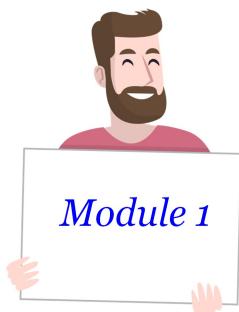




# 安裝Node.js



designed by freepik

Estimated time:

50 min.

資訊工業策進會 Institute for Information Industry

【Key Points】：

# 學習目標

- 1-1: 下載並安裝
- 1-2: 確認安裝版本
- 1-3: 執行測試的 JavaScript 檔



1-1

 財團法人資訊工業策進會  
INSTITUTE FOR INFORMATION INDUSTRY

Node.js 簡介

如何安裝 Node.js

安裝 VS Code

編寫簡單的程式

透過 Node.js 執行 JavaScript 檔案。

【Key Points】：

如何安裝 Node.js

編寫簡單的程式

執行 JavaScript 檔案

# 1-1：下載並安裝

- **Node.js簡介**
- **版本的生命周期**
- **如何選版本？**
- **Windows安裝Node.js**



designed by freepik



designed by freepik

1-2

 財團法人資訊工業策進會  
INSTITUTE FOR INFORMATION INDUSTRY

2008年，Google 推出 Chrome 瀏覽器，並且使用V8引擎來來解析 JavaScript。V8引擎滿足了 Ryan Dahl 對高性能Web伺服器的想法與要求。最終，Ryan Dahl 成功地將V8引擎移植到伺服器端，開發出 Node.js 。

Node.js 使用事件驅動、非阻塞 I/O 模型，輕量且效能卓著。

除了簡介 Node.js，我們還會學到 Node.js 的 LTS 與 Current 版本有何不同以及如何選擇。

最後，說明一遍實際在Windows安裝Node.js的過程。

<Note>

- Current版本 – 測試、嚐新
- LTS版本 – 適用於產品執行環境

【Key Points】：

Ryan Dahl 成功地將V8引擎移植到伺服器端，開發出 Node.js  
LTS 與 Current 版本有何不同以及如何選擇  
在Windows安裝Node.js的過程

# Node.js簡介

- Node.js 於 2009 年寫成，作者是 Ryan Dahl。



<https://zh.wikipedia.org/w/index.php?title=Node.js&oldid=2808011>  
File:Ryan\_Dahl.jpg

1-3

 財團法人資訊工業策進會  
INSTITUTE FOR INFORMATION INDUSTRY

Ryan Dahl 是一位專門幫忙客戶解決系統效能問題的專家。

他發現，Web伺服器在用量尖峰時，總伴隨效能問題。

他曾嘗試使用 Ruby、C、Lua 解決問題，可惜最終都失敗了。Ruby 的效能達不到要求；Lua 也因為同步I/O導致無法發揮效能優勢。C語言的效能雖高，但是要全部用 C 語言來寫 Web API 之類的伺服端程式，似乎不太合乎現實。

經過不斷地嘗試，Ryan Dahl 認為解決問題的關鍵，得透過事件驅動與異步I/O來完成。

2008年，Google 推出 Chrome 瀏覽器，並且使用V8引擎來來解析 JavaScript。V8引擎滿足了 Ryan Dahl 對高性能Web伺服器的想法與要求。最終，Ryan Dahl 成功地將V8引擎移植到伺服器端，開發出 Node.js。

## 【Key Points】：

Node.js 的作者是 Ryan Dahl。

事件驅動

異步I/O

# Node.js特性

- Node.js 是一個架構在Chrome V8引擎上的JavaScript執行環境。
- Node.js 使用事件驅動、非阻塞 I/O 模型，輕量且效能卓著。
- Node.js 能讓開發人員以 JavaScript 打造網頁伺服器程式。



1-4



2008年，Google 推出 Chrome 瀏覽器，並且使用V8引擎來解析 JavaScript。V8引擎滿足了 Ryan Dahl 對高性能Web伺服器的要求。

