

Heroku

簡介與申請與環境設置教學



Heroku簡介

- ◆ Heroku 是一個平台即服務(PaaS)的雲端應用程式平台，也可以想成是「雲端上的網頁空間」。
- ◆ 各個開發者或是開發團隊可以自行在Heroku平台開發&佈署各種網站。好處是可以減少維護管理系統底層的成本，也透過簡單的介面來調整所需要的硬體設備等級或是資源。
- ◆ PaaS的平台不少，但是Heroku操作相當簡單，可以讓開發者初期專心的開發程式，而不用煩惱管理機器或是其他維護。
- ◆ Heroku 這家公司提供了與 GAE(Google App Engine) 類似的 PaaS 雲端服務，以及提供 PostgreSQL 這套關聯式資料庫和其他許多的外掛服務。它的底層硬體資源是採用了可擴展的 Amazon EC2，透過 Heroku 的 dynos (它的資源單位) 只要拉上拉下就可以增加伺服器負載了。
- ◆ 不同於傳統網頁空間的計費方式，Heroku 是按照你所使用的資源來計費的，像是 CPU、RAM 使用越多就需要付越多錢，不過免費的方案已經夠一般應用程式使用了。
- ◆ 當然如果網站隨著上線後流量變大，那麼Heroku就不一定適合了。但對於早期的開發者是非常適合的。

註冊與登入

- ◆ 建立Heroku帳號與登入：<https://www.heroku.com/>



The Heroku homepage features a prominent illustration of a city skyline with skyscrapers and industrial elements like pipes and tanks. In the foreground, there's a large, stylized building block structure with a sunburst effect above it. To the right, there's a forest and some clouds.

HEROKU Products Marketplace Pricing Documentation Support More

Log in or Sign up

NEW

Heroku Postgres PGX: New Plans, More Possibilities

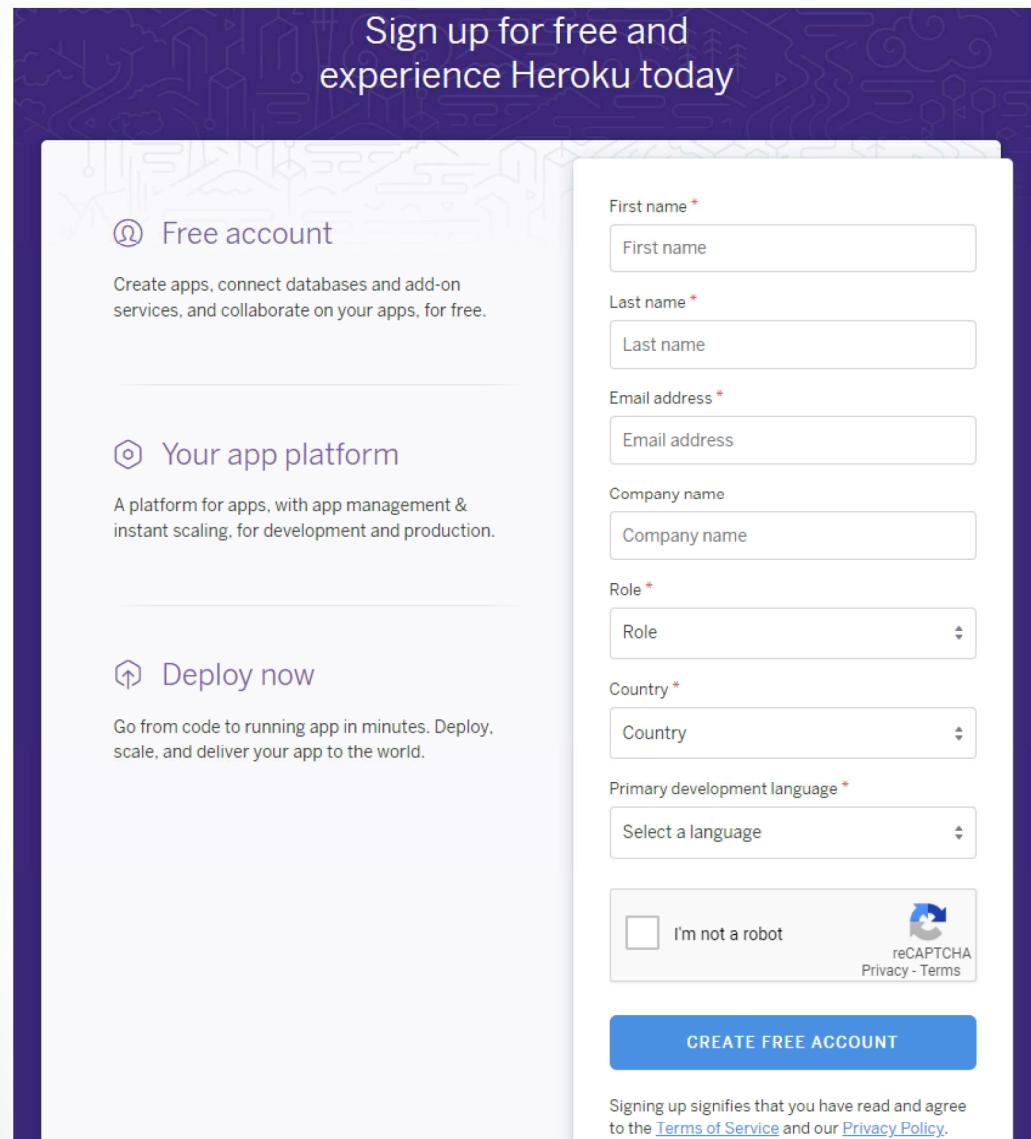
We're excited to announce our new lineup of Postgres production plans. PGX Plans introduce larger database sizes, more generous resource allocations, and a broader set of options to suit your needs and to help your applications scale more smoothly.

