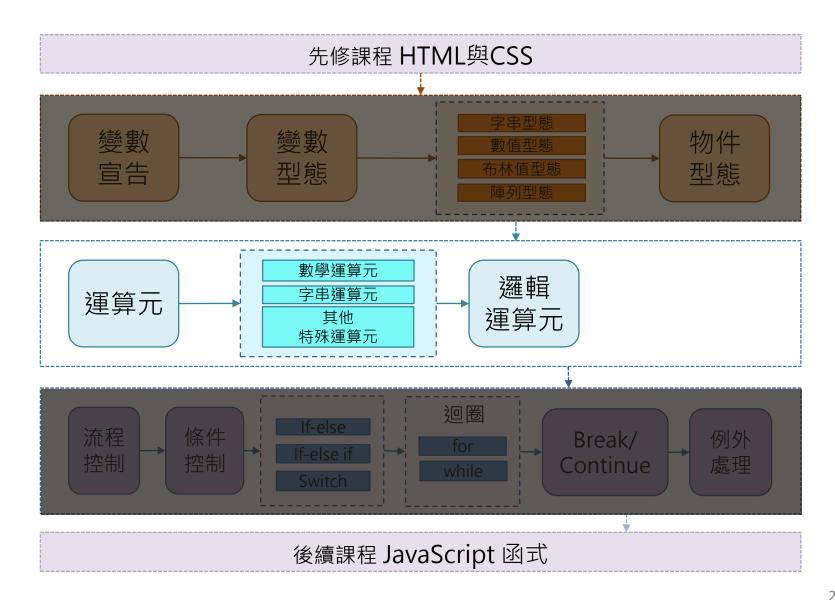
## 運算子 JavaScript

難易度★☆☆

#### 課程地圖



### 學習目標

1 定義運算元與運算子
 2 運 算 子 種 類

## 定義運算元與運算子

#### 運算元與運算子定義

運算子(operator): 運算元(operand): 運算式 執行運算動作的人 拿來被運算的人 運算子 運算元 (可以是數字或變數)

# 電腦一次只能執行一個運算前面的計算會影響到後面的結果

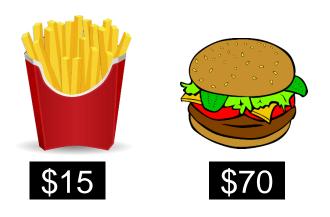
#### 例題應用

#### 把文字拆成幾句敘述,最後用程式碼寫出來

買3包薯條和6個漢堡

老闆需計算總金額

並告知客人要多少錢





#### 流程說明

#### 老闆要計算總價給客人

薯條 漢堡



15元



計算買3包薯條 6個漢堡

告知 價錢

















宣告 變數A,B A存放 數字**15**  B存放 數字**70**  再宣告變數buy 存放總金額 465 輸出在 網頁上 **465**元

 $(15 \times 3 + 70 \times 6 = 465)$ 



#### 運算元與運算子定義

讓我們把敘述寫成程式碼,訂單(



)表示流程跑到哪裡

薯條 漢堡





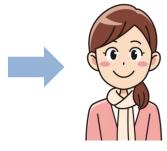
計算買3包薯條6個漢堡

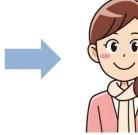
告知 價錢

















let A,B;

A = 15;

B = 70;

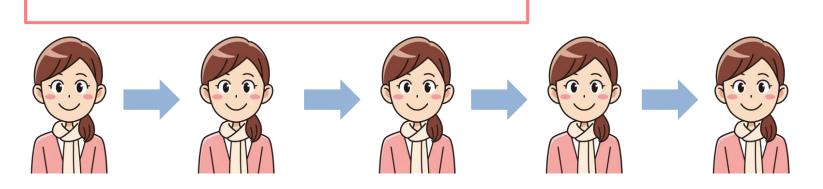
let buy = 3\*A+6\*B;

innerHTML = "共" + buy + "元";

