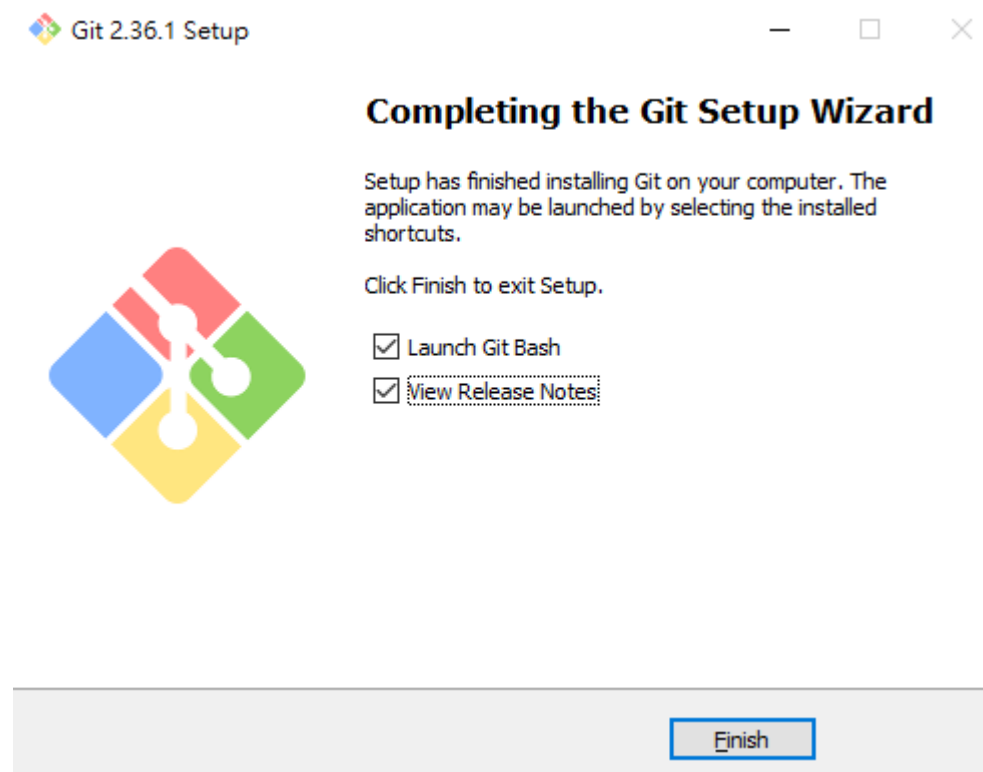


1. Git Bash

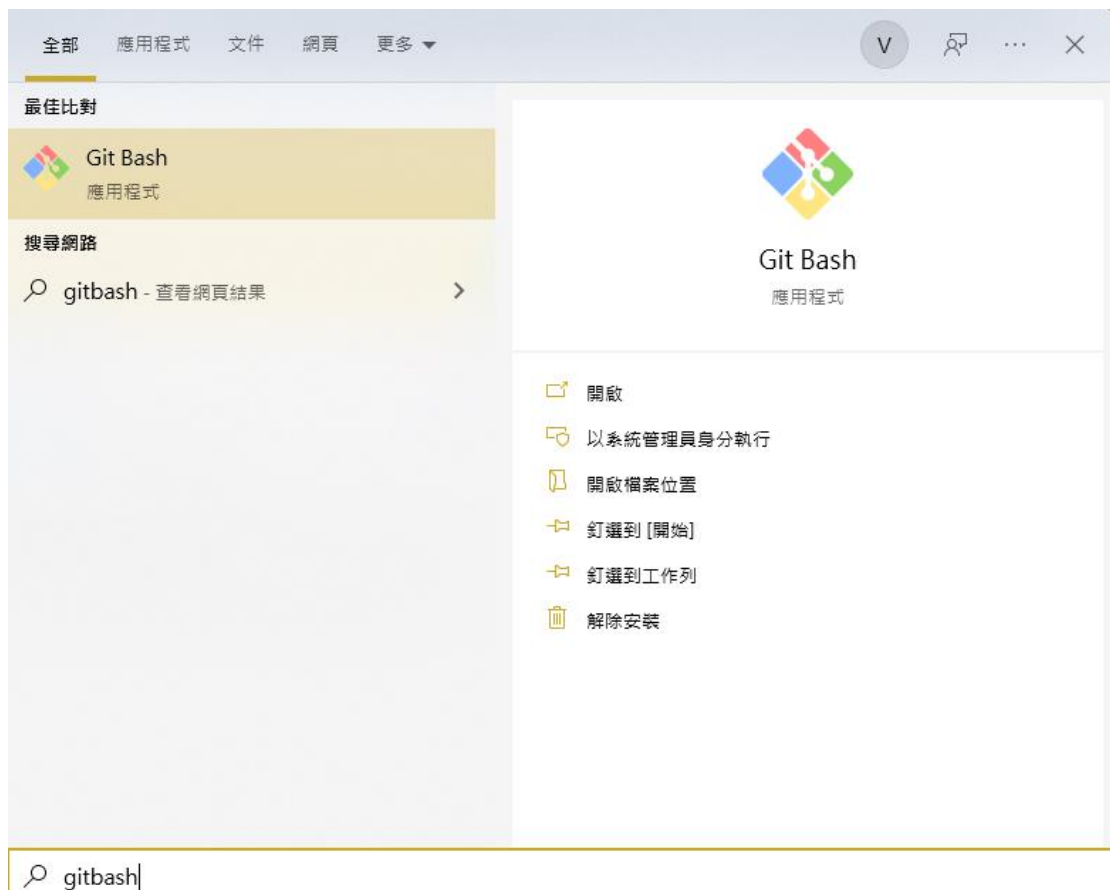
git bash 是 Windows 用戶安裝 git 時默認安裝的命令行工具，不僅界面漂亮功能也不錯，且大多數情況下可以替代 Windows 原生的 cmd 命令行。

