## **申論題**

* 請解釋什麼是 prototype inheritance

在 JavaScript 中，每個物件都有一個 prototype (原型)，物件可以從原型上繼承屬性和方法，達到復用程式碼的效果，這就是所謂的 prototypal inheritance (原型繼承)。除此之外，原型也能繼承其他物件，因此物件可以繼承一層又一層的屬性和方法，這形成了所謂的 prototype chain (原型鏈)。

* 請解釋什麼是 closure 及如何使用

會出現閉包（closure）主要是因為在 JavaScript 中，global variable 的錯用可能會使得我們的程式碼出現不可預期的錯誤。

所以我們必須透過閉包讓 function 能夠有 private 變數，利用閉包的作法，讓函式有自己私有變數，換句話說就是一個 function 裡面包了另一個 function，同時會 return 裡面的 function 讓我們可以在外面使用到它。變數間也都是獨立的執行環境不會干擾

* 請解釋什麼是 ajax

即非同步的 JavaScript 和 XML的縮寫，是一種用於創建快速動態網頁的技術，傳統的網頁（不使用 AJAX）如果需要更新內容，必需重載整個網頁頁面。

使用AJAX则網頁不用重新整理，就能即時地透過瀏覽器去跟伺服器溝通，讀取資料。

伺服器對 AJAX 資料請求回應通常是以三種資料格式其中之一（HTML、XML、JSON），最常與 Javascript 做搭配就是 JSON。

* 請解釋 event bubbling

當事件發生在DOM元素上時，該事件並不完全發生在那個元素上。在冒泡階段中，事件冒泡或向上傳播至父級，直到到達window為止；而在捕獲階段中，事件從window開始向下觸發元素 事件或event.target。

事件傳播有三個階段：

捕獲階段–事件從 window 開始，然後向下到每個元素，直到到達目標元素。

目標階段–事件已達到目標元素。

冒泡階段–事件從目標元素冒泡，然後上升到每個元素，直到到達 window。

* == 和 === 有何不同

1、對於string,number等基本型別，== 和 === 有區別

不同類型間比較，== 會進行型別轉換，來做比較彼此的值是否相等，=== 則不會進行型別轉換，所以如果型別不一樣的話就不會相等。

2、對於 Array,Object 等物件型別，==和===沒有區別，他們則是進行址的比較

* 請解釋 same-origin policy

主要是因為考量到使用者的資訊安全，防範駭客的攻擊，針對不同源的請求而定的規範，透過 伺服器 server端在 HTTP Header 的設定，讓瀏覽器能取得不同來源的資源。

同源政策，也就是說 JavaScript 存取資源時，如果是同源的情況下，存取不會受到限制，否則請求的回應就會被瀏覽器攔截掉，不交給 JavaScript 處理。

同源政策指的是雙方的來源要是

相同的通訊協定 (protocol)，即 http/https

相同的網域 (domain)

相同的通訊埠 (port)

* 什麼是 single page app，並怎麼讓它對 SEO 更友善。

單頁面網頁應用，是使用JS渲染網頁的一種技術，好處是所有的網路請求都是通過AJAX的方式進行數據的傳輸，路由跳轉也由前端的路由器進行處理，不用向後端請求新的網頁，但是有個缺點因爲所有的HTML是由JS動態生成，因此在第一次渲染畫面時，會是全空白的，因此對 SEO 不友好。

解決方法：透過SSR方式，在 server 端進行HTML的渲染，再發送至瀏覽器上，之後再由前端JS進行接管，如此一來就可以達到 SEO 的效果。

## **寫程式**

* 建個數到 100 的迴圈，當數字是 3 的倍數時輸出 “fizz”，當數字是 5 的倍數時輸出 "buzz"，當數字同時是 3 與 5 的倍數時輸出 “fizzbuzz”。

const fizzBuzz = () => {

  for (let i = 1; i <= 100; i++) {

    if (i % 15 === 0) {

      console.log('fizzbuzz')

    } else if (i % 3 === 0) {

      console.log('fizz')

    } else if (i % 5 === 0) {

      console.log('buzz')

    }

  }

}

* 用程式畫一個菱形如下

\*

\*\*\*

\*\*\*\*\*

\*\*\*

\*

function printDiamond(n) {

  let space = n - 1;

  for (let i = 0; i < n; i++) {

    for (let j = 0; j < space; j++) document.write("  ");

    for (let j = 0; j <= i; j++) document.write("\*" + "  ");

    document.write("<br>");

    space--;

  }

  space = 0;

  for (let i = n; i > 0; i--) {

    for (let j = 0; j < space; j++) document.write("  ");

    for (let j = 0; j < i; j++) document.write("\*" + "  ");

    document.write("<br>");

    space++;

  }

}

printDiamond(3);

* 實作符合下面的函式：  
  add(2, 5); // 7  
  add(2)(5); // 7

const add = function (a, b) {

  if (arguments.length === 1) {

    return function (c) {

      return a + c;

    };

  } else {

    return a + b;

  }

};

**回答問題**

* var foo = 10 + '20';  
  foo 的值是什麼？  
  ANS:

“1020”

* "i'm a lasagna hog".split("").reverse().join("");  
  上面的陳述式會回傳什麼？

ANS

“goh angasal a m'i”

* ( window.foo || ( window.foo = "bar" ) );  
  window.foo 的值是什麼？  
  ANS:

“bar”

* 下面的兩個 alerts 的結果會是什麼？  
  var foo = "Hello";  
  (function() {  
   var bar = " World";  
   alert(foo + bar);  
  })();  
  alert(foo + bar);  
  ANS:

"Hello Word"

bar is not defined

* var foo = [];  
  foo.push(1);  
  foo.push(2);  
  下面 foo.length 的值是什麼？  
  ANS:

2

* 下面這段會印出什麼？  
  console.log('one');  
  setTimeout(function() {  
   console.log('two');  
  }, 0);  
  console.log('three');  
  ANS:

'one'

'three'

'two'