目錄

[箭頭函數 1](#_Toc491731515)

[模組化物件 1](#_Toc491731516)

[this指標 1](#_Toc491731517)

[JS使用原則 1](#_Toc491731518)

[JSON 1](#_Toc491731519)

[JSON與JS物件轉換 2](#_Toc491731520)

[REQUIRE 2](#_Toc491731521)

[回乎函數 (CallBack Function) 2](#_Toc491731522)

[檔案系統-FS 2](#_Toc491731523)

# 箭頭函數

( ) => {…} no argument

x => {…} one argument

(x,x) => {…} several argument

greet = () =>console.log(‘L’);

greet();

# 模組化物件

module.exports 建立模組化物件

require 使用模組化物件

Node.js提供許多原生模組

# this指標

物件方法中的this---物件的方法是被物件呼叫

函數之中的this---函數是被系統呼叫

# JS使用原則

物件的方法使用正規函數宣告，this就會指向物件

物件的方式裡面的函數使用箭頭函數宣告，this就會指向物件

(wek3 p.24 picture)

一個類別繼承另一個類別，子類別的建構子內就必須用super關鍵字呼叫父類別的建構子

# JSON

資料格式

JavaScrip Object Notation(JSON)

宣告方式

設aaa=bbb

在JSON寫成

“aaa”:”bbb”

所有name與字串都要用雙引號” ”包起來

JSON資料型態支援:String, Number, Objects, Arrays, Booleans, null

# JSON與JS物件轉換

JSON→JS

JSON.parse(…)

把JSON轉成JS

---------------------------

JS→JSON

JSON.stringify(…)

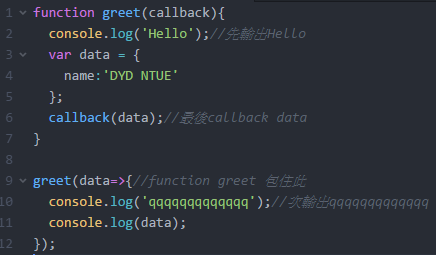
把JS轉成JSON

# REQUIRE

Require會自動把JSON資料轉成JS物件

(wk3 p.51)

# 回乎函數 (CallBack Function)



# 檔案系統-FS

Synchronous:同一時間只能執行一個動作

Asynchronous:同一時間只能執行多個動作



.readFileSync:同步檔案處理

.readFile:非同步檔案處理

.readFile在背景執行，當程式結束後會回傳data，有如callback功能