[SIGN UP FOR FREE](#)

[Explore Heroku Postgres](#)
[Read our Announcement](#)

註冊與登入

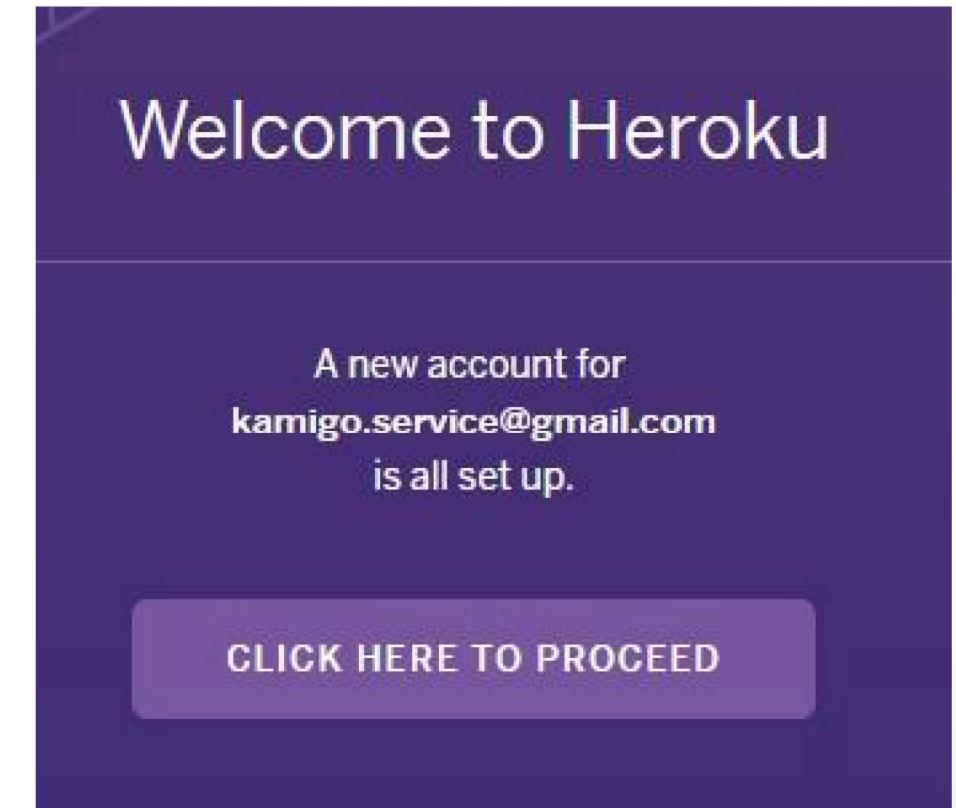
