

Revision Week 5 -

Week 5: Variables & Data Types

re-assigned and re-declared within their scope.

```
var name = "Maroof Shittu"; // Declaration and initialization
console.log(name); // Output: Maroof Shittu
name = "John Doe"; // Re-assignment
console.log(name); // Output: John Doe
```

- **let**: Introduced in ES6 (ECMAScript 2015), **let** allows you to declare block-scoped variables, which means they are only accessible within the block they are declared in. They can be re-assigned but not re-declared within the same scope.

```
let name = "Ekeson"; // Declaration and initialization
console.log(name); // Output: Ekeson
name = "John Doe"; // Re-assignment
console.log(name); // Output: John Doe
let name = "Jane Doe"; // Error: Identifier 'name' has already been declared
```

```
let student = Helen ; // Error: Invalid or unexpected token
```

- **Cannot contain hyphens or dashes:** Hyphens are not allowed in variable names.

```
let student-name = "Helen"; // Error: Unexpected token '-'
```

- **Special characters:** Only underscores and dollar signs are allowed.

```
let _student = "Helen"; // Valid  
let $student = "Helen"; // Valid
```

- **Camel Case:** Common convention for naming variables where the first word is lowercase and subsequent words are capitalized.

- **boolean**: Represents true or false values.

```
let isStudent = true;
```

- **null**: Represents an intentional absence of any value.

```
let emptyValue = null;
```

- **undefined**: Represents an uninitialized variable.

```
let notAssigned;
```

- **symbol**: Represents a unique identifier.

```
let sym = Symbol("unique");
```

- **Reference**: Complex data types that can hold multiple values.

Week 6: Operators

of a web page in a tree-like format, where each node is an object representing a part of the document, such as an element, attribute, or text.

Here's a simple analogy: Imagine a web page is like a family tree. The DOM is the family tree diagram that shows how all the family members (elements) are related to each other.

Key Concepts of the DOM

1. **Document Object:** This is the root of the DOM tree. It represents the entire HTML document.
2. **Elements:** These are the nodes in the DOM tree. Each