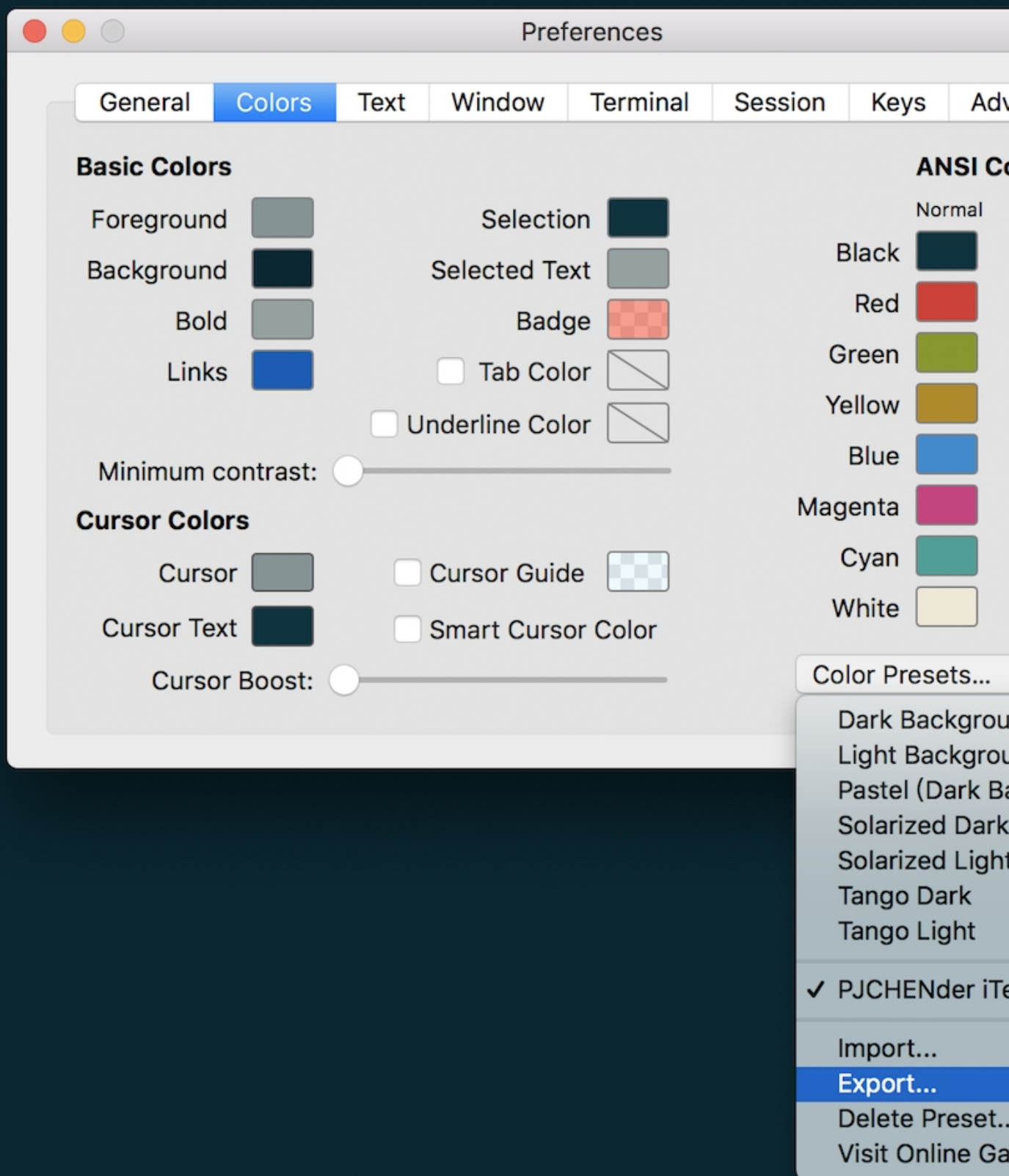
iTerm2 提供非常多的配色主題可以下載，可以到 [iterm2 color schemes](#) 的網站選擇喜歡的配色，右上角有下載的標示可以整包下載下來，下面則會顯示各種配色。

解壓縮後在 schemes 中有許多副檔名為 .itermcolors 的配色檔可以載入，這裡我們選擇 [Solarized Dark Theme Patched Version \(下載\)](#)：



## iTerm2 套用配色

選好喜歡的配色後，打開 iTerm2 後按快捷鍵 `cmd + i`，選擇上方名為 color 的頁籤，在右下方的 Color Presets 中透過 import 匯入配色後再點選一次 Color Preset 就可以選取：



## 安裝 ZSH

接著我們要安裝 ZSH，這是用來取代 BASH 的一種工具，打開 Terminal 輸入：

```
$ brew install zsh zsh-completions
```

確認安裝成功

```
$ zsh --version
```

接著要將預設終端機窗口改為 ZSH，一樣在終端機輸入：

```
$ sudo sh -c "echo $(which zsh) >> /etc/shells"  
$ chsh -s $(which zsh)
```