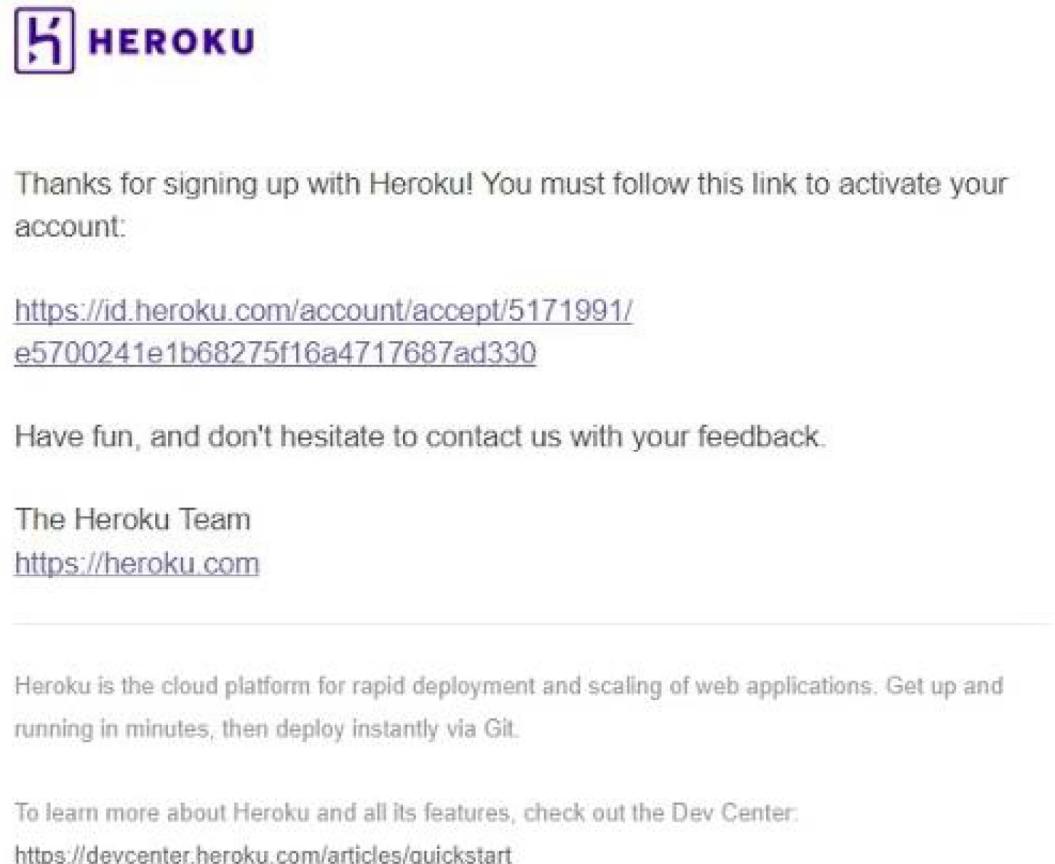
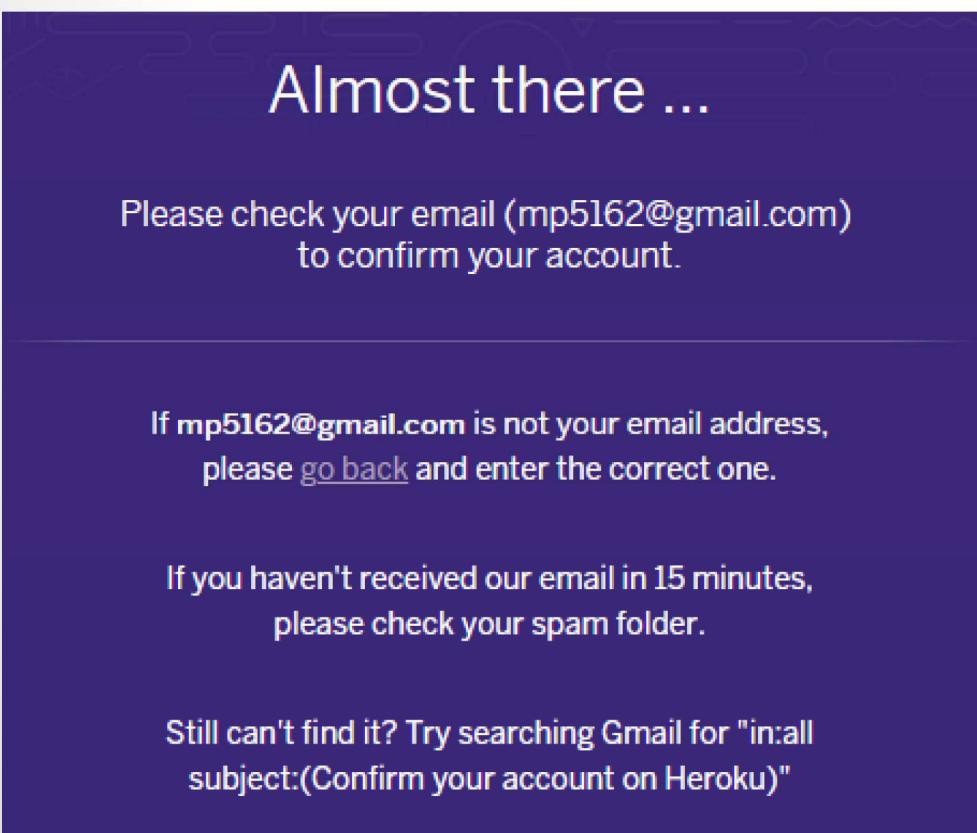
◆ 建立Heroku帳號與登入：<https://www.heroku.com/>。填好之後就按 CREATE FREE ACCOUNT。