2009 年，Ryan Dahl 把V8引擎移植到伺服器端，並且發表了 Node.js。

在 Node.js 出現之前，JavaScript 通常只用於用戶端瀏覽器內的網頁程式設計。

Node.js 出現之後，JavaScript 可寫作伺服器端程式。剛開始，Node.js 只能在 Linux 系統執行。

在 v0.6.0 版本之後，Node.js 可跨平台在 Windows、Linux 及 Mac OS X 系統運行。

Node.js 具有下列特性：

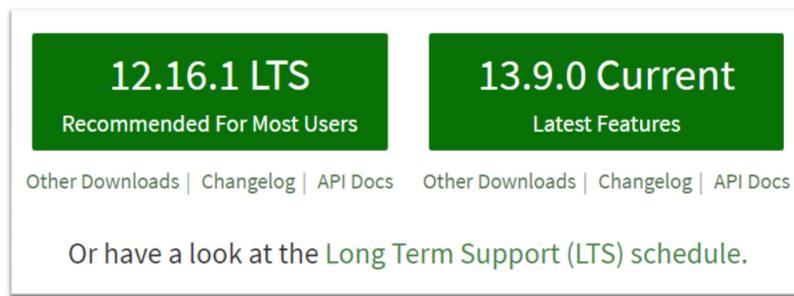
- ✓ 得益於V8引擎，效能卓著。
- ✓ Node.js 採用異步(非阻塞) I/O，不會有因為同步(阻塞) I/O 導致效能急遽下降的問題。
- ✓ JavaScript 固有的事件驅動、閉包等語言特性，使用起來非常方便。

## 【Key Points】：

- ✓ Node.js 採用異步(非阻塞) I/O，不會有因為同步(阻塞) I/O 導致效能急遽下降的問題。
- ✓ JavaScript 固有的事件驅動、閉包等語言特性，使用起來非常方便。
- ✓ Node.js 可跨平台在 Windows、Linux 及 Mac OS X 系統運行

## 版本介紹

- Node.js 使用 Long Term Support ( LTS ) 來規劃版本發佈  
( 官網 <https://nodejs.org/en/> 截圖如下 )
- LTS : 是 Long-Term Support 的縮寫，代表 Node.js 長期支持的版本 ( 版本號為單數 ) 。
- Current : 最新的 Node.js 版本系列 ( 版本號為偶數 ) 。



1-5

 財團法人資訊工業策進會  
INSTITUTE FOR INFORMATION INDUSTRY

Node.js 目前採用 Long Term Support ( LTS ) 策略來規劃版本發佈事宜。

因應 Node.js 的版本更迭，Node.js 的版本發佈使用了兩條主要的發佈路線：每年 4 月發佈偶數版本（具有最新功能的最新版），此版本也兼為未來 LTS 版的候選版本。經過半年的測試之後，同年 10 月發佈奇數版本（長期維護的版本）。

Node.js 的更新有其固定的節奏。主要的考量點是在保證相容性的前提下進行發佈，而不是累積了多少功能特性就改版。

最終，Node.js 的版本發佈變得有條不紊，開發者也有清楚的選擇依據。（下一張投影片會有更詳盡的說明）

Node.js 官網建議大部分的開發人員安裝使用 LTS 版。

【Key Points】：

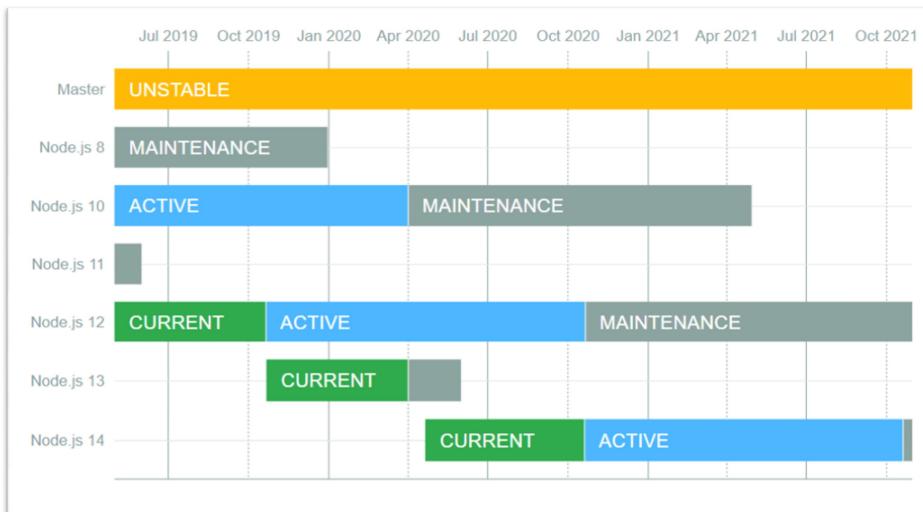
Node.js 目前採用 Long Term Support ( LTS ) 策略來規劃版本發佈事宜

