Kode MK	:	12211013
Nama MK	:	*Mobile Computing*
SKS	:	2
Jumlah Jam	:	3
Prodi	:	Teknik Informatika Strata Satu
Dosen Pengampu	:	Emanuel Safirman Bata, S.Kom., M.T.

Materi: Dasar-Dasar JS (Lanjutan)

1. JavaScript Operators

   Example. Assign values to variables and add them together:

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

<h2>JavaScript Operators</h2>

<p>x = 5, y = 2, calculate z = x + y, and
display z:</p>

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

<script>
var x = 5;
var y = 2;
var z = x + y;
document.getElementById("demo").innerHTML =
z;
</script>

</body>
</html>
```

JavaScript Operators

x = 5, y = 2, calculate z = x + y, and display z:

7

   The assignment operator (=) assigns a value to a variable.

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

<h2>The = Operator</h2>

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

<script>
var x = 10;
document.getElementById("demo").inne
rHTML = x;
</script>

</body>
</html>
```

The = Operator

10

2. JavaScript Arithmetic Operators

   Arithmetic operators are used to perform arithmetic on numbers:

Operator	Description
+	Addition
-	Subtraction
*	Multiplication
/	Division
%	Modulus (Remainder)
++	Increment
--	Decrement

   Example (1):

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

<h2>The + Operator</h2>

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

<script>
var x = 5;
var y = 2;
var z = x + y;
document.getElementById("demo").innerHTML = z;
</script>

</body>
</html>
```

The + Operator

7

Example (2):

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

<h2>The ++ Operator</h2>

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

<script>
var x = 5;
x++;
var z = x;
document.getElementById("demo").innerHTML = z;
</script>

</body>
</html>
```

The ++ Operator

6

3. JavaScript Assignment Operators

Assignment operators assign values to JavaScript variables.

Operator	Example	Same As
=	x = y	x = y
+=	x += y	x = x + y
-=	x -= y	x = x - y
*=	x *= y	x = x * y
/=	x /= y	x = x / y
%=	x %= y	x = x % y

<<=	x <<= y	x = x << y
>>=	x >>= y	x = x >> y
>>>=	x >>>= y	x = x >>> y
&=	x &= y	x = x & y
^=	x ^= y	x = x ^ y
|=	x |= y	x = x | y
**=	x **= y	x = x ** y

Example (1):

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

<h2>The += Operator</h2>

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

<script>
var x = 10;
x += 5;
document.getElementById("demo").in
nerHTML = x;
</script>

</body>
</html>
```

The += Operator

15

Example (2):

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

<h2>The %= Operator</h2>

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

<script>
var x = 10;
x %= 5;
document.getElementById("demo").innerHTML = x;
</script>

</body>
</html>
```

The %= Operator

0

4. Operator Precedence
   Operator precedence describes the order in which operations are performed in an arithmetic expression. Example

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

<p>Multiplication has precedence over addition.</p>

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

<script>
document.getElementById("demo").innerHTML = 100 + 50 * 3;
</script>

</body>
</html>
```

Multiplication has precedence over addition.

250

Is the result of example above the same as 150 * 3, or is it the same as 100 + 150? Is the addition or the multiplication done first? As in traditional school mathematics, the multiplication is done first. Multiplication (*) and division (/) have higher precedence than addition (+) and subtraction (-). And (as in school mathematics) the precedence can be changed by using parentheses: Example:

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

<p>Multiplication has precedence over addition.</p>
<p>But parenthesis has precedence over multiplication.</p>

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

<script>
document.getElementById("demo").innerHTML = (100 + 50) * 3;
</script>

</body>
</html>
```

Multiplication has precedence over addition.

But parenthesis has precedence over multiplication.

450

5. JavaScript Operator Precedence Values

Value	Operator	Description	Example
19	( )	Expression grouping	(3 + 4)
18	.	Member	person.name
18	[]	Member	person["name"]
17	()	Function call	myFunction()
17	new	Create	new Date()
16	++	Postfix Increment	i++
16	--	Postfix Decrement	i--
15	++	Prefix Increment	++i
15	--	Prefix Decrement	--i
15	!	Logical not	!(x==y)
15	typeof	Type	typeof x
14	*	Multiplication	10 * 5
14	/	Division	10 / 5
14	%	Modulo division	10 % 5
14	**	Exponentiation	10 ** 2
13	+	Addition	10 + 5
13	-	Subtraction	10 - 5
12	<<	Shift left	x << 2
12	>>	Shift right	x >> 2
12	>>>	Shift right (unsigned)	x >>> 2
11	<	Less than	x < y
11	<=	Less than or equal	x <= y
11	>	Greater than	x > y
11	>=	Greater than or equal	x >= y
10	==	Equal	x == y
10	===	Strict equal	x === y
10	!=	Unequal	x != y
10	!==	Strict unequal	x !== y
6	&&	Logical and	x && y
5	||	Logical or	x || y

3	=	Assignment	x = y
3	+=	Assignment	x += y
3	-=	Assignment	x -= y
3	*=	Assignment	x *= y
3	%=	Assignment	x %= y
3	<<=	Assignment	x <<= y
3	>>=	Assignment	x >>= y
3	>>>=	Assignment	x >>>= y
3	&=	Assignment	x &= y
3	^=	Assignment	x ^= y
3	|=	Assignment	x |= y

6. JavaScript String Operators

   The + operator can also be used to add (concatenate) strings. Example

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

<h2>JavaScript Operators</h2>

<p>The + operator concatenates (adds) strings.</p>

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

<script>
var txt1 = "John";
var txt2 = "Doe";
document.getElementById("demo").innerHTML = txt1 + " " + txt2;
</script>

</body>
</html>
```

JavaScript Operators

The + operator concatenates (adds) strings.

John Doe

   The += assignment operator can also be used to add (concatenate) strings: Example

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

<h2>JavaScript Operators</h2>

<p>The assignment operator += can concatenate strings.</p>

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

<script>
txt1 = "What a very ";
txt1 += "nice day";
document.getElementById("demo").innerHTML = txt1;
</script>

</body>
</html>
```

JavaScript Operators

The assignment operator += can concatenate strings.

