

University of Science, HCMC



Lab:

JavaScript Cơ bản

Phát triển ứng dụng Web –JavaScript

Dec 14th, 2020



MSc. Trần Thị Bích Hạnh – BSc. Nguyễn Đức Huy

1 Mục tiêu

Hướng dẫn sinh viên làm quen với ngôn ngữ JavaScript.

Nguồn tham khảo:

- <http://htmlcheatsheet.com/js/>
- <http://jsdares.com>

2 Quy định bài thực hành và định dạng nộp bài

- Thời gian thực hiện: 2 tiết
- Sinh viên sẽ tạo một thư mục webroot có định dạng đặt tên: **<MSSV>_JS**
- Mỗi bài tập sẽ được lưu theo định dạng sau: **<MSSV>_<Tên của file>.js**
- Ví dụ: MSSV = **1234567**, file **Ex01** định dạng nộp bài làm là **1234567_Ex01.js**
- Công cụ sử dụng để lập trình là **Visual Studio Code**
- Link download:
 - o Visual Studio Code: <https://code.visualstudio.com/download>
- Yêu cầu:
 - o Các trường hợp làm bài không đúng quy định sẽ không được chấp nhận.

3 Hướng dẫn một số cú pháp của JavaScript

3.1 Tạo và thực thi file JavaScript trên môi trường Node.js

- o **Tạo file JavaScript làm việc trong Visual Studio Code**

Mở Visual Studio Code > Menu File > Open ... > Chọn thư mục **<MSSV>_JS** > Open
Menu File > New Document > Đặt tên file **<MSSV>_<Tên của file>.js**

- o **Thực thi mã lệnh JavaScript trên môi trường Node.js**

Menu View > Terminal (hoặc nhấn tổ hợp phím Ctrl + `)

Gõ lệnh: **\$ node <MSSV>_<Tên của file> + Enter**

3.2 Cú pháp cơ bản

○ Khai báo biến và kiểu dữ liệu

```
var age = 18;           // number
var name = "Jane";      // string
var name = {first:"Jane", last:"Doe"}; // object
var truth = false;      // boolean
var sheets = ["HTML", "CSS", "JS"]; // array
var a; typeof a;         // undefined
var a = null;           // value null
```

○ Cấu trúc điều khiển:

▪ If - Else

```
if ((age >= 14) && (age < 19)) {
    status = "Eligible.";
} else {
    status = "Not eligible.";
}
```

▪ Switch

```
switch (new Date().getDay()) {
    case 6:
        text = "Saturday";
        break;
    case 0:
        text = "Sunday";
        break;
    default:
        text = "Whatever";
}
```

○ Vòng lặp:

▪ For

```
var sum = 0;
for (var i = 0; i < a.length; i++) {
    sum += a[i];
} // parsing an array
```

- While

```
var i = 1;
while (i < 100) {
    i *= 2;
    console.log(i + ", ");
}
```

- Do While

```
var i = 1;
do {
    i *= 2;
    console.log(i + ", ");
} while (i < 100);
```

- Khai báo hàm:

```
function addNumbers(a, b) {
    return a + b; ;
}
x = addNumbers(1, 2);
```

- Đối tượng:

```
var student = {                // object name
    firstName: "Jane",         // list of properties
    lastName: "Doe",
    age: 18,
    height: 170,
    fullName: function() {    // object function
        return this.firstName + " " + this.lastName;
    }
};
student.age = 19;              // setting value
student[age]++;               // incrementing
name = student.fullName();     // call object function
```

- Sắp xếp mảng:

```
var points = [40, 100, 1, 5, 25, 10];
points.sort(function(a, b){return a - b});
```

4 Bài tập

4.1 Exo1-Syntax

Gõ lại tất cả mã nguồn có trong mục 3 Cú pháp. Chạy chương trình và kiểm tra kết quả.

4.2 Exo2-FindTheLargest

Gõ lại mã nguồn chương trình như hình bên dưới, hãy bổ sung nội dung hàm getLargest sao cho kết quả trả về giá trị lớn nhất trong 3 tham số truyền vào.

Ex02-FindTheLargest.js

Kiểm tra kết quả khi chạy chương trình:

```
function getLargest(a, b, c) {  
    // instead of always returning a, change it to  
    // return the biggest of a, b, and c!  
    return a;  
}  
  
console.log(getLargest(2, 3, 4));  
console.log(getLargest(4, 3, 2));  
console.log(getLargest(3, 4, 2));  
console.log(getLargest(3, 2, 4));  
console.log(getLargest(17, 1, 8));  
console.log(getLargest(5, 5, 9));  
console.log(getLargest(1, 22, 3));  
console.log(getLargest(88, 87, 86));  
console.log(getLargest(4, 2, 3));
```

```
4  
4  
4  
4  
17  
9  
22  
88  
4
```

4.3 Exo3-Fibonacci

Viết chương trình in ra 30 số đầu tiên của dãy Fibonacci.

```
Fibonacci's number 0 is: 0
Fibonacci's number 1 is: 1
Fibonacci's number 2 is: 1
Fibonacci's number 3 is: 2
Fibonacci's number 4 is: 3
Fibonacci's number 5 is: 5
Fibonacci's number 6 is: 8
Fibonacci's number 7 is: 13
Fibonacci's number 8 is: 21
Fibonacci's number 9 is: 34
Fibonacci's number 10 is: 55
Fibonacci's number 11 is: 89
Fibonacci's number 12 is: 144
Fibonacci's number 13 is: 233
Fibonacci's number 14 is: 377
Fibonacci's number 15 is: 610
Fibonacci's number 16 is: 987
Fibonacci's number 17 is: 1597
Fibonacci's number 18 is: 2584
Fibonacci's number 19 is: 4181
Fibonacci's number 20 is: 6765
Fibonacci's number 21 is: 10946
Fibonacci's number 22 is: 17711
Fibonacci's number 23 is: 28657
Fibonacci's number 24 is: 46368
Fibonacci's number 25 is: 75025
Fibonacci's number 26 is: 121393
Fibonacci's number 27 is: 196418
Fibonacci's number 28 is: 317811
Fibonacci's number 29 is: 514229
```

4.4 Exo4-Average

Định nghĩa hàm **average** tính giá trị trung bình của một mảng số, biết rằng nếu gọi hàm với các tham số truyền vào như hình bên dưới:

```
// define your function here

// call your function
average([0]);
average([1,2,3]);
average([1,2,3,4]);
average([1,4,4,4,1]);
average([-12,-13,512,1337]);
```

Sẽ nhận được kết quả như sau:

```
The average of
0
is: 0

The average of
1
2
3
is: 2

The average of
1
2
3
4
is: 2.5

The average of
1
4
4
4
1
is: 2.8

The average of
-12
-13
512
1337
is: 456
```

4.5 Ex05-SortObjects

Định nghĩa một mảng chứa các đối tượng student như trong mục 3 Cú pháp, mỗi student có giá trị cụ thể tự cho trước. Thực hiện sắp xếp mảng tăng dần theo tuổi của student và xuất kết quả ra màn hình console.