為你自己學 系列課程

JavaScript 入門

課程代碼 JS101

五倍紅寶石 ASTRO

ES6 / ES2015 語法補充

字串與變數的組合

```
let name = "悟空"
let age = 18

// 用 + 號串接
console.log("大家好，我的名字是" + name + "，我今年" + age + "歲")

// 把變數帶進去
console.log(`大家好，我的名字是${name}，我今年${age}歲`)
```

箭頭函數

```
const addNumber = function (a, b) {
  return a + b
}

addNumber(1, 2)
```

```
// 箭頭函數
const addNumber = (a, b) ⇒ {
  return a + b
}

addNumber(1, 2)
```

```
// 箭頭函數
const addNumber = (a, b) ⇒ a + b

addNumber(1, 2)
```

箭頭函數不是一般的 function 的簡寫

物件簡寫

```
const age = 18
const attack = () ⇒ console.log("使用大絕招！")

const hero = {
  name: "悟空",
  age: age,
  attack: attack,
}

hero.attack() // 執行 attack 函數
```

如果 key 的名字跟變數同名的話...

```
const age = 18
const attack = () ⇒ console.log("使用大絕招！")

const hero = {
  name: "悟空",
  age, // 寫一個代表就好
  attack, // 寫一個代表就好
}

hero.attack() // 執行 attack 函數
```

解構

```
const hero = {
  name: "悟空",
  age: 18,
  location: "地球",
}

const name = hero.name
const age = hero.age

console.log(name)
console.log(age)
```

```
const hero = {
  name: "悟空",
  age: 18,
  location: "地球",
}

// 解構
const { name, age } = hero

console.log(name)
console.log(age)
```

用在 function 的參數...

```
function printUser(userData) {
  const name = userData.name
  const age = userData.age

  console.log(name)
  console.log(age)
}

const user = {
  name: "悟空",
  age: 18,
}

printUser(user)
```

```
function printUser(userData) {
  const { name, age } = userData

  console.log(name)
  console.log(age)
}

const user = {
  name: "悟空",
  age: 18,
}

printUser(user)
```

```
function printUser({ name, age }) {
  console.log(name)
  console.log(age)
}

const user = {
  name: "悟空",
  age: 18,
}

printUser(user)
```

點占點占點占

展開

```
const comicHeroes = ["悟空", "魯夫", "娜美"]
const marvelHeroes = ["鋼鐵人", "索爾", "奇異博士"]
const allHeroes = comicHeroes.concat(marvelHeroes)

console.log(allHeroes)
```

```
const comicHeroes = ["悟空", "魯夫", "娜美"]
const marvelHeroes = ["鋼鐵人", "索爾", "奇異博士"]
const allHeroes = [...comicHeroes, ...marvelHeroes]

console.log(allHeroes)
```

剩下的我全收了！

```
function sayHello(user, others) {
  console.log(user)
  console.log(others)
}

sayHello("悟空", "魯夫", "娜美", "琦玉")
```

```
function sayHello(user, ...others) {
  console.log(user)
  console.log(others)
}

sayHello("悟空", "魯夫", "娜美", "琦玉")
```

解構的時候也可以用

```
const heroes = ["悟空", "魯夫", "娜美", "琦玉"]

// 解構
const [h1, h2] = heroes

console.log(h1) // 印出 悟空
console.log(h2) // 印出 魯夫
```

```
const heroes = ["悟空", "魯夫", "娜美", "琦玉"]

// 解構
const [h1, ...h2] = heroes

console.log(h1) // 印出 悟空
console.log(h2) // 印出 ['魯夫', '娜美', '琦玉']
```