

STEM Digital Academy

School of Science & Technology

Data Type And Operators

Web Technologies



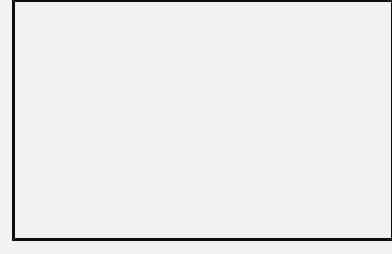
Lecture by Dr Elahe Kani-Zabihi

```
<!doctype html>
<html>
    <head>
     <title>
       Web Technologies
     </title>
      <style>
        p {
         color: ■blue;
     </style>
     <script>
      console.log("client-side");
     </script>
    </head>
    <body>
        >
           Welcome!
       </body>
</html>
```



What we will cover

- Data types
- Operators
- Functions





Data Types

- Number: 15, -21, 3.14
 - let x = 5;
- String: "JavaScript", "hello_world", "myId\$"
 - let y = "javaScript";
- Boolean: true/false
 - let z = true;
- Null
- Undefined
- Symbol
- Bigint





Arithmetic Operators

Addition	+
Subtraction	_
Multiplication	*
Division	/
Modulus	%
Increment	++
Decrement	



Compound Assignment Operators

```
Addition assignment
     let subtotal = 74;
     Subtraction assignment
     let counter = 10;
     counter -= 1; // counter = 9
Multiplication assignment
     let price = 100;
     price *= 0.8;
                  // price = 80
```



The Concatenation Operators for Strings

- Concatenates two values.
- += Concatenates the result of the expression to the end of the variable.



Template Literal - String

Syntax

```
`$ {....}`
```

```
const firstName = " Grace "
const lastName = "Hopper";
```

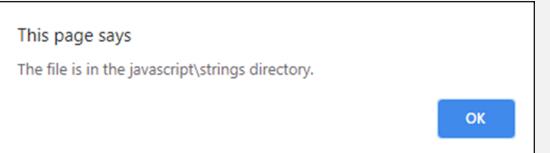
```
const name = `${lastName}, ${firstName}`;
The name "Hopper, Grace" will display on the page
```



Escape Sequence

- \n
- alert("This isn\'t the time to talk about this.");
- **-** \"
- alert("The file is in the javascript\\strings directory.");







Unicode Characters

- \u00A9 ©
- \u00AE®
- \u263A ③
- \u2665 ♥



alert ("I \u2665 Murach\u00AE Publishing! \u263A \n(\u00A9 2020)");



Conditional Expression

Equal to ==

Not equal to

Greater than

Less than

• Greater than or equal to >=

Less than or equal to <=</p>



isNaN()

Syntax isNaN(expression)

Examples

```
isNaN("123") // Returns false
isNaN("abc") // Returns true
```



The Logical Operators

- **&&** AND expression
- | OR expression
- ! NOT expression





Functions

```
function addNumbers (num1, num2) {
    let sum = num1 + num2;
    return sum;
}
let result = addNumbers (5, 10);
document.write ("The result value is " + result);
```



Splitting a Statement

Split a statement after:

- +, -, *, /, =, ==, >, or <
- opening curly bracket
- opening square bracket
- round opening parenthesis
- closing curly bracket

Do not split a statement after:

- an identifier, a value, or the return keyword
- closing square bracket
- round closing paranthesis



Try It Yourself

In this activity, you can practice creating a new function.

- 1. Create a function using the script provided in slide 14.
- 2. You can either create the function inside the <script> element of your HTML file or create an external JavaScript file and link it to your HTML page.

