在es5的時候我們在宣告變數都是用var，

今天我在這邊var一個 a = 1;這是一個很正常的變數宣告

但是我今天var了這個a之後這個a就會跑到我們的window環境底下，

是因為JS的特性函數提升，也可以說Hoisting，

也就是說呢，在瀏覽器運行得時候只要我有var的變數他就會先被Hoisting到瀏覽器的環境底下，

就是window的裡面，我們繼續看，如果我在這邊console.log(a)，

有沒有看到，他是undefined，

為什麼是undefined呢，是因為Hoisting的是變數，而等號後面的1是等瀏覽器渲染完成才賦予他的值，

因此console.log(a)是找的到變數的，

但是沒有值喔，因為值是在執行等於1才會取得的。

那問題就來啦，這邊有一個if判斷式我今天給他一個true，

而裡面的變數只想要在這個括號的範圍裏面被使用到的話，

我不想要在括號裏面的變數會影響到我的全局環境，

可是你會發現他還是undefined，為什麼，

因為VAR沒有所謂的區域概念，他會直接把這個變數提升到全局的window底下，

所以為了解決這個問題ES6就會用let，

let b等於2好了，我們來看一下，

有沒有發現他有報錯喔，為什麼呢

原因是因為let的作用域範圍是以這個括號決定的喔，

也就是說我今天寫了let他就會去看這個括號，

如果超過括號的範圍，他就不會給予執行，也不會影響到全域的變數喔。

好接下來我們來看const

有的時候我們在寫程式時，會有一些不想被更改的值，我們叫做常數，

就是ES6的const，

這邊可以看一下這個範例，我把name改成james然後再看log他會跟你說這個變數不可以被修改

所以你如果有不想被修改的東西就用const，如果要在區域使用的，就用let。

但是還有一個部分要講

const如果今天對象是一個物件或是陣列，會發生甚麼事情呢?

今天我const一個空的陣列，然後呢我在陣列的第0個位置給他一個1，

我們來看一下這個陣列會變怎樣喔，有沒有看到這個陣列會被塞1進去，

你們會不會想說剛剛不是說CONST不能被改變嗎?為什麼你可以塞一個1到空陣列裡面，

仔細聽喔，這個const在對物件或是陣列的時候有一個特例就是

如果是對物件或是陣列賦予內容進去的話是可以的，

但是不能修改他的類型。

來仔細看，如果今天他是一個陣列，我直接把陣列等於1，

有沒有看到他就錯了，也就是說我今天對陣列和物件更改他的類型他就會有問題喔。

你看如果我今天用OBJ，還是會報錯，那你把這把這改成.idx等於1的話，

我在我的物件裡面就塞了一個索引，這個idx等於1的部分啦。

所以你會知道const在對陣列和物件時他只是不能改它的類型，

但不代表我不能寫值進去，這就是變數之間的差別喔。