宣告 變數A,B A存放 數字15 B存放 數字**70**  再宣告變數buy 存放總金額 465 輸出在網頁上 465元<sup>3</sup>

#### 組合程式碼

再將程式碼寫在一起,這就是最後我們要寫的東西



let A,B; A=15; B=70; let buy =3\*A+6\*B; innerHTML ="共" + buy + "元";

```
<!DOCTYPE HTML>
<html>
<head></head>
<body>
    <script>
        let A, B;
        A = 15;
        B = 70;
        let buy = 3 * A + 6 * B;
        document.body.innerHTML = "共" + buy + "元";
    </script>
</body>
</html>
```

```
<!DOCTYPE HTML>
<html>
                          HTML章回教過的網頁格式
<head></head>
<body>
    <script>
        let A, B;
        A = 15;
        B = 70;
                  3 * A + 6 * B;
        let buy =/
        document/.body.innerHTML = "共" + buy + "元";
   </script>
</body>
</html>
```

```
<!DOCTYPE HTML>
                          javascript語法務必
<html>
                          寫在<script>標籤內
<head></head>
<body>
    <script>
        let A, B;
       A = 15;
        B = 70;
        let buy = 3 * A + 6 * B;
        document.body.innerHTML = "共" + buy + "元";
    </script>
</body>
</html>
```

```
<!DOCTYPE HTML>
<html>
<head></head>
<body>
    <script>
                                於網頁中插入文字
        let A, B;
                                (可參考HTML章節內容)
       A = 15;
        B = 70;
       let buy = 3 * A +
       document.body.innerHTML = "共"
                                      + buy + "元";
    </script>
</body>
</html>
```



## 運算子種類

#### 運算子種類

- 算術運算子(Arithmetic Operators)
- 字串運算子(String Operators)
- 指定運算子(Assignment Operators)
- 比較運算子 (Comparison Operators)
- 邏輯運算子(Comparison Operators)

+a	正號	實際上無作用
-a	負號	將a正負變號
a + b	加法	a和b的總和
a - b	減法	a減去b
a * b	乘法	a乘上b
a/b	除法	a除以b
a % b	取餘數	a除以b的餘數(餘數的正負號與a相同)
++a	前置增值	a先+1,再回傳
a++	後置增值	a先回傳,再+1
<b></b> a	前置減值	a先-1,再回傳
a	後置減值	a先回傳,再-1

算術運算子-四則運算即加減乘除,是數學最基本的算術運算。





加

除

滅

餘數

乘

-1

+1

#### 正(+)、負(-)

+a	正號	實際上無作用
-a	負號	將a正負變號
a + b	加法	a和b的總和
a - b	減法	a減去b
a * b	乘法	a乘上b
a/b	除法	a除以b
a%b	取餘數	a除以b的餘數(餘數的正負號與a相同)
++2	前署幢信	a失+1,再同值

++a	前置增值	a先+1,再回傳
a++	後置增值	a先回傳,再+1
a	前置減值	a先-1,再回傳
a	後置減值	a先回傳,再-1

**-7** → **-7** 

正正得正

負負得正

負正得負

正負得負

$$+ (+9) \rightarrow 9$$

$$-(-1) \rightarrow 1$$

$$-(+3) \rightarrow -3$$

+ 
$$(-6) \rightarrow -6$$

$$a = 15$$
 $-a = -15$ 

$$b = -6$$
  $+ b = -6$   $- b = 6$ 

**+a** 

保持原樣

**-a** 

正變負、負變正

```
<!DOCTYPE HTML>
<html>
<head></head>
<body>
    <script>
        let a = 3;
        document.body.innerHTML = "+a = " + (+a);
        document.body.innerHTML += "<br/>';
        document.body.innerHTML += "-a = " + (-a);
    </script>
                                           顯示結果
</body>
                                             +a = 3
</html>
                                             -a = -3
```

#### 網頁中顯示的結果為

$$+a = 3$$
  
 $-a = -3$ 

```
<!DOCTYPE HTML>
<html>
<head></head>
<body>
    <script>
                                        運算完後得到3
       > let a = 3;
        document.body.innerHTML = "+a =
        document.body.innerHTML += "<br/>';換行
        document.body.innerHTML += "-a =
    </script>
                                        運算完後得到-3
</body>
</html>
```

#### 加(+)、減(-)、乘(\*)、除(/)、取餘數(%)

+a	正號	實際上無作用
-a	負號	將a正負變號
a + b	加法	a和b的總和
a - b	減法	a減去b
a * b	乘法	a乘上b
a/b	除法	a除以b
a % b	取餘數	a除以b的餘數(餘數的正負號與a相同)