每年 4 月發佈最新，10 月發佈長期維護的版本

Node.js 官網建議大部分的開發人員安裝使用 LTS 版

# 版本的生命周期

- Node.js 版本更新時程及狀態  
( <https://nodejs.org/en/about/releases/> )
- Node.js 中 LTS 和 Current 的有什麼區別？
- Node.js LTS 計劃是什麼？



1-6

 財團法人資訊工業策進會  
INSTITUTE FOR INFORMATION INDUSTRY

先來了解幾個名詞：

- ✓ **Current**：最新的 Node.js 版本，各項功能都還在測試期，不一定穩定可靠。
- ✓ **Active LTS (活躍期)**：正在積極維護和升級的版本，持續進行功能改進、除錯以及修補安全漏洞。
- ✓ **Maintenance LTS (維護階段)**：功能穩定的LTS版，在軟體生命週期終止之前，只會進行錯誤修復和安全漏洞補強。軟體生命週期終止之後，便轉為EOL版本(End of Life)。進入到EOL後的版本，便不再進行更新維護。

有關於 Node.js 版本的生命週期：

1. 每年的四月份，發佈最新的 Current 版本（偶數）並作為下一個長期維護(LTS)的候選版本。接下來，開始為期六個月的測試期。
2. 經過了六個月，時序來到了十月，發行LTS版（奇數），宣告一個長期維護版進入其活躍期（Active）。
3. 活躍期(Active)的LTS版為期18個月，這18個月會持續進行功能改進、除錯以及修補安全漏洞。
4. 活躍期的18個月過後，進入Maintenance期，成為功能穩定的LTS版。接下來的12個月，只會進行錯誤修復和安全漏洞補強。
5. Maintenance 期的12個月時間到，就會進入EOL (End of Life)狀態，功成身退。

【Key Points】：

每年的四月份，發佈最新的 Current 版本（偶數）

十月，發行LTS版（奇數）

LTS版有30個月的效期 (18+12)

# 如何選版本？

- 進入官網(<https://nodejs.org/en/>)下載LTS版本。
- 下載完畢後，執行安裝程式開始安裝，若無特殊需求，可直接按Next到底。
- 另外有各種客製化安裝，會有較多的彈性與選擇(<https://nodejs.org/en/download/>)。

The screenshot shows the Node.js download page. At the top, it says "Node.js® is a JavaScript runtime built on Chrome's V8 JavaScript engine." Below that, a green box says "New security releases now available for all release lines". Under "Download for Windows (x64)", there are two buttons: "12.16.1 LTS Recommended For Most Users" (highlighted with a red box) and "13.9.0 Current Latest Features". At the bottom, there are links for "Other Downloads | Changelog | API Docs" and "Other Downloads | Changelog | API Docs".

選擇LTS版本進行安裝

The screenshot shows the Node.js download page with two main sections: "LTS Recommended For Most Users" and "Current Latest Features". Under "LTS", there are links for "Windows Installer", "macOS Installer", and "Source Code". Under "Current", there are links for "Windows Binary (.zip)", "macOS Binary (.pkg)", "Linux Binaries (x64)", "Linux Binaries (ARM)", and "Source Code". A table below shows file sizes and architectures:

	32-bit	64-bit
Windows Binary (.zip)	32-bit	64-bit
macOS Binary (.pkg)	64-bit	64-bit
Linux Binaries (x64)	64-bit	64-bit
Linux Binaries (ARM)	ARMv7	ARMv8
Source Code	node-v12.16.1.tar.gz	

另提供各種客製化安裝

1-7



## Current版本 – 測試、嚐新

- 適合於想要嚐試一些新功能，或者想實做一些實驗性題目的開發人員。
- 由於功能穩定與相容性無法提供任何保證，請謹慎使用，切勿用於產品執行環境。

## LTS版本 – 適用於產品執行環境

- 產品執行環境最重要的是穩定。
- 可以考慮在每年發佈Active LTS版本的時候跟進升級。
- 每個Active LTS版本會有為期18個月的軟體改進支援，再加上後來的 Maintenance LTS 12個月，在一個版本退出歷史舞台前，會有 30 個月的時間，長期維護程式臭蟲與安全漏洞修訂。

### 【Key Points】：

測試、嚐新: Current版本  
產品執行環境: LTS版本  
不知道怎麼選: LTS版本

# Windows 安裝 Node.js

## • 步驟一：下載 / 選擇合適版本進行安裝操作

The screenshot shows two windows side-by-side. On the left is the Windows System Settings window, specifically the 'About' section, which displays system information including processor (Intel Core i3-4160 CPU @ 3.60GHz), RAM (8.00 GB), and system type (64-bit). A red box highlights the 'System Type' entry. On the right is the Node.js official download page. It features a 'Downloads' section with tabs for 'LTS' (selected) and 'Current'. Under 'LTS', there are links for Windows (msi), macOS (pkg), and source code (tar.gz). Under 'Current', there are also links for Windows (msi), macOS (pkg), and source code (tar.gz). A red box highlights the '64-bit' link under the Windows section.

