

JavaScript

5b Variables

Systems and Web Development Workshop
2025 Spring

Dr. Jefferson Fong



Test 1

- Test 1 will be held next week during lab.
- People who **cannot take Test 1** as scheduled for your section must have **valid reasons**.
 - Need to get permission form from CST office (T3-602)
- The test
 - Covers everything up to and including material covered this week
 - Is a written test 1:50 long; bring a pen or two; we will provide scratch papers
 - Is closed book; no books, notes or electronic devices allowed
 - Questions will be multiple choice, short answers and writing code fragments
 - Venue will be announce in iSpace and WeCom



JavaScript Variables, et al (“and others” in Latin)

- Variables and Arithmetic
 - https://www.w3schools.com/js/js_variables.asp
- Assignment
 - https://www.w3schools.com/js/js_assignment.asp
- Operators
 - https://www.w3schools.com/js/js_operators.asp
- For Chinese mirror site, change “.com” to “.cn”, e.g.
 - https://www.w3schools.cn/js/js_variables.asp



JavaScript Variables, et al (“and others” in Latin)

(Mostly) **very similar to C**

- We will go over these quickly.
- If you forgot your C, you can study these slides carefully and run these codes:
 - W3 Schools
 - sample codes from iSpace



JavaScript Variables

Just like C

- JavaScript **variables** are containers for **storing data values**.
 - In this example, x, y, and z, are variables:

```
<h2>JavaScript Variables</h2>
```

```
<p>In this example, x, y, and z are variables.</p>
```

```
<p id="demo"></p>
```

```
<script>
```

```
var x = 5;
```

```
var y = 6;
```

```
var z = x + y;
```

```
document.getElementById("demo").innerHTML =  
"The value of z is: " + z;  
</script>
```

- JS Syntax
- JS Comments
- JS Variables
- JS Let
- JS Const
- JS Operators
- JS Arithmetic
- JS Assignment
- JS Data Types
- JS Functions

What are Variables?

Variables are containers for storing data (storing data values).

In this example, **x**, **y**, and **z**, are variables, declared with the **var** keyword:

Example

Try it Yourself at W3 school (above)
or sample code (below)

5b1_Variables.html



JavaScript Variables

Just like C

- From the example above, you can expect:

- x stores the value 5
- y stores the value 6
- z stores the value 11



```
var x = 5;  
var y = 6;  
var z = x + y;
```

- In this example, price1, price2, and total, are variables:

```
var price1 = 5;  
var price2 = 6;  
var total = price1 + price2;
```



JavaScript Variables

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Variables</h2>

<p id="demo"></p>

<script>
var price1 = 5;
var price2 = 6;
var total = price1 + price2;
document.getElementById("demo").innerHTML =
"The total is: " + total;
</script>

</body>
</html>
```

The code is at left, and
the output is below.

JavaScript Variables

The total is: 11



JavaScript Variables

(Almost) just like C

- JavaScript Identifiers

- All JavaScript variables must be identified with unique names.
- These unique names are called identifiers.

- The general rules

- Names can contain letters, digits, underscores, and **dollar signs**.
- Names must begin with a letter
- Names can also begin with '\$' and '_'
- Names are **case sensitive** (y and Y are different variables)
- Reserved words (like JavaScript keywords) cannot be used as names

unlike C



Assignment Operator

Just like C

- In JavaScript, the equal sign (=) is an "assignment" operator, **not** an "equal to" operator.
 - "replaced by"; x = 3 means "x is replaced by 3"
- The "equal to" operator is written like == in JavaScript.

```
x = x + 5
```

- "New x is replaced by current x + 5".
 - It calculates the value of x + 5 and puts the result into x.
 - The value of x is incremented by 5.
- Run the sample code 5b1_Variables.html (or in W3 School) with the line

```
x = x + 5
```



```
<script>
var x = 5;
x = x + 5;
var y = 6;
var z = x + y;
```



Data Types

- JavaScript variables can hold numbers like 100 and text values like "John Doe".

```
var pi = 3.14;  
var person = "John Doe";  
var answer = 'Yes I am!';
```

Unlike C

- Text values are called text strings.
 - **Strings** are written inside **double** or **single quotes**.
 - **Numbers** are written **without quotes**.
- **Unlike C**, JavaScript uses “**weak data type**”
 - Don't have to explicitly specify the data type of a variable.
 - Interpreter figures out the data type from the assigned value.



Data Types

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Variables</h2>

<p>Strings are written with quotes.</p>
<p>Numbers are written without quotes.</p>

<p id="demo"></p>

<script>
var pi = 3.14;
var person = "John Doe";
var answer = 'Yes I am!';

document.getElementById("demo").innerHTML =
pi + "<br>" + person + "<br>" + answer;
</script>

</body>
</html>
```

JavaScript Variables

Strings are written with quotes.

Numbers are written without quotes.

3.14

John Doe

Yes I am!

https://www.w3schools.com/js/tryit.asp?filename=tryjs_variables_types

5b2_VariablesDataTypes.html



Declaring Variables

Unlike C

- **Declare** (or create) a JavaScript variable with the **var** keyword.
- After the declaration, the variable has no value.
 - Technically it has the value “undefined”.
- **Assign** a value to the variable with the equal sign ‘=’
- You can also assign a value to the variable when you declare it.

```
var carName;
```

```
carName = "Volvo";
```

```
var carName = "Volvo";
```



Arithmetic

Just like C

- Arithmetic are done with operators such as = and +.

```
var x = 5 + 2 + 3;  
var x = 5 + 2 * 3;  
var x = 5 * 2 / 3;  
var x = 5 - 2 * 3;  
var x = (5 / 2) + 3;
```

- Arithmetic operation operates on numbers or variable

- E.g. `var fullName = firstName + " " + lastName;`

```
var x = 100 + 50;
```

```
var x = a + b;
```



Operators

Just like C, arithmetic operators are

`+, -, *, **, /, %, ++, --`

- See
 - https://www.w3schools.com/js/js_operators.asp
 - https://www.w3schools.cn/js/js_operators.asp