⁃ Tìm hiểu về javascript: các kiểu dữ liệu, biến, hàm, hằng số, các loại vòng lặp, mảng, objects... (1 tuần)

+var: Phạm vi hoạt động trong toàn bộ function hoặc global (nếu khai báo ngoài function).

+let: Có phạm vi hoạt động trong block code (giới hạn trong cặp dấu {}).

+const: Tương tự như let, nhưng giá trị của nó không thể thay đổi sau khi được gán.

JavaScript có nhiều kiểu dữ liệu, bao gồm:

**+Primitive Types (Kiểu nguyên thủy):**

Number: Số nguyên hoặc số thực.

String: Chuỗi ký tự, như văn bản.

Boolean: Chỉ nhận hai giá trị true hoặc false.

Undefined: Một biến được khai báo nhưng chưa được gán giá trị.

Null: Một giá trị đặc biệt cho thấy rằng một biến không có giá trị.

Symbol: Được giới thiệu trong ES6, nó là một kiểu dữ liệu nguyên thủy độc nhất.

**+Non-Primitive Types:**

Object: Lưu trữ nhiều thuộc tính (property), mỗi thuộc tính có một cặp key-value.

Array: Một dạng object đặc biệt dùng để lưu trữ danh sách các phần tử theo thứ tự.

Hằng số được khai báo bằng từ khóa const. Giá trị của hằng số không thể thay đổi sau khi được gán.

**Hàm (Functions)**

* Hàm là các đoạn mã có thể được định nghĩa và gọi nhiều lần. Có nhiều cách khai báo hàm:

**Hàm thông thường**:

function greet(name) {

return `Hello, ${name}!`;

}

**Hàm ẩn danh (Anonymous Function)**:

const greet = function(name) {

return `Hello, ${name}!`;

};

**Arrow Function**: Cú pháp ngắn gọn hơn cho hàm ẩn danh.

const greet = (name) => `Hello, ${name}!`;

**Các loại vòng lặp (Loops)**

**for**: Vòng lặp phổ biến nhất, lặp với số lần xác định.for (let i = 0; i < 5; i++) {

console.log(i);

}

**while**: Lặp khi điều kiện còn đúng.

let i = 0;

while (i < 5) {

console.log(i);

i++;

}

**do...while**: Tương tự while, nhưng đảm bảo chạy ít nhất một lần.

let i = 0;

do {

console.log(i);

i++;

} while (i < 5);

**for...of**: Dùng để lặp qua các phần tử của Array, String, hoặc các đối tượng iterable khác.

const array = [1, 2, 3];

for (let value of array) {

console.log(value);

}

**for...in**: Dùng để lặp qua các thuộc tính của một đối tượng.

const obj = { a: 1, b: 2, c: 3 };

for (let key in obj) {

console.log(key, obj[key]);

}

**Mảng (Arrays)**

* Mảng là một danh sách các phần tử, có thể chứa các giá trị khác nhau.

const fruits = ["apple", "banana", "cherry"];

* Một số phương thức cơ bản với mảng:
  + push(): Thêm phần tử vào cuối mảng.
  + pop(): Xóa phần tử cuối cùng của mảng.
  + shift(): Xóa phần tử đầu tiên của mảng.
  + unshift(): Thêm phần tử vào đầu mảng.
  + map(), filter(), reduce(): Duyệt qua mảng và thực hiện các thao tác khác nhau.

**Đối tượng (Objects)**

* Đối tượng là tập hợp các cặp key: value, dùng để mô tả các thuộc tính của một thực thể.

const person = {

name: "John",

age: 30,

greet: function() {

console.log("Hello, my name is " + this.name);

}

};

person.greet(); // Kết quả: "Hello, my name is John"

* Có thể truy cập vào thuộc tính của đối tượng bằng dấu chấm . hoặc dấu ngoặc vuông []:

console.log(person.name); // John

console.log(person["age"]); // 30