2. 安裝 Git Bash

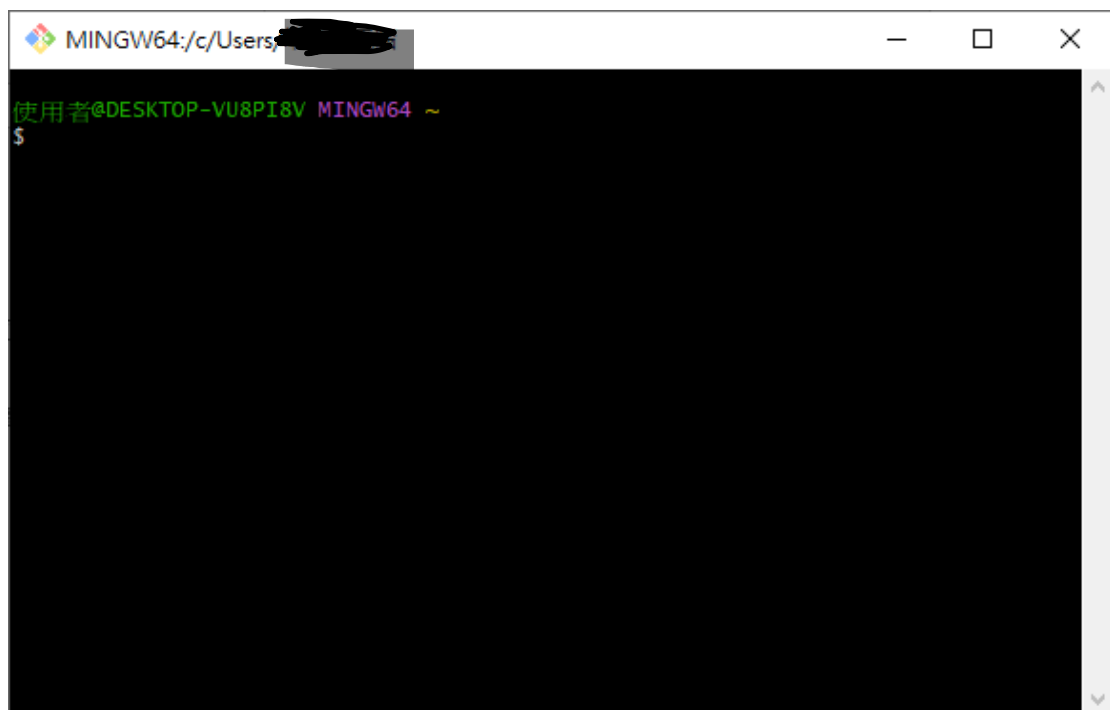
從[官網](#)中安裝



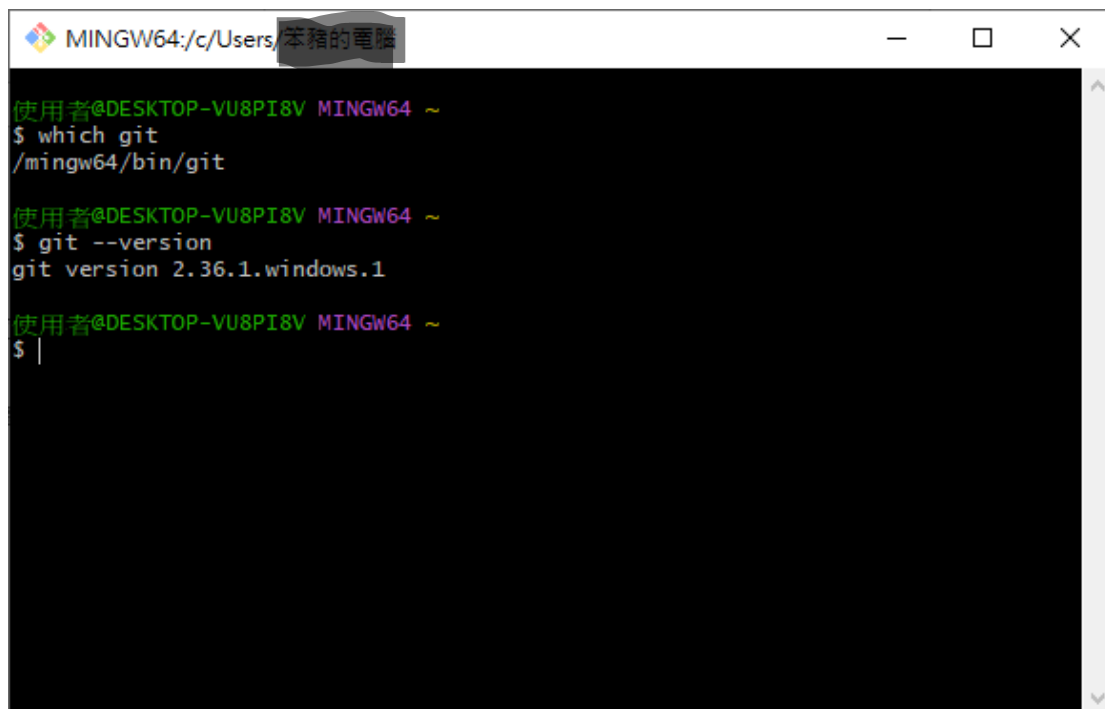
在最後要記得勾選 **Launch Git Bash**，安裝完成後就會自動開啟 **Git Bash** 主要操作的命令界面，如果沒跑出來也可以從電腦的搜尋中找出其並開啟。