1-8

III 財團法人資訊工業策進會  
INSTITUTE FOR INFORMATION INDUSTRY

以Windows 系統為例，在我們連上 Node.js 官網首頁後，預設是下載64位元版的長期支援版(LTS)或最新版(Current)。

點按上 Node.js 官網首頁上方的「DOWNLOADS」，Windows 作業系統的使用者可進一步選擇32位元或64位元版本。

目前的電腦的作業系統，大多都已是64位元。

如果想確定你的電腦是32或64位元，以 Windows 10 為例：

1. 先點一下螢幕桌面左下角的「開始」
2. 點按「設定」按鈕 (看起來像齒輪的那個小圖示)
3. 找到並點按「系統」選項
4. 在左側的側邊欄點按「關於」

其實，還有一個方法：就直接下載 64 位元版回來安裝。如果是32位元的作業系統，安裝程式會發現無法安裝，報出錯 y 誤訊息後，安裝程序也就準備結束。

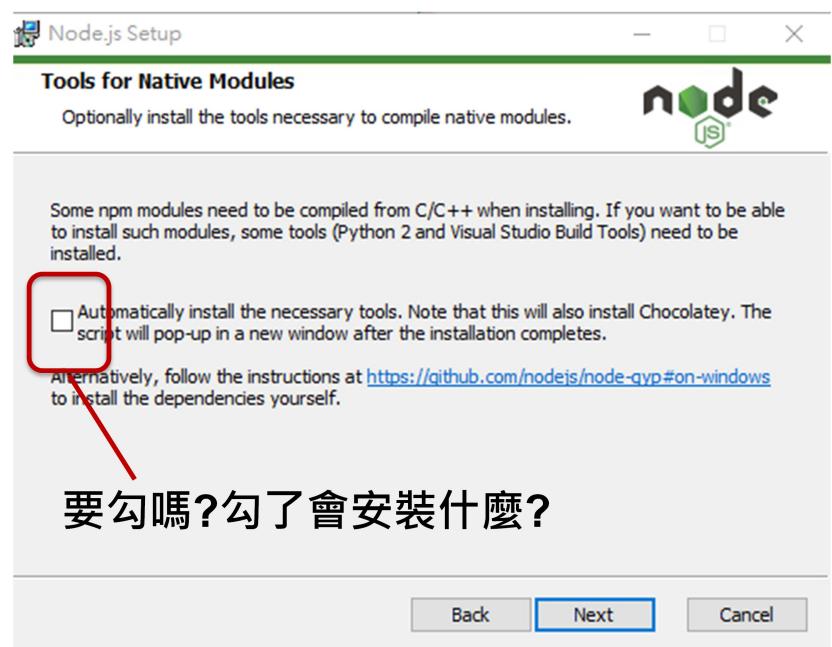
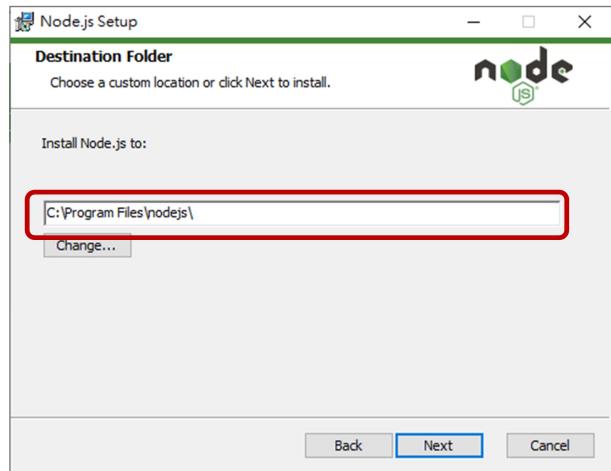
### 【Key Points】：

下載安裝程式

預設是下載64位元版

如何檢查Windows的電腦是32或64位元

- 步驟二：下載完後，接著下一步，選擇對您要的應目錄後安裝。
- 步驟三：依預設安裝完成。



1-9

確認下載版本後，執行安裝程式：

1. 點選下一步
2. 勾選擬同，同意條款，進行下一步
3. 安裝目錄若無特別需求，請按預設建議
4. 接著繼續按下一步，可依預設安裝完成。

【請注意】若勾選擬投影片右側的那個核選方塊，會額外安裝兩個工具，建構工具(Python 2和Visual Studio)、Chocolatey。建議先按照預設值，不要勾選擬額外加選的功能以節省安裝時間，此外，本課程不會用到加選的功能

