JavaScript Essentials

Numbers in JavaScript

JavaScript provides several ways to work with numbers. Numbers can be integers or floating-point values. You can perform various operations such as addition, subtraction, multiplication, division, and more. JavaScript also supports different number types like NaN (Not a Number) and Infinity.

Example:
```javascript
let x = 10;
let y = 3.14;
let z = x + y; // 13.14
```

Initializing and Manipulating Strings in JavaScript

Strings are sequences of characters used to store and manipulate text. You can initialize strings using single quotes, double quotes, or backticks for template literals.

Example:
```javascript
let singleQuoteString = 'Hello';
let doubleQuoteString = "World";
let templateLiteral = `Hello, ${doubleQuoteString}!`; // "Hello, World!"
```

Analyzing and Modifying Strings in JavaScript

JavaScript provides many methods to analyze and modify strings. Common methods include `.length`, `.charAt()`, `.substring()`, `.indexOf()`, `.replace()`, and `.toUpperCase()`.

Example:
```javascript
let str = "JavaScript";
console.log(str.length); // 10
```

```
console.log(str.charAt(0)); // "J"
console.log(str.substring(0, 4)); // "Java"
console.log(str.indexOf('Script')); // 4
console.log(str.replace('Java', 'ECMA')); // "ECMAScript"
console.log(str.toUpperCase()); // "JAVASCRIPT"
```

Dates in JavaScript

JavaScript provides the `Date` object to work with dates and times. You can create a new date, get the current date, and perform various operations on dates.

Example:
```javascript
let now = new Date();
console.log(now); // current date and time

let specificDate = new Date('2024-07-26');
console.log(specificDate); // specific date

let year = now.getFullYear();
let month = now.getMonth() + 1; // January is 0!
let day = now.getDate();
console.log(`${year}-${month}-${day}`); // formatted date
```

Using the Math Library for Common Math Operations

The `Math` object in JavaScript provides several mathematical functions and constants. Common methods include `Math.round()`, `Math.floor()`, `Math.ceil()`, `Math.random()`, `Math.sqrt()`, and `Math.pow()`.

Example:
```javascript
console.log(Math.round(4.7)); // 5
console.log(Math.floor(4.7)); // 4
console.log(Math.ceil(4.3)); // 5
console.log(Math.random()); // random number between 0 and 1
console.log(Math.sqrt(16)); // 4
```

```
console.log(Math.pow(2, 3)); // 8
```

Arithmetic Operators

Arithmetic operators are used to perform arithmetic on numbers. Common operators include `+` (addition), `-` (subtraction), `*` (multiplication), `/` (division), and `%` (modulus).

Example:
```javascript
let a = 10;
let b = 3;
console.log(a + b); // 13
console.log(a - b); // 7
console.log(a * b); // 30
console.log(a / b); // 3.3333...
console.log(a % b); // 1
```

Logical and Conditional Operators

Logical operators include `&&` (AND), `||` (OR), and `!` (NOT). Conditional (ternary) operator `? :` is used as a shorthand for `if-else`.

Example:
```javascript
let x = true;
let y = false;
console.log(x && y); // false
console.log(x || y); // true
console.log(!x); // false

let result = x ? "Yes" : "No";
console.log(result); // "Yes"
```

Type Casting

Type casting is the process of converting one data type to another. JavaScript provides functions like `Number()`, `String()`, and `Boolean()` for explicit type conversion. Implicit type conversion happens automatically in certain operations.

Example:
```javascript
let str = "123";
let num = Number(str); // 123

let num2 = 456;
let str2 = String(num2); // "456"

let bool = Boolean(num); // true
```

Looping Control Structures

JavaScript supports several looping structures: `for`, `while`, and `do...while`. These loops are used to execute a block of code multiple times.

Example:
```javascript
// for loop
for (let i = 0; i < 5; i++) {
  console.log(i);
}

// while loop
let j = 0;
while (j < 5) {
  console.log(j);
  j++;
}

// do...while loop
let k = 0;
do {
  console.log(k);
  k++;
```

```
} while (k < 5);
```