

jQuery程式碼逐行解釋

一、基本概念說明

1.1 什麼是jQuery？

- jQuery是一個JavaScript函式庫，讓操作網頁元素變得更簡單
- `$` 符號是jQuery的核心，用來選取和操作HTML元素

1.2 基本語法結構

```
javascript
```

```
$(選擇器).動作();
```

二、程式碼逐行詳解

2.1 頁面載入完成後執行

```
javascript
```

```
$(document).ready(function () {  
    // 這裡的程式碼會在網頁載入完成後執行  
});
```

解釋：

- `$(document)` = 選取整個網頁文件
- `.ready()` = 等待網頁載入完成
- `function() {}` = 要執行的程式碼放在這裡

2.2 綁定HTML元素到變數

```
javascript
```

```
const money = $('#money');
const rate = $('#rate');
const year = $('#year');
const myBtn = $('#myBtn');
const myTable = $('#myTable');
```

逐行解釋：

- `const money = $('#money');`
 - `const` = 宣告一個常數變數
 - `money` = 變數名稱
 - `$('#money')` = 選取id="money"的HTML元素
 - 簡單說：把網頁上id叫"money"的輸入框，存到變數money裡

對應的HTML可能長這樣：

html

```
<input type="text" id="money" placeholder="請輸入金額">
<input type="text" id="rate" placeholder="請輸入利率">
<input type="text" id="year" placeholder="請輸入年限">
<button id="myBtn">送出</button>
<table id="myTable"> </table>
```

2.3 按鈕點擊事件

javascript

```
myBtn.click(function () {
    // 當按鈕被點擊時，執行這裡的程式碼
});
```

詳細解釋：

- `myBtn` = 前面綁定的按鈕元素

- `.click()` = 監聽點擊事件
- `function() {}` = 點擊時要執行的程式碼

生活化比喻：就像在按鈕上裝了一個"監聽器"，當有人按下按鈕時，就會執行裡面的指令。

2.4 取得輸入框的值

javascript

```
let moneyValue = Number(money.val());  
let rateValue = Number(rate.val());  
let yearValue = Number(year.val());
```

逐行分解：

第一行：`let moneyValue = Number(money.val());`

- `let` = 宣告一個變數
- `moneyValue` = 變數名稱
- `money.val()` = 取得money輸入框裡的文字
- `Number()` = 將文字轉換成數字
- `=` = 把結果存到moneyValue變數裡

舉例說明：

javascript

```
// 假設使用者在money輸入框裡輸入了 "100"  
money.val()    // 結果是 "100" (文字)  
Number(money.val()) // 結果是 100 (數字)
```

2.5 console.log() 除錯

javascript

```
console.log('moneyValue', moneyValue);
console.log('rateValue', rateValue);
console.log('yearValue', yearValue);
```

用途：

- 在瀏覽器的開發者工具顯示變數內容
- 用來檢查程式是否正確運行
- 類似"印出來看看"的概念

2.6 字串模板

javascript

```
let tmpText = `  
  <tr>  
    <td>1</td>  
    <td>10</td>  
  </tr>  
`;
```

解釋：

- 反引號 ``` 可以建立多行字串
- 這裡建立了一個HTML表格的行
- `<tr>` = 表格行
- `<td>` = 表格欄位

2.7 加入內容到表格

javascript

```
myTable.append(tmpText);
```

解釋：

- `myTable` = 前面綁定的表格元素
- `.append()` = 在表格最後面加入新內容
- `tmpText` = 要加入的HTML字串

三、整體運作流程

1. 網頁載入完成
- ↓
2. 綁定HTML元素到變數
- ↓
3. 設定按鈕點擊監聽
- ↓
4. 使用者點擊按鈕
- ↓
5. 取得輸入框的值
- ↓
6. 將文字轉換成數字
- ↓
7. 建立HTML表格內容
- ↓
8. 加入到網頁表格中

四、常見問題解答

Q1: 為什麼要用 `Number()` 轉換？

A: 因為輸入框的值永遠是文字，需要轉成數字才能做數學運算

javascript

`"100" + "50" = "10050"` // 文字相加

`100 + 50 = 150` // 數字相加

Q2: `$('#money')` 和 `document.getElementById('money')` 有什麼差別？

A: 功能相同，但jQuery寫法更簡潔

```
javascript
```

```
// 原生JavaScript
```

```
document.getElementById('money').value
```

```
// jQuery
```

```
$('#money').val()
```

Q3: 為什麼要用 `const` 宣告元素？

A: 因為HTML元素不會改變，用const防止意外修改

```
javascript
```

```
const money = $('#money'); // 元素不會變
```

```
let moneyValue = 100; // 值可能會變
```

五、實際運用練習

試著理解這段完整程式碼：

```
javascript
```

```
// 1. 等網頁載入完成
$(document).ready(function () {
  // 2. 找到HTML元素
  const inputBox = $('#userInput');
  const submitBtn = $('#submitButton');

  // 3. 設定按鈕點擊事件
  submitBtn.click(function() {
    // 4. 取得使用者輸入的文字
    let userText = inputBox.val();

    // 5. 在控制台顯示結果
    console.log('使用者輸入了 : ', userText);

    // 6. 清空輸入框
    inputBox.val('');
  });
});
```

這段程式碼做了什麼？

1. 等網頁準備好
2. 找到輸入框和按鈕
3. 當按鈕被點擊時
4. 讀取輸入框的內容
5. 在控制台顯示
6. 清空輸入框