5. 按下install，執行安裝
6. 等待安裝完成
7. 按下finish 關閉視窗，完成安裝。

### 【Key Points】：

執行安裝程式

建議學生先按照預設值，不要勾選擬額外加選的功能以節省安裝時間

本課程不會用到加選的功能

## 1-2：確認安裝版本

- 開啟「命令提示字元」
- 使用「node -v」確認安裝版本
- Node.js更新改版
- NPM是什麼？



designed by freepik



designed by freepik

1-10

 財團法人資訊工業策進會  
INSTITUTE FOR INFORMATION INDUSTRY

CMD是Windows「命令提示字元」的程式檔名(cmd.exe)，作用是Windows系統下的命令視窗，你可以透過這個視窗，向作業系統下達指令。

我們會詳細說明如何開啟「命令提示字元」的各種方式。

然後，在「命令提示字元」以 node -v，查看Node.js版本並確認Node.js已正確安裝。

接下來，會說明到日後如果要昇級改版，該怎麼做。

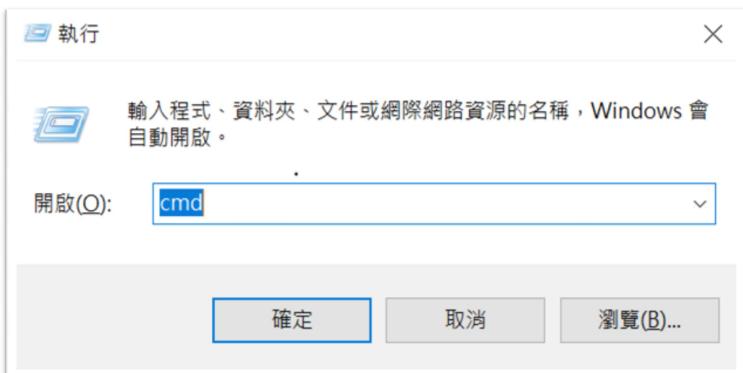
最後則是說明NPM套件管理程式。

### 【Key Points】：

如何開啟「命令提示字元」  
以 node -v 查看 Node.js 版本  
說明 NPM 套件管理程式

# 開啟「命令提示字元」

- 按下組合鍵「Windows + R」，輸入「cmd」後，點按「確定」按鈕。
- 按下組合鍵「Windows + S」，然後鍵盤輸入「cmd」
- 或者滑鼠點按桌面左下角的「開始」，然後鍵盤輸入「cmd」



1-11

1. 按下組合鍵「Windows + R」
2. 輸入「cmd」後，點按「確定」按鈕。

或者，滑鼠點按桌面左下角的「開始」，然後鍵盤輸入「cmd」，也可找到「命令提示字元」。

或者，按下組合鍵「Windows + S」，然後鍵盤輸入「cmd」，可找到「命令提示字元」。

CMD是Windows命令提示字元的程式檔名 (cmd.exe)，作用是 Windows 系統下的命令視窗，你可以透過這個視窗，向作業系統下達指令。

## 【Key Points】：

你可以透過CMD視窗，向作業系統下達指令。

開啟CMD之各種方式

CMD之基礎介紹

# 使用「node -v」確認安裝版本

- 開啟「命令提示字元」
- 輸入：`node -v`，查看 Node.js 版本

```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [版本 10.0.18362.657]
(c) 2019 Microsoft Corporation. 著作權所有，並保留一切權利。
C:\Users\User>node -v
v12.16.1 → 版本號
C:\Users\User>
```

1-12

- 開啟「命令提示字元」，輸入「`node -v`」並且按下 Enter 鍵。
- 正常的話，畫面會顯示例如：v 12.16.1 的版本號碼，表示我們成功安裝了 Node.js 的 12.16.1 版本。

如果出現「不是內部或外部命令、可執行的程式或批次檔」，表示Node.js沒有安裝或者軟體安裝設定不正確（例如，環境變數path未包括Node.js的安裝目錄）。

<Note>

另外一個確認「Node.js有沒有安裝」的作法：透過 `where` 指令。

- 開啟「命令提示字元」，輸入「`where node`」並且按下 Enter 鍵。
- 正常的話，畫面會顯示Node主程式的檔名（含路徑），預設是 C:\Program Files\nodejs\node.exe。

