JavaScript Variables, Data Types, and Objects

1. **JavaScript Variables**

A **variable** in JavaScript is like a container where you can store different types of values, such as numbers, text, or even more complex things like objects. You can think of it as a label for some information.

To declare a variable, you can use `let`, `const`, or `var`.

- **`let`**: Allows you to change the value later.
- **`const`**: The value cannot be changed after it's set.
- **`var`**: An older way to declare variables (mostly replaced by `let` and `const`).

```
let name = "John";  // You can change this later
const age = 30;     // You can't change this value
var city = "New York"; // Older way, but still works
```

- **`name`** is a variable storing a string `"John"`.
- **`age`** is a variable storing a number `30`.
- **`city`** is a variable storing a string `"New York"` (using the old way).

2. **JavaScript Data Types**

JavaScript has different **data types** that define the kind of data stored in a variable. Here are the basic ones:

1. **String**: Used for text, surrounded by quotes (`" "` or `' '`).

    ```
    let myString = "Hello, world!";
    ```

2. **Number**: Used for numbers (integers, floats).

```
let myNumber = 25;
```

3. **Boolean**: True or False values.

```
let isLoggedIn = true;
```

4. **Undefined**: A variable that has been declared but not assigned a value.

```
let myVar;
console.log(myVar); // Output: undefined
```

5. **Null**: Represents "nothing" or "empty value."

```
let emptyValue = null;
```

6. **Object**: A more complex data type that can store collections of data (key-value pairs).

```
let person = {
    name: "Alice",
    age: 28
};
```

7. **Array**: A type of object that stores multiple values in a list.

```
let myArray = [1, 2, 3, "apple", "banana"];
```

3. **JavaScript Objects**

An **object** in JavaScript is a collection of related data or functionality, stored as **key-value pairs**. You can think of an object as a real-world item like a car, where the properties like color, model, and speed describe it.

Example of a simple object:

```
let car = {
    brand: "Tesla",
    model: "Model 3",
    year: 2022,
    isElectric: true
};
```

In this object:

- **brand**: `"Tesla"` (string)
- **model**: `"Model 3"` (string)
- **year**: `2022` (number)
- **isElectric**: `true` (boolean)

You can access values inside the object like this:

```
console.log(car.brand);  // Output: "Tesla"
console.log(car.year);   // Output: 2022
```

Example of adding a function inside an object:

Objects can also have **methods** (functions inside an object).

```
let person = {
    name: "John",
```

```
    greet: function() {
        console.log("Hello, my name is " + this.name);
    }
};

person.greet();  // Output: Hello, my name is John
```

Recap

- **Variables**: Store data (using `let`, `const`, or `var`).
- **Data types**: Different kinds of data like strings, numbers, booleans, objects, etc.
- **Objects**: Collections of data stored as key-value pairs (e.g., a car or person object).

Example Combining Everything:

```
let productName = "Laptop";  // string
let price = 999.99;          // number
let isAvailable = true;      // boolean

let product = {
    name: "Laptop",
    price: 999.99,
    inStock: true,
    showDetails: function() {
        console.log(this.name + " costs $" + this.price);
    }
};

product.showDetails();  // Output: Laptop costs $999.99
```

This is the foundation of JavaScript: using variables to store information, defining the type of data, and using objects to organize related data and behaviors!