然後重新開啟你的 Terminal，輸入：

```
$ echo $SHELL
```

如果有成功變更 SHELL 為 ZSH 的話應該會看到 /usr/local/bin/zsh。

## 安裝 Oh My ZSH

[Oh My Zsh](#) 是一個用來管理 ZSH 設定檔（configuration）的框架，提供了很多的外掛（plugin）和主題（theme）可以選擇。

安裝的方式一樣是透過 Terminal，輸入：

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

接下來我們要套用主題，輸入：

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

接著會打開 zsh 的設定檔，找到 ZSH\_THEME="..."，將這段的内容改成下面這段後，在透過 cmd + s 存檔：

```
ZSH_THEME="agnoster"
```

## 字體修改

在安裝的過程中，如果你發現出現一些看不懂的文字或亂碼時，很可能是因為沒有相對應的字體，因此我們要下載字體。

下載字體檔：[Meslo](#)（截圖中所使用的）或者也可以到 [Powerline fonts](#) @ github 中找到其他你要的字體後，在「view raw」上點右鍵選擇「另存連結」將字體檔下載後並安裝。

在 iTerm2 中套用字體（Meslo）：iTerm --> Preferences --> Profiles --> Text --> Change Font

## 其他外掛或建議

### Auto Suggestions (for Oh My ZSH)

[Auto Suggestions](#) 會根據你過去所輸入的指令來提示要輸入的內容。

在終端機輸入：

```
$ git clone git://github.com/zsh-users/zsh-autosuggestions
$ZSH_CUSTOM/plugins/zsh-autosuggestions
```

接著打開 ZSH 設定檔，找到 `plugins = ""` 的地方，在 `plugins` 的欄位中加入 `zsh-autosuggestion` 後存檔（若有多個 `plugin` 則以空白分開：

```
plugins=(zsh-autosuggestions)
```

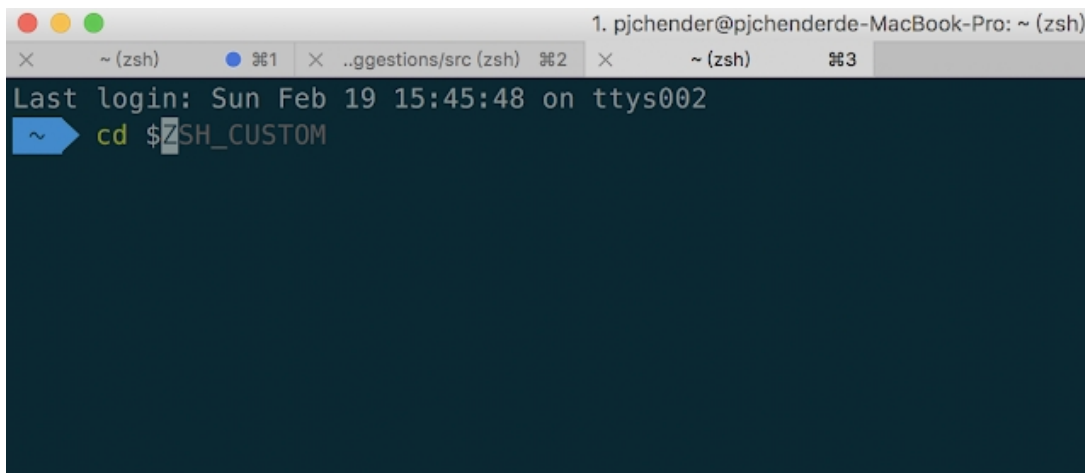
有時候你的背景色和提示的文字會太過接近，以致於你看不到 `autosuggestions` 的文字，這時候需要去設定顯示的文字。一樣先在終端機輸入以下文字，以開啟設定檔：

```
open $ZSH_CUSTOM/plugins/zsh-autosuggestions/zsh-autosuggestions.zsh
```

打開設定檔後，在文件中找到這行（預設會是 `fg=8`），這就是給終端機所使用的 256 色碼，你可以改成自己喜歡的顏色，這裡我用 `fg=240` 感覺比較合適。

```
ZSH_AUTOSUGGEST_HIGHLIGHT_STYLE='fg=240'
```

接著，只要你輸入過相關的文字，下次再次打入同樣的文字的時候，就會產生自動建議的文字，按下一→後就會自動帶入，像是這樣的效果：



### 程式碼高亮 (Syntax Highlighting)

在終端機輸入下面這段，下載 Syntax Highlighting：

```
$ brew install zsh-syntax-highlighting
```

接著一樣打開 ZSH 設定檔：

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

在設定檔的最下面加上這段

```
# For zsh syntax-highlighting
```

```
source /usr/local/share/zsh-syntax-highlighting/zsh-syntax-highlighting.zsh
```

## 客制化 Prompt

在終端機中預設的 prompt 會是 "user@hostname"，非常長一串，我們可以透過以下的方式讓那一長串不要顯示出來。

打開 ZSH 設定檔：

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

找個地方加入下面這段，DEFAULT\_USER 中 "yourname" 要填的是你使用者的名稱：

```
# .zshrc
```

```
# optionally set DEFAULT_USER in ~/.zshrc to your regular username to hide the  
"user@hostname" info when you're logged in as yourself on your local machine.  
DEFAULT_USER=yourname
```

如果不知道使用者的名稱的話，可以在終端機輸入 whoami。

## 參考

- [iTerm2 + Oh My Zsh + Solarized color scheme + Meslo powerline font + \[Powerlevel9k\] - \(macOS\) @ Gist](#)
- [Making ZSH default Shell in MacOSX @ StackOverflow](#)