【Key Points】：

命令提示字元

`node -v`

`where node`

## Node.js更新改版

- 重新下載新版本的 msi 安裝程式
- 安裝到先前的同一個資料夾，即可覆蓋/更新之前的版本。
- 如需檢查之前安裝 node 的路徑，請執行：  
**where node**

1-13



1. 先查一下目前這版的 Node.js 安裝在哪裡
2. 關於步驟1，在命令提示字元執行 **where node** 指令可查出來
3. 重新下載新版本的 msi 安裝程式
4. 安裝到 Node.js 先前安裝的同一個資料夾
5. 相關套件例如 NPM 也會一併更新

【Key Points】：

**where node**

安裝到同一個資料夾

相關套件例如 NPM 也會一併更新

# NPM是什麼？

- **NPM = Node.js Package Manager**
- **Node.js 的套件管理程式**
- **為什麼需要套件管理程式？**
- **操作語法：**  
**npm install 套件模組的名稱**

1-14



NPM 是 Node.js Package Manager 的簡寫。

顧名思義，NPM 是 Node.js 的套件管理程式。

為什麼需要套件管理程式呢？請想想看，如果你聽說有一個叫 express 的模組非常好用，你可能會有下列疑問？

- 要到哪裡抓檔案？不會遇到釣魚網站吧？
- 該安裝到哪一個資料夾？該注意什麼設定選項？
- 這個套件有沒有什麼相依套件是應該預先下載安裝呢？
- express 模組日後如何昇級改版？
- 如何解除安裝？

很簡單，下一個指令就好了 `npm install express`

本課程後來會談到 NPM，如有興趣進一步了解，請參閱 npm官網文件說明：

<https://docs.npmjs.com/>

## 【Key Points】：

顧名思義，NPM 是 Node.js 的套件管理程式

為什麼需要套件管理程式

`npm install express`

## 1-3：執行測試的 JavaScript 檔

- 進入Node模式
- 撰寫 JavaScript 檔
- 執行 JavaScript 檔
- 如何不透過 JavaScript 檔案來執行程式



designed by freepik



designed by freepik

1-15

 財團法人資訊工業策進會  
INSTITUTE FOR INFORMATION INDUSTRY

Node.js 安裝好了，如何執行 JavaScript 程式呢？

我們會學到如何進入Node模式，在Node 模式採「一次一行」、「一問一答」的方式執行程式。

接下來，會學習如何在「命名提示字元」，輸入「node 檔名」指令來執行 JavaScript 程式

此外，本課程建議安裝 VS Code，以它為整合開發環境。我們會學到如何在 VS Code 啟動終端機視窗，以「node 檔名」執行 JavaScript 程式。

最後，會介紹一種直接執行 JS 程式的作法：

```
node -e "console.log('Hello World')"
```

### 【Key Points】：

如何進入Node模式

在「命名提示字元」，輸入「node 檔名」指令來執行 JavaScript 程式

```
node -e "console.log('Hello World')"
```

# 進入Node模式

- Node 模式採「一次一行」、「一問一答」的方式執行程式。
- 在「命令提示字元」輸入：  
**node <Enter鍵>**  
即可進入Node模式。
- 輸入你想執行的 JavaScript 程式，例如：  
`console.log("Hello");`
- 輸入：**.exit** 或組合鍵「**Ctrl + D**」可結束 Node 模式。
- 輸入：**.help** 可查看 Node 模式相關指令。

1-16



Node 模式採「一次一行」、「一問一答」的方式執行程式。

在「命令提示字元」輸入下列指令，可進入 Node 模式

`node <Enter鍵>`

輸入你想執行的 JavaScript 程式，例如：

`console.log("Hello");`

<Note>因為 `console.log()` 方法並沒有傳回值。程式執行時，除了顯示 Hello，還會在第二列顯示 undefined 字樣。

輸入 `.exit` 或組合鍵「**Ctrl + D**」可結束 Node 模式。

輸入 `.help` 可查看 Node 模式相關指令。

【Key Points】：

何謂 Node 模式採

如何進入 Node 模式

如何結束 Node 模式

# 撰寫 JavaScript 檔

- 本課程建議使用 VS Code，安裝程式下載網址：  
<https://code.visualstudio.com/>
- 安裝 VS Code 時，建議勾選下列選項：  
將「以Code開啟」動作加入 Windows 檔案總管目錄的操作功能表中
- 1. 在檔案總管，以滑鼠右鍵點按例如 c:\lab 資料夾 | 以 Code 開啟
- 2. 在 VS Code 左側的 Explorer 窗格，  
滑鼠右鍵 | New File | 輸入檔名，例如: hello.js
- 3. 編寫 JavaScript 程式（如下），並且存檔（組合鍵 Ctrl + S）  
`console.log("Hello!");`