++a	前置增值	a先+1,再回傳
a++	後置增值	a先回傳,再+1
a	前置減值	a先-1,再回傳
a	後置減值	a先回傳,再-1

a+b

加法

a-b

減法

```
<!DOCTYPE HTML>
<html><head></head>
<body>
    <script>
        let a = 3;
        let b = 20;
        document.body.innerHTML = "a+b = " +
        document.body.innerHTML += "<br/>';
        document.body.innerHTML += "a-b = " +
    </script>
</body>
</html>
```

a+b = 23 a-b = -17

乘法

a\*b

```
顯示結果
<!DOCTYPE HTML>
                                 a*b = 60
<html><head></head><body>
                                 a/b = 0.15
<script>
                                 a\%b = 3
    let a = 3;
    let b = 20;
    document.body.innerHTML = "a*b = "
    document.body.innerHTML += "<br/>>";
    document.body.innerHTML += "a/b = "
    document.body.innerHTML += "<br/>>";
    document.body.innerHTML += "a%b = "
</script>
</body></html>
```

a/b

除法

a%b

餘數

前置減值

後置減值

++ ` --

--a

a--

+a	正號	實際上無作用
-a	負號	將a正負變號
a + b	加法	a和b的總和
a - b	減法	a減去b
a * b	乘法	a乘上b
a/b	除法	a除以b
a % b	取餘數	a除以b的餘數(餘數的正負號與a相同)
++a	前置增值	a先+1,再回傳
a++	後置增值	a先回傳,再+1

a先-1,再回傳

a先回傳,再-1

讓我們先理解一下, 甚麼是加加(++)、減減(--) 在大學, 你在幫大家訂便當... 請問要訂幾個?



前面的訂便當事件我們可以知道

bentou=0 bentou

可寫成 $\rightarrow$  bentou = bentou + 1 (原來的值 $\frac{1}{1}$ 之後再放回)

1+1

或使用簡寫 > bentou ++

或是 ++ bentou

同理
 若我想將 bentou 的值減─
 可寫成→
 或使用簡寫→
 bentou = bentou - 1 (原來的值減1之後再放回)
 或使用簡寫→
 或是→
 -- bentou
 教們打入程式執行看看...

## 進階補充

++a

先加一,後傳值給innerHTML

```
顯示結果 -----
++a = 4
a = 4
```

a++

先傳值給innerHTML,後加一

顯示結果 ----a++=3 a=4

#### 顯示結果

--a = 2

a = 2

**--a** 

先減一,後傳值給innerHTML

顯示結果

a - - = 3

a = 2

**a--**

先傳值給innerHTML,後減一

## 隨堂小練習

柯B想要計算自己的BMI值

身高為180cm,體重為65kg

請寫一個程式幫助他計算!

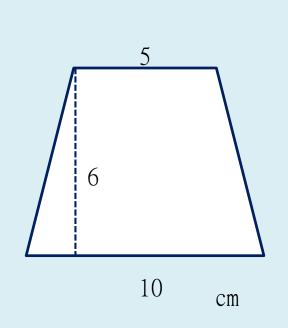
BMI的公式為:體重(公斤)/身高²(公尺²)



```
BMI = 20.061728395061728
```

```
<!DOCTYPE HTML>
<html>
<head></head>
<body>
<script>
    let KP height cm = 180.0;
    let KP weight = 65.0;
   //公分轉公尺
    let KP_height_m = KP_height_cm / 100;
    let BMI = KP_weight / (KP_height_m * KP_height_m);
    document.body.innerHTML = "BMI = " + BMI;
</script>
</body>
</html>
```

### 隨堂小練習



國小的小櫻在算數學,題目如下

求梯形面積

身為小櫻的表兄姊的你

決定寫一個程式幫助他解決這道題目

梯形面積公式為: (上底+下底)\*高/2)



### 顯示結果

梯形面積 = 45

```
<!DOCTYPE HTML>
<html>
<head></head>
<body>
<script>
    let up = 5;
    let down = 10;
    let height = 6;
    let area = (up + down) * height / 2;
    document.body.innerHTML = "梯形面積 = " + area;
</script>
</body>
</html>
```

## 字串運算子(String Operators)

定義:用來做一些與字串有關的工作

例如:將兩個字串連接,得出一個更長的字串

因為字串和數目型態都可以用 + 和 += , 所以如

果不留意變數的型態就很容出錯。

符號	用法	結果
+	a = "hello," + "world"	a="hello,world"

## 範例

目前的菜單(menu)上有

漢堡70元、薯條15元、奶昔20元

老闆想要增加菜單"冰淇淋10元"

請寫個程式



### **ANS**



\$15



\$70

### 目前的菜單用字串來表示為

let menu = "漢堡70元/薯條15元/奶昔20元/";

