

W05 Reading (L10)

Principle of JavaScript

- Store stuff in memory
- Go through the code line by line

e.g.

```
const num = 3;  
function multiply By 2 (inputNumber){  
}  
Const output = multiply By2(4);  
Const newOutput = multiplyBy2(10)
```

1. Declaring constant number variable called num
2. Function declaration in memory called multiply
3. Declare variable name called 'will'

Thread in JavaScript

- Single threaded (one thing at a time)
- Synchronous execution (for now)

Higher Order Functions: 高階関数 (関数を返す、関数を引数に取る)

関数を呼び出す時に引数として関数を指定したり、関数の中から戻り値として関数を返すことができます。このように関数を受け取ったり返したりする関数のことを高階関数と呼びます。また関数に引数として渡されて、関数内から呼び出される関数のことをコールバック関数と呼びます。

Higher-Order functions: a function that accepts functions as parameters and/or

returns a function.

Call back: a function passed as an argument to another function

https://www.w3schools.com/js/js_callback.asp

e.g.

```
function myFirst() {  
    myDisplayer("Hello");  
}
```

```
function mySecond() {  
    myDisplayer("Goodbye");  
}
```

```
myFirst();  
mySecond();
```

This example will show “Goodbye”

Differences between higher-order functions and call back:

A higher-order function is a function that takes another function(s) as an argument(s) and/or returns a function to its callers.

A callback function is a function that is passed to another function with the expectation that the other function will call it.