1-17



你可以使用任何文字編輯器編寫 JavaScript 程式。

本課程建議使用 VS Code，安裝程式下載網址：

<https://code.visualstudio.com/>

安裝 VS Code 時，建議勾選下列選項：

- 將「以Code開啟」動作加入 Windows 檔案總管檔案的操作功能表中
- 將「以Code開啟」動作加入 Windows 檔案總管目錄的操作功能表中
- 加入 PATH 中

撰寫 JavaScript 檔：

1. 在檔案總管，滑鼠右鍵點按例如 c:\lab 資料夾 | 以 Code 開啟
2. 在 VS Code 左側的 Explorer 窗格，滑鼠右鍵 | New File，檔名: Hello.js
3. 編寫 JavaScript 程式（如下），並且存檔（組合鍵 Ctrl + S）  
`console.log("Hello!");`

<Note> 以本例來說，JavaScript 程式存檔於 c:\lab\hello.js

【Key Points】：

本課程建議使用 VS Code

安裝 VS Code 時，建議勾選「以Code開啟」選項

請示範一下建立檔案的過程

# 執行 JavaScript 檔

1. 開啟「命名提示字元」

2. 輸入: node 檔名 (含路徑) , 例如:

node c:\lab\hello.js

• 承接上一張投影片，透過 VS Code 編輯、執行 JavaScript 程式：

1. 在檔案總管，以滑鼠右鍵點按例如 c:\lab 資料夾 | 以 Code 開啟

2. 在 VS Code , 按下「 Ctrl + 撇號 」 ( 撇號在鍵盤的左上角 )

3. 輸入 node hello.js

1-18



1. 開啟「命名提示字元」

2. 輸入: node 檔名 (含路徑) , 例如:

node c:\lab\hello.js

<Note>承接上一張投影片，如果你已將 JavaScript 程式存檔於 c:\lab\hello.js

1. 在檔案總管，以滑鼠右鍵點按例如 c:\lab 資料夾 | 以 Code 開啟

2. 在 VS Code , 按下「 Ctrl + 撇號 」 ( 撇號在鍵盤的左上角 ) 啟動終端機視窗

3. 輸入 node hello.js

<Note> 建議安裝 VS Code , 以它為整合開發環境。

【Key Points】：

在命名提示字元，輸入: node 檔名 (含路徑)

在 VS Code , 按下「 Ctrl + 撇號 」啟動終端機視窗

建議安裝 VS Code , 以它為整合開發環境

# 如何不透過 JavaScript 檔案來執行程式？

- 在 VS Code 按下「 Ctrl + 撇號 」開啟終端機，輸入指令
  - node -e "console.log('Hello World')"
  - 使用上述指令，將可以達到如同使用JavaScript檔案結果



```
輸出 終端機 值錯主控台 問題 1: powershell
PS C:\Users\ada\Desktop\node.js> node -e "console.log('Hello World')"
Hello World
PS C:\Users\ada\Desktop\node.js>
```

1-19



如何不透過JavaScript檔案執行程式？

先在 VS Code，按下「 Ctrl + 撇號 」（撇號在鍵盤的左上角）啟動終端機視窗

接下來，輸入下列程式：

```
node -e "console.log('Hello World')"
```

請注意，字串若以單引號開始，就以單引號結尾；雙引號開始，雙引號結束，不可交錯使用。

因此，如果引號的內容有引號的話，內外層應該分別使用對應的引號。以下列指令來說，外層用一對雙引號，內層便改用一對單引號：

```
node -e "console.log('Hello World')"
```

## 【Key Points】：

在 VS Code，按下「 Ctrl + 撇號 」啟動終端機視窗

協助學員了解基本 node -e 指令使用方式

單雙引號要正確配對

# Summary〈精華回顧〉

- **Node.js簡介**
- **Node.js使用Long Term Support ( LTS )來規劃版本發布**
- **版本的生命周期**
- **Windows安裝Node.js**
- **使用「node -v」確認安裝版本**
- **Node 模式採「一次一行」、「一問一答」的方式執行程式**
- **撰寫 JavaScript 檔**
- **執行 JavaScript 檔**