- First name : 名字
- Last name : 姓氏
- Email Address : 信箱
- Role : 職業
- Primary Development Language : 主要開發語言

註冊與登入

- ◆ 註冊後會有如下訊息：「請去收信，謝謝」，這個步驟是為了確認你真的有你所填的信箱。
- ◆ 這裡要輸入兩次密碼，密碼最少要 8 個字，輸入完之後按下 SET PASSWORD AND LOG IN。
- ◆ 他說：「歡迎光臨」，這裡點一下 CLICK HERE TO PROCEED。



註冊與登入

- ◆ 這是平常登入會看到的畫面，他提供了各種教學參考。
- ◆ 我們平常不會透過這個後台去操作，而是使用 Heroku 提供的 Heroku CLI 來操作他。

The screenshot shows the Heroku Dashboard interface. At the top, there's a navigation bar with the Heroku logo, a search bar labeled "Jump to Favorites, Apps, Pipelines, Spaces...", and user profile icons. Below the navigation is a sidebar with "Personal" and "New" buttons. The main content area features a large "Getting Started with Heroku" section with a purple hexagonal icon. It includes a sub-section for "If you're new to Heroku, choose your language and follow our Getting Started guide to create a new app." Below this, there are eight language-specific "Get Started" buttons: Ruby (red gem icon), PHP (blue PHP icon), Node.js (green JS icon), Python (yellow Python icon), Java (brown Java icon), Go (light blue Go icon), Clojure (green Clojure icon), and Scala (orange Scala icon). At the bottom of this section is a purple "Create New App" button.

建立服務

- ◆ New>Create new app
- ◆ 選填服務名稱建立服務：服務名稱App Name 可自行輸入，或是由系統隨機產生。(最多5個，超過要付費)

The screenshot shows the Heroku dashboard interface. On the left, there's a sidebar with a user icon labeled "Personal" and a "Create new app" button. The main area displays a message: "You don't have any apps yet" and "Every app and pipeline you create or become a collaborator on will appear here". Below this is another "Create new app" button. On the right, a "Create New App" form is open. It includes fields for "App name" (with a placeholder "app-name") and "Choose a region" (with options for "United States" and "Europe", where "United States" is selected). There's also a "Create app" button at the bottom of the form. A red dashed box highlights the "New" button in the top navigation bar.