### 新增冰淇淋可以直接加進menu

menu = menu + "冰淇淋10元";

### 或存放在變數,再加進menu

let icecream = "冰淇淋10元"; menu = menu + icecream;



### 字串運算子

顯示結果

menue = 漢堡70元/薯條15元/奶昔20元/冰淇淋10元

a+b

字串連接

```
<!DOCTYPE HTML>
<html>
<head></head>
<body>
<script>
  let menue = "漢堡70元/薯條15元/奶昔20元/";
  let icecream = "冰淇淋10元";
  menue = menue + icecream;
  </script>
</body>
</html>
```

### 隨堂小練習

現有兩變數a="單槓"、b="鋼彈"

請輸出



### 顯示結果

單槓盪單槓 鋼彈盪鋼彈 單槓盪鋼彈

```
<!DOCTYPE HTML>
<html>
<head></head>
<body>
<script>
    let a = "單槓";
    let b = "鋼彈";
    document.body.innerHTML = a + "盪" + a + "<br/>>";
    document.body.innerHTML += b + "盪" + b + "<br/>>";
    document.body.innerHTML += a + "盪" + b;
</script>
</body>
</html>
```

## 隨堂小練習

請新增兩個變數

分別為湯、塔

試著寫出以下繞口令

和尚端湯上塔,塔滑湯燙,湯趙塔

再將變數的內容顛倒(即內容分別是塔、湯)

使輸出為

和尚端塔上湯,湯滑塔燙,塔燙湯

# ANS

### 顯示結果

和尚端湯上塔,塔滑湯燙,湯燙塔 和尚端塔上湯,湯滑塔燙,塔燙湯

```
<!DOCTYPE HTML>
<html>
<head></head>
<body>
<script>
   let a = "湯";
   let b = "塔";
   document.body.innerHTML = "和尚端" + a + "上" + b + "," + b
   + "滑" + a + "燙," + a + "燙" + b + "<br/>";
   a = "塔";
   b = "湯";
   document.body.innerHTML += "和尚端" + a + "上" + b + "," + b
   + "滑" + a + "燙" + a + "燙" + b;
</script>
</body>
</html>
```

### 指定運算子(Assignment Operators)

a += b	a = a + b
a -= b	a = a - b
a *= b	a = a * b
a /= b	a = a / b
a %= b	a = a % b

### 指定運算子(Assignment Operators)

a += b	a = a + b
a -= b	a = a - b
a *= b	a = a * b
a /= b	a = a / b
a %= b	a = a % b

### ,將計算完的結果存放在變數a

$$a = a - b$$

$$a = a * b$$

$$a = a / b$$

$$a = a \% b$$

```
a+b放入a
   a+=b
                                     a*b放入a
                                                 a*=b
    a-=b
                                                 a/=b
                                     a/b放入a
              a-b放入a
<!DOCTYPE HTML>
                                                a = 25
<html>
                                                a = 20
<head></head><body>
<script>
                                                 a = 100
    let a = 20, b = 5;
                                                a = 20
    a += b;
    document.body.innerHTML = "a = " + a + "<br/>";
   a -= b;
    document.body.innerHTML += "a = " + a + "<br/>>";
    a *= b;
    document.body.innerHTML += "a = " + a + "<br/>>";
    a /= b;
    document.body.innerHTML += "a = " + a ;
</script>
</body></html>
```

a == b	相等	如果a和b的值相等則為true
a === b	完全相等	如果a和b的值與型別都相等則為true 物件、陣列和函式必須是同一個物件才為true
a != b	不相等	如果a和b的值不相等則為true
a !== b	不完全相等	如果a和b的值或型別不相等則為true
a < b	小於	如果a小於b則為true
a > b	大於	如果a大於b則為true
a <= b	小於等於	如果a小於等於b則為true
a >= b	大於等於	如果a大於等於b則為true

a == b	相等	如果a和b的值相等則為true
a === b	完全相等	如果a和b的值與型別都相等則為true 物件、陣列和函式必須是同一個物件才為true
a != b	不相等	如果a和b的值不相等則為true
a !== b	不完全相等	如果a和b的值或型別不相等則為true
a < b	小於	如果a小於b則為true
a > b	大於	如果a大於b則為true
a <= b	小於等於	如果a小於等於b則為true
a >= b	大於等於	如果a大於等於b則為true

```
a == b 意思: a 等於 b 假設 a = 666 b = 666 c = 999 則 a == b 回傳 true a == c 回傳 false
```



a	>	b
u		U

$$a > = b$$

$$a < = b$$

```
a==b
```

a、b是否相等

a!=b

a、b是否不相等

```
<!DOCTYPE HTML>
<html>
<head></head>
<body>
    <script>
                                             將運算式()起來
        let a = 5;
        let b = 6;
        document.body.innerHTML = "a == b "
        document.body.innerHTML += "<br/>';
        document.body.innerHTML += "a != b "
    </script>
</body>
                                            顯示結果 -
</html>
                                            a == b false
                                            a!= b true
```