開啟後的視窗會長這樣



可以從視窗中輸入指令，驗證是否有將 **git** 安裝成功，以及驗證版本資訊是否正確。

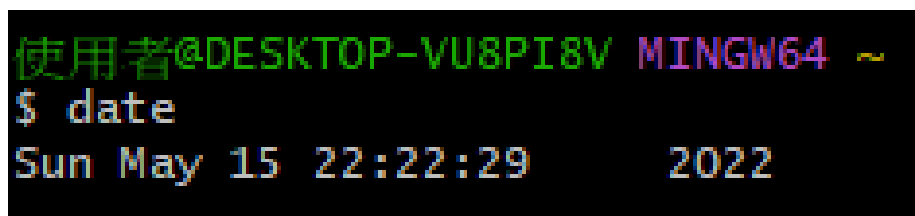
A terminal window titled 'MINGW64:/c/Users/笨豬的電腦' with standard window controls. The terminal shows three commands and their outputs: 'which git' returns '/mingw64/bin/git', 'git --version' returns 'git version 2.36.1.windows.1', and the prompt '\$ |' is shown on the third line.

```
使用者@DESKTOP-VU8PI8V MINGW64 ~  
$ which git  
/mingw64/bin/git  
  
使用者@DESKTOP-VU8PI8V MINGW64 ~  
$ git --version  
git version 2.36.1.windows.1  
  
使用者@DESKTOP-VU8PI8V MINGW64 ~  
$ |
```

這樣表示成功。

3.簡單的運用

\$ date

A terminal window showing the command 'date' and its output. The prompt is '使用者@DESKTOP-VU8PI8V MINGW64 ~'. The output is 'Sun May 15 22:22:29 2022'.

```
使用者@DESKTOP-VU8PI8V MINGW64 ~  
$ date  
Sun May 15 22:22:29 2022
```

在指令上打出 \$ date，就會顯示出當前的日期以及時間

\$ clear

```
使用者@DESKTOP-VU8PI8V MINGW64 ~  
$ which git  
/mingw64/bin/git  
  
使用者@DESKTOP-VU8PI8V MINGW64 ~  
$ git --version  
git version 2.36.1.windows.1  
  
使用者@DESKTOP-VU8PI8V MINGW64 ~  
$ date  
Sun May 15 22:22:29      2022  
  
使用者@DESKTOP-VU8PI8V MINGW64 ~  
$ clear|
```

打這個指令後，能夠將使用者介面的東西清除，如下

```
使用者@DESKTOP-VU8PI8V MINGW64 ~  
$
```

4.範例

輸入想輸出的文字

```
使用者@DESKTOP-VU8PI8V MINGW64 ~  
$ echo "Hello World"  
Hello World  
  
使用者@DESKTOP-VU8PI8V MINGW64 ~  
$ printf "Hello World\n"  
Hello World
```

輸出數字 0 開始 | 直到 1000

```
使用者@DESKTOP-VU8PI8V MINGW64 ~  
$ while ((i<1000))  
> do  
> echo $i  
> let i++  
> done  
0  
1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16
```

輸出數字直到 i 變成 100

```
使用者@DESKTOP-VU8PI8V MINGW64 ~  
$ for ((i=0; i<100;i++))  
> do  
> echo $i  
> done  
0  
1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17
```

5.心得

原本在做之前，比較想用 Webgoat，但是安裝的過程太過於複雜，所以選擇 Git Bash，而在第一次接觸 Git Bash 後，我發現到這個十分方便，且 Git Bash 也可以使用 Linux 的指令來操作 Windows 十分的有趣，希望在之後的課程能夠更加熟悉的使用 Git Bash，且學到更多有趣的指令。