Variables and Data Types

var, let, const

var: Function-scoped. Can be re-declared and updated.

```
var x = 10;
var x = 20; // valid
x = 30;     // valid
```

let: Block-scoped. Cannot be re-declared in the same scope but can be updated.

```
let y = 10;
// let y = 20; // Error
y = 30;       // valid
```

const: Block-scoped. Cannot be re-declared or updated. Must be initialized at the time of declaration.

```
const z = 10;
// const z = 20; // Error
// z = 30;       // Error
```

Primitive Types

String: Textual data.

```
let name = "John";
```

Number: Includes integers and floating-point numbers.

```
let age = 25;
let score = 92.5;
```

Boolean: Represents true or false.

```
let isValid = true;
```

Null: Represents an intentional absence of value.

```
let value = null;
```

Undefined: A variable declared but not assigned a value.

```
let data;
console.log(data); // undefined
```

BigInt: Used for very large integers.

```
let bigNumber = 1234567890123456789012345678901234567890n;
```

Automatic or implicit conversion of values from one type to another.

```
console.log('5' + 1);    // '51' - Number is coerced to string
console.log('5' - 1);    // 4 - String is coerced to number
console.log(true + 1);   // 2 - true is coerced to 1
```

Type Conversion: Manual or explicit conversion using functions like Number(), String(), Boolean().

To String:

```
String(123);     // "123"
(123).toString() // "123"
```

To Number:

```
Number("123");   // 123
parseInt("123"); // 123
parseFloat("123.45"); // 123.45
```

To Boolean:

```
Boolean(0);      // false
Boolean(1);      // true
Boolean("");     // false
Boolean("hi");   // true
```