</body></html>

```
a<b
     a>b
                                          a是否小於b
                a是否大於b
    a>=b
                                                            a<=b
                a是否大於等於b
                                      a是否小於等於b
                                                        顯示結果
<!DOCTYPE HTML>
                                                        a = 5
<html><head></head><body>
                                                        b = 5
<script>
                                                        c = 6
   let a = 5;
                                                        a > b false
   let b = 5;
                                 > 為 > 的跳脫字元
                                                        a > = b true
   let c = 6;
                                 < 為 < 的跳脫字元
   document.body.innerHTML = "a = "+ a,
                                                        a < c true
   document.body.innerHTML += "b = "+/
                                                       a < = c true
   document.body.innerHTML += "c = "/+
   document.body.innerHTML += "a > /b " + (a > b) + " <br/> ";
   document.body.innerHTML += "a \sqrt[8]{gt} = b " + (a >= b) + " <br/> ";
   document.body.innerHTML += "a < c " + (a < c) + " <br/> ";
   document.body.innerHTML += "a < = c " + (a <= c);</pre>
</script>
```

## 邏輯運算子

a && b	且	如果a和b的值均為true則結果為true
a II b	或	如果a和b的值其中一個為true則結果為true
!a	相反	如果a為false則結果為true

我們常常拿 && 或 || 左右各放一個條件 判斷事件是否成立

# 範例&&

假如你想申請獎學金,必須...

平均總成績達80以上且操行成績不能低於60

# ANS

可以表示成(平均總成績達80以上) && (操行成績不能低於60)

```
if ((平均總成績達80以上) && (操行成績不能低於60)){
拿到獎學金;
```

## 邏輯運算子

a&&b

a且b

```
拿到獎學金
```

```
<!DOCTYPE HTML>
<html>
<head></head>
<body>
<script>
    let Average = 84;
    let Performance = 92;
    if(Average >= 80 && Performance >= 60){
        document.body.innerHTML = "拿到獎學金";
</script>
</body>
</html>
```

# 範例||

FB上有一個 iPhone 抽獎活動

想參加則必須

留言標註三位朋友或 分享貼文,即可參加抽獎!

# ANS

```
可以表示成(留言標註三位朋友) || (分享貼文) || ( (留言標註三位朋友) || (分享貼文) ){
    可以參加抽獎;
```

## 邏輯運算子

a||b

a或b

```
可以參加抽獎
```

```
<!DOCTYPE HTML>
<html>
<head></head>
<body>
<script>
    let Share = false;
    let Tag_three_friends = true;
    if( Share | Tag_three_friends){
        document.body.innerHTML = "可以參加抽獎";
</script>
</body>
</html>
```



老師說

只要不出席考試,就拿不到學分

# ANS

```
可以表示成!(出席)
if (!(出席)){
    拿不到學分;
}
```

!a

a的相反

```
顯示結果
```

```
<!DOCTYPE HTML>
<html>
<head></head>
<body>
<script>
    let Attend = false;
    if( !Attend ){
        document.body.innerHTML = "拿不到學分";
</script>
</body>
</html>
```

### 隨堂小練習

- 小幸在這一生中,只買了一張保單,內容只有100萬的壽險
- 保險公司只會在以下情况理賠100萬
- 1.小幸不幸身亡
- 2.小幸完全失能 (例如,失去一手一眼、全盲...等)
- 3.保單到期
- 小幸想寫一個簡單的判斷是來看他何時可以拿到100萬

### 已知資訊



# | 100萬元GET!

```
<!DOCTYPE HTML>
<html>
<head></head>
<body>
   <script>
       let deadline = 20351231; //2035年12月31日
                                   //2026年01月01日
       let today = 20360101;
       let disability = false;
       let is dead = false;
       if (today > deadline | disability | is_dead) {
            document.body.innerHTML = "100萬元GET!";
   </script>
</body>
</html>
```

