


釀造升級日誌

==> 完成安裝 `ln -s ../../Cellar/azure-cli/2.68.0/etc/bash_completion.d/az az` `ln -s ../../Cellar/azure-cli/2.68.0/bin/az`
`az` `ln -s ../../Cellar/azure-cli/2.68.0/share/fish/vendor_completions.d/az.fish az.fish` `ln -s ../../Cellar/azure-`
`cli/2.68.0/share/zsh/site-functions/_az az`==> 注意事項 zsh 自動完成功能已安裝至：`/opt/homebrew/share/zsh/site-`
`functions`==> 摘要  `/opt/homebrew/Cellar/azure-cli/2.68.0`: 24,507 個檔案，580.4MB ==> 執行 `brew cleanup`
`azure-cli`... 移除：`/opt/homebrew/Cellar/azure-cli/2.67.0_1`... (27,401 個檔案，647.1MB) 移除：`/Users/lzwjava/Library/Cac`
`cli_bottle_manifest-2.67.0_1`... (22.5KB) 移 除：`/Users/lzwjava/Library/Caches/Homebrew/azure-cli-2.67.0_1`...
(54MB) ==> 注意事項 ==> openjdk 為使系統 Java 封裝程式找到此 JDK，請使用符號連結：`sudo ln -sf`
`/opt/homebrew/opt/openjdk/libexec/openjdk.jdk /Library/Java/JavaVirtualMachines/openjdk.jdk`

openjdk 為 keg-only，意指它沒有符號連結到 `/opt/homebrew`，因為 macOS 提供類似的軟體，並行安裝此軟體
可能會造成各種問題。

如果您需要將 openjdk 放在 PATH 的最前面，請執行：`echo 'export PATH="/opt/homebrew/opt/openjdk/bin:$PATH"`
' `>> ~/.zshrc`

為使編譯器找到 openjdk，您可能需要設定：`export CPPFLAGS="-I/opt/homebrew/opt/openjdk/include"` ==>
ruby 預設情況下，gem 安裝的二進位檔將放置於：`/opt/homebrew/lib/ruby/gems/3.4.0/bin`

您可能需要將其新增至 PATH。

ruby 為 keg-only，意指它沒有符號連結到 `/opt/homebrew`，因為 macOS 已提供此軟體，並行安裝另一個版本可
能會造成各種問題。

如果您需要將 ruby 放在 PATH 的最前面，請執行：`echo 'export PATH="/opt/homebrew/opt/ruby/bin:$PATH"`
' `>> ~/.zshrc`

為使編譯器找到 ruby，您可能需要設定：`export LDFLAGS="-L/opt/homebrew/opt/ruby/lib"` `export CPPFLAGS=`
`"-I/opt/homebrew/opt/ruby/include"` ==> yt-dlp zsh 自動完成功能已安裝至：`/opt/homebrew/share/zsh/site-`
`functions`==> redis 升級後重新啟動 redis：`brew services restart redis` 或者，如果您不想要/不需要背景服
務，您只需執行：`/opt/homebrew/opt/redis/bin/redis-server /opt/homebrew/etc/redis.conf` ==> perl 預設情況
下，非 Homebrew 安裝的 cpan 模組會安裝到 Cellar。如果您希望 模組在更新後仍然存在，我們建議您使用
`local::lib`。

您可以這樣設定：`PERL_MM_OPT="INSTALL_BASE=HOME/perl5" cpanlocal :: lib shell /.profile /.zshrc`
`-IHOME/perl5/lib/perl5 -Mlocal :: lib =HOME/perl5"` ==> awscli “examples” 目錄已安裝至：
`/opt/homebrew/share/awscli/examples`

zsh 自動完成功能和函數已安裝至：`/opt/homebrew/share/zsh/site-functions`==> php 若要在 Apache 中啟用 PHP，
請將以下內容新增至 `httpd.conf` 並重新啟動 Apache：`LoadModule php_module /opt/homebrew/opt/php/lib/httpd/module`

```
<FilesMatch \.php$>
    SetHandler application/x-httpd-php
</FilesMatch>
```

最後，檢查 `DirectoryIndex` 是否包含 `index.php` `DirectoryIndex index.php index.html`

php.ini 和 php-fpm.ini 檔案位於：`/opt/homebrew/etc/php/8.4/`

立即啟動 php 並在登入時重新啟動：`brew services start php` 或者，如果您不想要/不需要背景服務，您只需執行：`/opt/homebrew/opt/php/sbin/php-fpm -nodaemonize` ==> nginx 文件根目錄為：`/opt/homebrew/var/www`
預設埠已在 `/opt/homebrew/etc/nginx/nginx.conf` 中設定為 8080，以便 nginx 可以不用 sudo 執行。

nginx 將載入 `/opt/homebrew/etc/nginx/servers/` 中的所有檔案。

升級後重新啟動 nginx：`brew services restart nginx` 或者，如果您不想要/不需要背景服務，您只需執行：`/opt/homebrew/opt/nginx/bin/nginx -g daemon off;` ==> git-lfs 更新您的 git 設定以完成安裝：

更新全域 git 設定 `$ git lfs install`

更新系統 git 設定 `$ git lfs install --system`

zsh 自動完成功能已安裝至：`/opt/homebrew/share/zsh/site-functions` ==> wireshark 此公式預設僅安裝命令列工具。

使用 Homebrew Cask 安裝 Wireshark.app：`brew install --cask wireshark`

如果您的可用擷取介面清單為空（預設 macOS 行為），請安裝 ChmodBPF：`brew install --cask wireshark-chmodbpf`
==> doctl zsh 自動完成功能已安裝至：`/opt/homebrew/share/zsh/site-functions` ==> azure-cli zsh 自動完成功能已安裝至：`/opt/homebrew/share/zsh/site-functions`