

# 第1次作業題目-作業-HW1

學號：112111113

姓名：林品承

作業撰寫時間：160 (mins，包含程式撰寫時間)

最後撰寫文件日期：2025/10/17

本份文件包含以下主題：(至少需下面兩項，若是有多者可以自行新增)

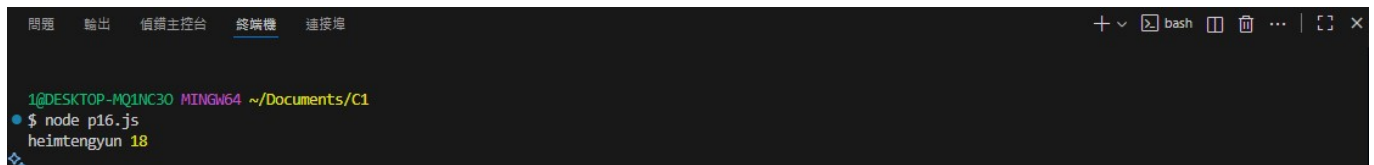
- ☒ 說明內容
- ☒ 個人認為完成作業須具備觀念

1. 請撰寫Topic 1的P. 16，並□1□ 寫程式後印出結果，該程式名為p16.js，與□2□ 承□1□程式碼中的 Line3 下中斷點，並印出此時變數age裡面的值為多少觀察。

Ans:

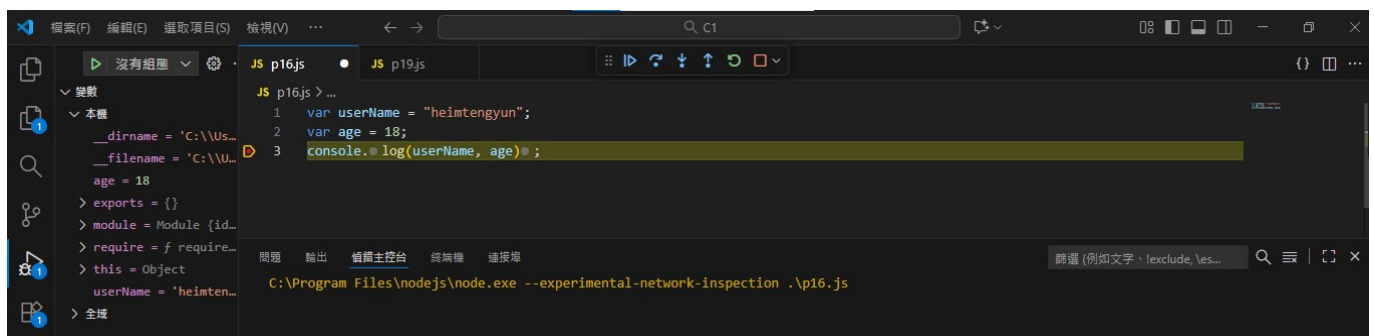
## (1) 撰寫程式後印出結果

```
var userName = "heimtengyun"; // 宣告一個變數叫 userName，並給予值為字串  
"heimtengyun"  
var age = 18; // 宣告一個變數 age，並給予值為數字 18  
console.log(userName, age); // 將 userName 和 age 的值輸出
```



```
1@DESKTOP-MQ1NC30 MINGW64 ~/Documents/C1  
$ node p16.js  
heimtengyun 18
```

## (2) 在Line3下中斷點，並印出此時變數age裡面的值為多少觀察



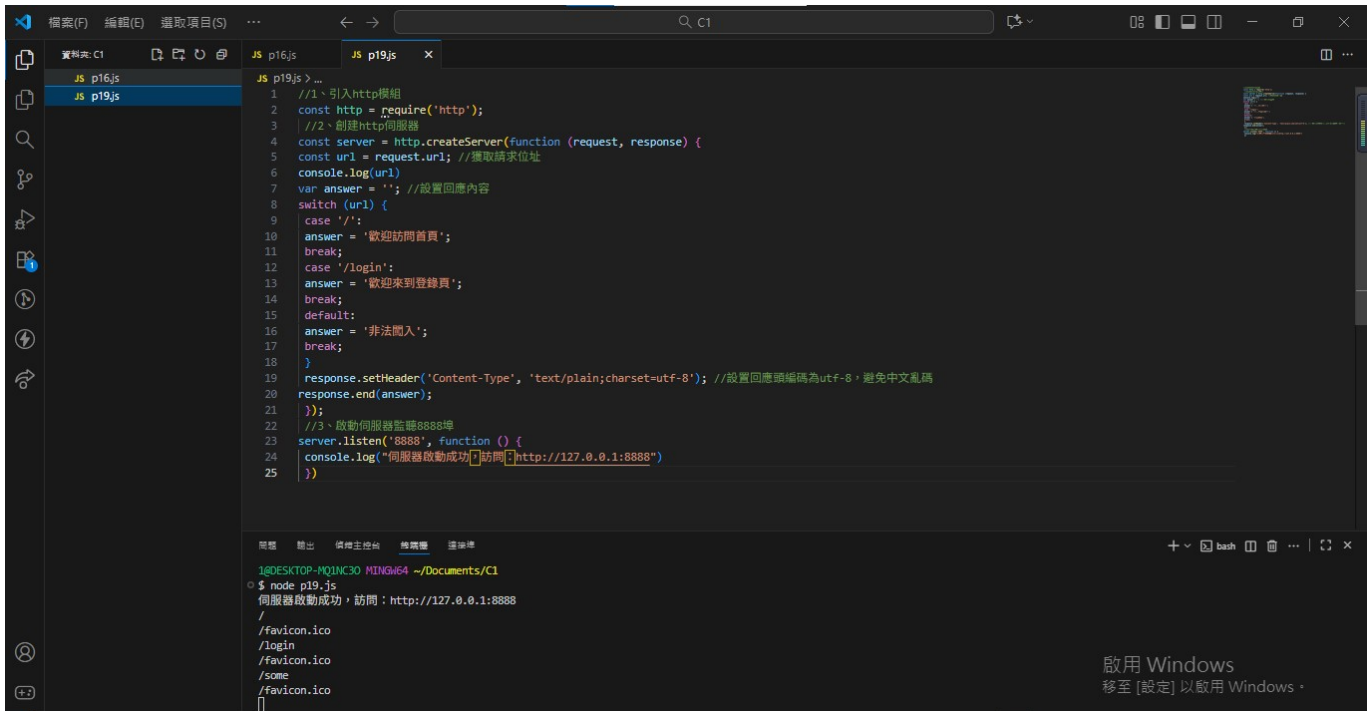
- 點擊該行左側旁邊的空白處，會顯示一個紅色圓點，並可下中斷點。
- 點擊左側工具列的「執行與除錯」圖示，選擇「Node.js」作為執行環境，即開始執行。
- 最後可在左側的「變數」面板中查看 age 的值，或將滑鼠移到程式碼中的 age 上方，也會顯示目前的值。
- 顯示結果，變數 age 值為18。