環境建置

◆ 下載並安裝Heroku CLI (Command Line Interface)

◆ 皆以安裝程式預設選項，進行安裝步驟

The screenshot shows the Heroku web interface for the application 'murmuring-caverns-94370'. At the top, there's a navigation bar with links for Overview, Resources, Deploy, Metrics, Activity, Access, and Settings. Below the navigation, there's a section titled 'Add this app to a pipeline' with instructions on how to create a new pipeline or add it to an existing one. It also mentions that Pipelines let you connect multiple apps together and promote code between them, and that Pipelines connected to GitHub can enable review apps and create apps for new pull requests. A dropdown menu labeled 'Choose a pipeline' is shown. Further down, there's a 'Deployment method' section with options for Heroku Git, GitHub, and Container Registry, all associated with the Heroku CLI. At the bottom, there's a link to 'Deploy using Heroku Git' and a note about using git in the command line.

Command Line > The Heroku CLI

The Heroku CLI

Last updated 17 August 2018

Table of Contents

- Download and install
- Other installation methods
- Verifying your installation
- Getting started
- Staying up to date
- Useful CLI plugins
- CLI architecture
- Troubleshooting
- Uninstalling the Heroku CLI

Download and install

The Heroku CLI requires Git, the popular version control system. If you don't already have Git installed, complete the following before installing the CLI:

- Git installation
- First-time Git setup

macOS

[Download the installer](#)

Also available via Homebrew:

```
$ brew install heroku/brew/heroku
```

Windows

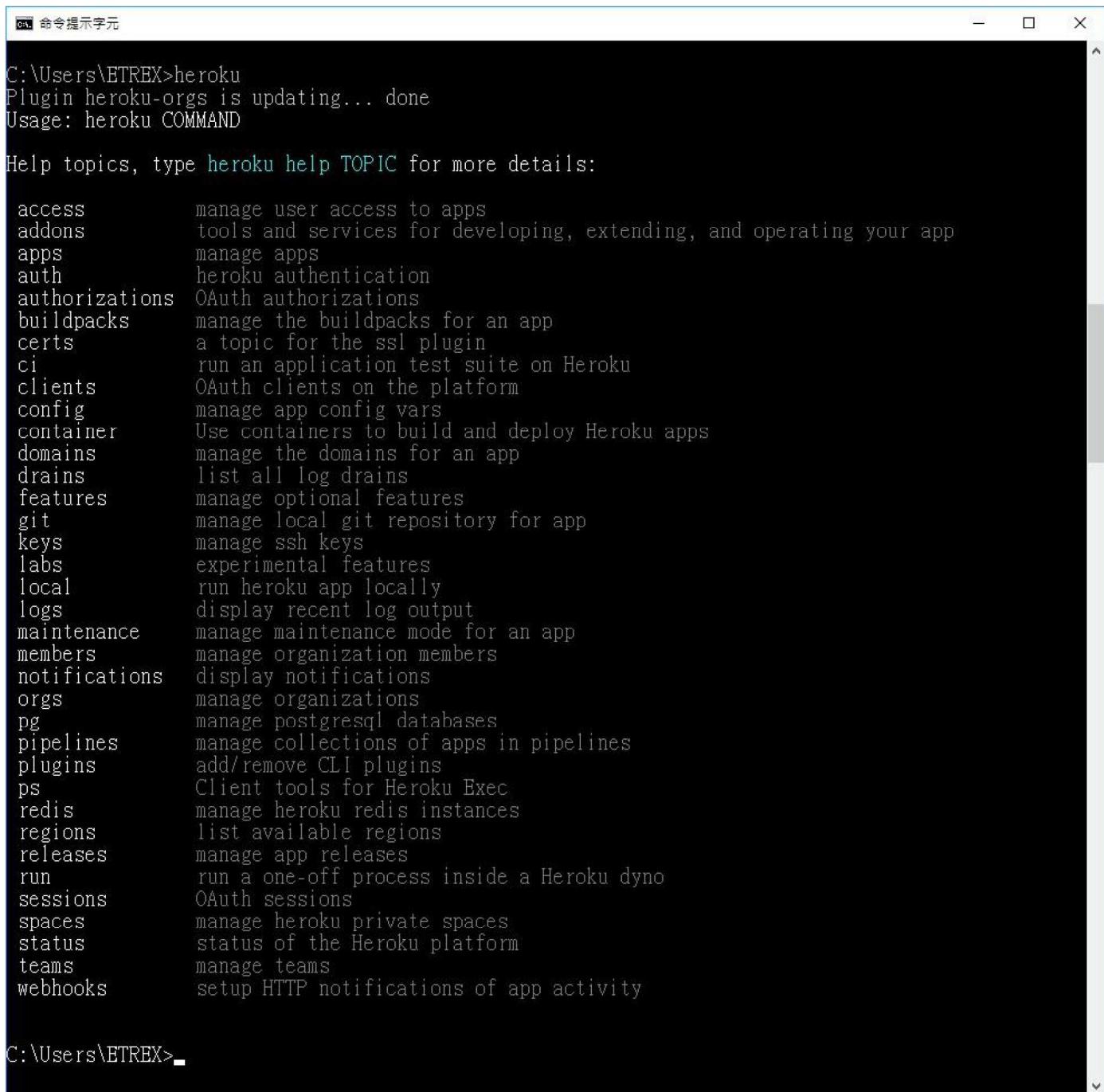
Download the appropriate installer for your Windows installation:

[64-bit installer](#)

[32-bit installer](#)

試用 Heroku CLI

- ◆ 開啟「命令提示字元」(可搜尋/執行 cmd)，開啟一個小黑框，輸入 heroku(大小寫均可)。



```
C:\Users\ETREX>heroku
Plugin heroku-orgs is updating... done
Usage: heroku COMMAND

Help topics, type heroku help TOPIC for more details:

access      manage user access to apps
addons       tools and services for developing, extending, and operating your app
apps        manage apps
auth         heroku authentication
authorizations OAuth authorizations
buildpacks   manage the buildpacks for an app
certs        a topic for the ssl plugin
ci           run an application test suite on Heroku
clients     OAuth clients on the platform
config       manage app config vars
container   Use containers to build and deploy Heroku apps
domains     manage the domains for an app
drains       list all log drains
features    manage optional features
git          manage local git repository for app
keys         manage ssh keys
labs         experimental features
local        run heroku app locally
logs         display recent log output
maintenance manage maintenance mode for an app
members     manage organization members
notifications display notifications
orgs        manage organizations
pg          manage postgresql databases
pipelines   manage collections of apps in pipelines
plugins     add/remove CLI plugins
ps          Client tools for Heroku Exec
redis       manage heroku redis instances
regions     list available regions
releases    manage app releases
run         run a one-off process inside a Heroku dyno
sessions    OAuth sessions
spaces      manage heroku private spaces
status      status of the Heroku platform
teams       manage teams
webhooks    setup HTTP notifications of app activity

C:\Users\ETREX>
```

試用 Heroku CLI

- ◆ 輸入 heroku login 進行登入，輸入e-mail信箱。接著輸入密碼
- ◆ 出現「Logged in as 你的heroku帳號」，代表登入成功畫面。