1-20

 財團法人資訊工業策進會  
INSTITUTE FOR INFORMATION INDUSTRY

Ryan Dahl 成功地將V8引擎移植到伺服器端，開發出 Node.js。Node.js 使用事件驅動、非阻塞 I/O 模型，輕量且效能卓著。

除了簡介 Node.js，我們還學到 Node.js 的 LTS 與 Current 版本有何不同以及如何選擇。並且，實際看到在 Windows 安裝 Node.js 的過程。

接下來，在「命令提示字元」以 node -v，查看 Node.js 版本並確認 Node.js 已正確安裝。

好了，Node.js 已安裝妥當，如何執行 JavaScript 程式呢？

我們學到如何進入 Node 模式，在 Node 模式採「一次一行」、「一問一答」的方式執行程式。

也學習到如何在「命名提示字元」，輸入「node 檔名」指令來執行 JavaScript 程式。

最後，我們學到如何在 VS Code 啟動終端機視窗，以「node 檔名」執行 JavaScript 程式。

## 【Key Points】：

在 Windows 安裝 Node.js

確認 Node.js 已正確安裝

在「命名提示字元」，輸入「node 檔名」指令來執行 JavaScript 程式

# 線上程式題

- **1-1 請問安裝 Node.js 之後，如何確認版本？**  
請問安裝 Node.js 之後，如何確認版本？
- **1-2 請問如何查看npm版本？**  
請問如何查看npm版本？
- **1-3 如何以 Node.js 執行 app.js 程式？**  
如何以 Node.js 執行 app.js 程式？

1-21



題目一: 1-1 請問安裝 Node.js 之後，如何確認版本？

難易程度: 易

請問安裝 Node.js 之後，如何確認版本？

題目二: 1-2 請問如何查看npm版本？

難易程度: 中

請問如何查看npm版本？

題目三: 1-3 如何以 Node.js 執行 app.js 程式？

難易程度: 難

如何以 Node.js 執行 app.js 程式？

請閱讀教學系統的程式與說明，然後，到「// 作答區」填入答案。

答案有區分大寫小寫。

## 【Key Points】：

1-1 請問安裝 Node.js 之後，如何確認版本？

1-2 請問如何查看npm版本？

1-3 如何以 Node.js 執行 app.js 程式？

# 課後練習 (Lab)

- **1. 下載 Node.js**
  - 1-1. 進入官網: <https://nodejs.org>
  - 1-2. 選擇左邊 LTS 的版本(建議)下載
- **2. 安裝**
  - 2-1. 執行 Node.js 安裝程式
  - 2-2. 依照安裝流程，使用預設便可
- **3. 確認是否安裝成功，並查看版本號**
  - 3-1. VS code 打開終端機
  - 3-2. 先查看 Node 的版本號，輸入以下指令
  - **\$ node -v or \$ node --version**

Estimated time:

**20 min.**

1-22



進入官網: <https://nodejs.org>  
選擇左邊 LTS 的版本(建議)下載  
執行 Node.js 安裝程式  
依照安裝流程，使用預設便可  
輸入以下指令查看 Node.js 的版本號:  
`node -v`

## 【Key Points】：

下載 LTS 版本的 Node.js  
安裝 Node.js  
以 `node -v` 指令，確認 Node.js 已安裝妥當

# 範例程式使用說明

- 本課程各模組之範例程式，使用步驟如下：
  1. 安裝 Node.js
  2. 安裝 Visual Studio Code
  3. 以 Visual Studio Code 開啟本模組的範例資料夾
  4. 在 Visual Studio Code 按下「Ctrl + 滑鼠右鍵」開啟 terminal 終端機視窗。
  5. 在終端機視窗，輸入 node 檔案名稱.js
- 本模組的主題在於安裝 Node.js，暫時沒有範例程式。

1-23



使用步驟：

1. 安裝 Node.js  
<https://nodejs.org/en/>
2. 安裝 Visual Studio Code  
<https://code.visualstudio.com/>
3. 以 Visual Studio Code 開啟範例資料夾  
在檔案總管，滑鼠右鍵點按「本範例資料夾」，從快捷功能表點選「以Code開啟」  
或者，啟動 Visual Studio Code 之後，功能表 File | Open Folder，選擇「本範例資料夾」
4. 在 Visual Studio Code 按下「Ctrl + 滑鼠右鍵」開啟 terminal 終端機視窗。
5. 在終端機視窗，輸入 node 檔案名稱.js

【Key Points】：

程式執行環境 Node.js

程式開發編輯環境：Visual Studio Code

在 Visual Studio Code 按下「Ctrl + 滑鼠右鍵」開啟 terminal 終端機視窗，輸入：「node 檔名.js」