2. 請撰寫Topic 1的P. 19，並□1□ 寫程式後印出結果，該程式名為p19.js，與□2□ 請同樣印出p. 20的三種結果，並說明為何可以造成該種原因

Ans:

(1) 撰寫程式後印出結果

```
//1、引入http模組
const http = require('http');
//2、創建http伺服器
const server = http.createServer(function (request, response) {
const url = request.url; //獲取請求位址
console.log(url)
var answer = ''; //設置回應內容
switch (url) {
case '/':
answer = '歡迎訪問首頁';
break;
case '/login':
answer = '歡迎來到登錄頁';
break;
default:
answer = '非法闖入';
break;
}
response.setHeader('Content-Type', 'text/plain;charset=utf-8'); //設置回應頭編碼為
utf-8，避免中文亂碼
response.end(answer);
});
//3、啟動伺服器監聽8888埠
server.listen('8888', function () {
console.log("伺服器啟動成功，訪問：http://127.0.0.1:8888")
})
```



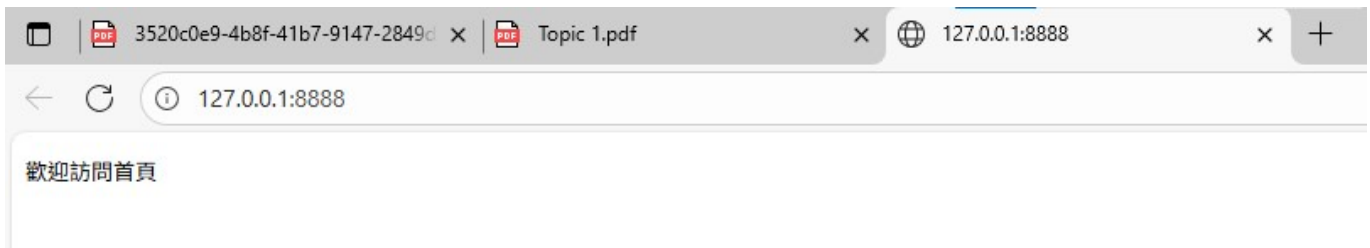
```
JS p19.js > ...
1 //1、引入http模組
2 const http = require('http');
3 //2、創建http伺服器
4 const server = http.createServer(function (request, response) {
5   const url = request.url; //獲取請求地址
6   console.log(url)
7   var answer = ''; //設置回應內容
8   switch (url) {
9     case '/':
10      answer = '歡迎訪問首頁';
11      break;
12     case '/login':
13      answer = '歡迎來到登錄頁';
14      break;
15     default:
16      answer = '非法闖入';
17      break;
18   }
19   response.setHeader('Content-Type', 'text/plain;charset=utf-8'); //設置回應編碼為utf-8，避免中文亂碼
20   response.end(answer);
21 });
22 //3、啟動伺服器監聽8888埠
23 server.listen('8888', function () {
24   console.log("伺服器啟動成功！訪問http://127.0.0.1:8888")
25 })
```

```
$ node p19.js
伺服器啟動成功，訪問：http://127.0.0.1:8888
/
/favicon.ico
/login
/favicon.ico
/some
/favicon.ico
```

## (2) 說明為何可以造成該種原因

因為前面有定義當收到請求時要執行的函式，並取得使用者請求的 url 路徑。

因此在switch (url) 這裡就可以設定不同的回應內容，且根據變數 url 的值來判斷要執行哪個 case。



如果 url 是 '/'，則執行這一個 case，如上圖。



如果 url 是 '/login'，則執行這一個 case，如上圖。



如果 url 的值都不是上述兩者，則會執行 default，也就是「非法闖入」，如上圖。

因為 url 的值不在預期範圍內都不符合任何 case，所以會落入 default；

又或是大小寫或格式錯誤、變數沒設定好或來源錯誤等原因。

---

## 須具備觀念

(1)需知道在程式碼中如何去下中斷點，善用執行與偵錯的功能，去排解或了解關於程式的執行端相關訊息。

(2)在撰寫關於創建web伺服器的這種程式碼時，要注意像設定的回應內容時，類型需要為純文字，並使用 UTF-8 編碼，才能確保中文不會亂碼；以及在設定switch (url)的變數時，要注意大小寫格式、變數是否設定正確、整體邏輯是否設定正確等因素，否則都會被視為非法。