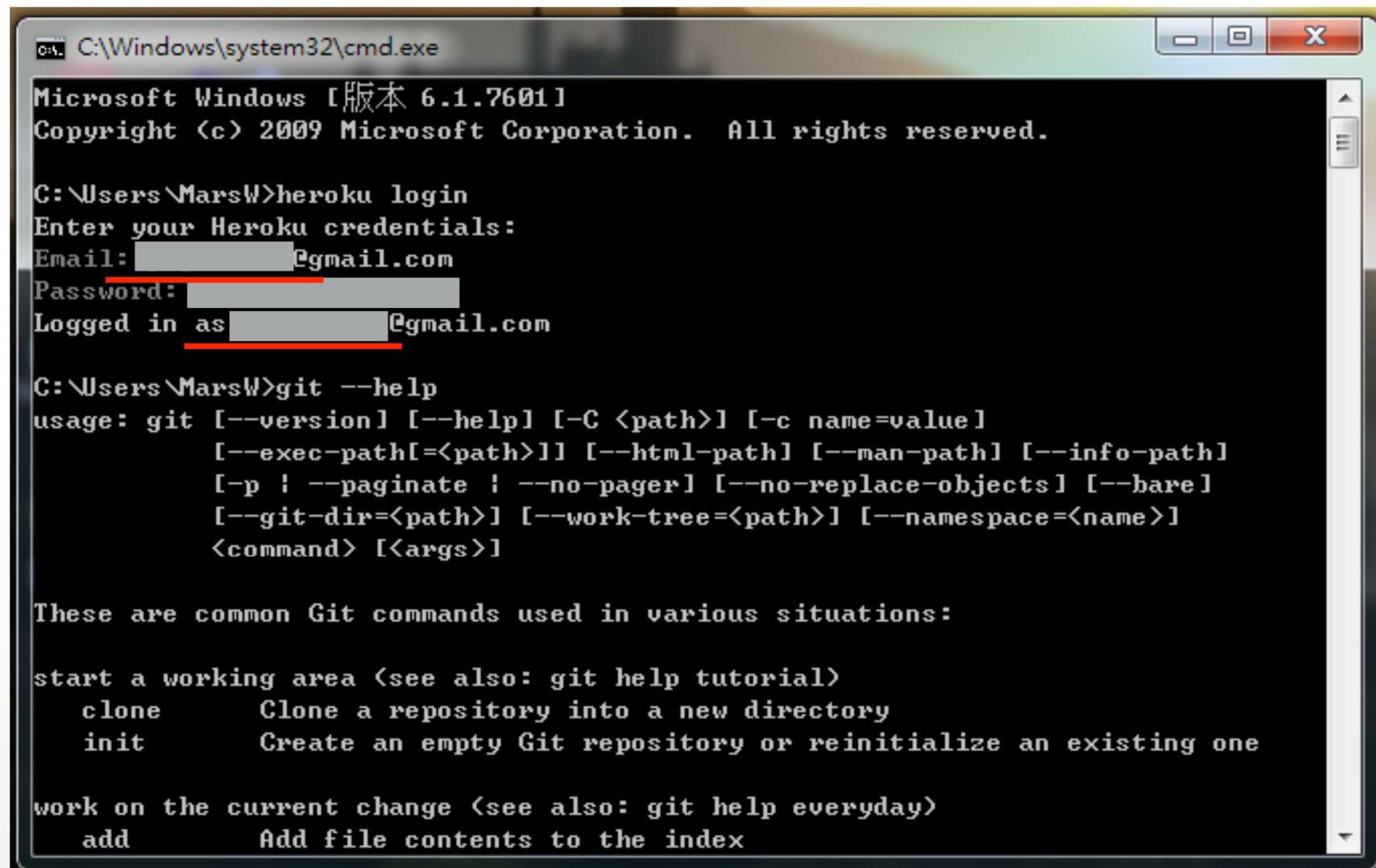
```
命令提示字元 - heroku login  
C:\Users\ETREX>heroku login  
Enter your Heroku credentials:  
Email: [REDACTED]
```

```
命令提示字元 - heroku login  
C:\Users\ETREX>heroku login  
Enter your Heroku credentials:  
Email: [REDACTED]@gmail.com  
Password:
```

```
命令提示字元  
C:\Users\ETREX>heroku login  
Enter your Heroku credentials:  
Email: [REDACTED]@gmail.com  
Password: [REDACTED]  
Logged in as [REDACTED]@gmail.com  
C:\Users\ETREX>
```

環境建置 確定安裝成功

- ◆ 接著輸入 git -- help，有跑出以下畫面，代表成功。
- ◆ Heroku 的初始設定到這裡就算完成了，接下來我們安裝 Git。



```
C:\Windows\system32\cmd.exe
Microsoft Windows [版本 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\MarsW>heroku login
Enter your Heroku credentials:
Email: [REDACTED]@gmail.com
Password: [REDACTED]
Logged in as [REDACTED]@gmail.com

C:\Users\MarsW>git --help
usage: git [--version] [--help] [-C <path>] [-c name=value]
           [--exec-path[=<path>]] [--html-path] [--man-path] [--info-path]
           [-p | --paginate | --no-pager] [--no-replace-objects] [--bare]
           [--git-dir=<path>] [--work-tree=<path>] [--namespace=<name>]
           <command> [<args>]

These are common Git commands used in various situations:

start a working area (see also: git help tutorial)
  clone      Clone a repository into a new directory
  init       Create an empty Git repository or reinitialize an existing one

work on the current change (see also: git help everyday)
  add        Add file contents to the index
```

Q & A