What a very nice day

7. Adding Strings and Numbers

   Adding two numbers, will return the sum, but adding a number and a string will return a string: Example

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

<h2>JavaScript Operators</h2>

<p>Adding a number and a string, returns a string.</p>

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

<script>
var x = 5 + 5;
var y = "5" + 5;
var z = "Hello" + 5;
document.getElementById("demo").innerHTML =
x + "<br>" + y + "<br>" + z;
</script>

</body>
</html>
```

JavaScript Operators

Adding a number and a string, returns a string.

10
55
Hello5

8. JavaScript Comparison Operators

Operator	Description
==	equal to
===	equal value and equal type
!=	not equal
!==	not equal value or not equal type
>	greater than
<	less than
>=	greater than or equal to
<=	less than or equal to
?	ternary operator

Example:

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

<h2>JavaScript Comparison</h2>

<p>Assign 5 to x, and display the value of the comparison
(x <= 8).</p>

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

<script>
var x = 5;
document.getElementById("demo").innerHTML = (x <= 8);
</script>

</body>
</html>
```

JavaScript Comparison

Assign 5 to x, and display the value of the comparison (x <= 8).

true

9. JavaScript Logical Operators

Operator	Description
&&	logical and
||	logical or
!	logical not

Example:

```
<!DOCTYPE html>
<html>
<body>
<h2>JavaScript Comparison</h2>

<p>The AND operator (&&) returns true if both expressions
are true, otherwise it returns false.</p>

<p id="demo"></p>
<script>
var x = 6;
var y = 3;

document.getElementById("demo").innerHTML =
(x < 10 && y > 1) + "<br>" +
(x < 10 && y < 1);
</script>

</body>
</html>
```

JavaScript Comparison

The AND operator (&&) returns true if both expressions are true, otherwise it returns false.

true
false

10. JavaScript Data Types

JavaScript variables can hold many data types: numbers, strings, objects and more:

```
var length = 16;                               // Number
var lastName = "Johnson";                      // String
var x = {firstName:"John", lastName:"Doe"};    // Object
```

The Concept of Data Types

In programming, data types is an important concept. To be able to operate on variables, it is important to know something about the type. Without data types, a computer cannot safely solve this:

```
var x = 16 + "Volvo";
```

Does it make any sense to add "Volvo" to sixteen? Will it produce an error or will it produce a result? JavaScript will treat the example above as:

```
var x = "16" + "Volvo";
```

Example (1)

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

<h2>JavaScript</h2>

<p>When adding a string and a number, JavaScript will treat
the number as a string.</p>

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

<script>
var x = "Volvo" + 16;
document.getElementById("demo").innerHTML = x;
</script>

</body>
</html>
```

JavaScript

When adding a string and a number, JavaScript will treat the number as a string.

Volvo16

JavaScript evaluates expressions from left to right. Different sequences can produce different results:

Example (2):

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

<h2>JavaScript</h2>

<p>JavaScript evaluates expressions from left to right.
Different sequences can produce different results:</p>

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

<script>
var x = 16 + 4 + "Volvo";
document.getElementById("demo").innerHTML = x;
</script>

</body>
</html>
```

JavaScript

JavaScript evaluates expressions from left to right. Different sequences can produce different results:

20Volvo

Example (3):

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

<h2>JavaScript</h2>

<p>JavaScript evaluates expressions from left to right.
Different sequences can produce different results:</p>

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

<script>
var x = "Volvo" + 16 + 4;
document.getElementById("demo").innerHTML = x;
</script>

</body>
</html>
```

JavaScript

JavaScript evaluates expressions from left to right. Different sequences can produce different results:

Volvo164

11. JavaScript Numbers

JavaScript has only one type of numbers. Numbers can be written with, or without decimals:

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

<h2>JavaScript Numbers</h2>
<p>Numbers can be written with, or without decimals:</p>

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

<script>
var x1 = 34.00;
var x2 = 34;
var x3 = 3.14;

document.getElementById("demo").innerHTML =
x1 + "<br>" + x2 + "<br>" + x3;
</script>

</body>
</html>
```

JavaScript Numbers

Numbers can be written with, or without decimals:

34
34
3.14

Extra large or extra small numbers can be written with scientific (exponential) notation:

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

<h2>JavaScript Numbers</h2>
<p>Extra large or extra small numbers can be written with
scientific (exponential) notation:</p>

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

<script>
var y = 123e5;
var z = 123e-5;

document.getElementById("demo").innerHTML =
y + "<br>" + z;
</script>

</body>
</html>
```

JavaScript Numbers

Extra large or extra small numbers can be written with scientific (exponential) notation:

12300000
0.00123

12.JavaScript Booleans

Booleans can only have two values: true or false.

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

<h2>JavaScript Booleans</h2>

<p>Booleans can have two values: true or false:</p>

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

<script>
var x = 5;
var y = 5;
var z = 6;
document.getElementById("demo").innerHTML =
(x == y) + "<br>" + (x == z);
</script>

</body>
</html>
```

JavaScript Booleans

Booleans can have two values: true or false:

true
false

13.JavaScript Arrays

JavaScript arrays are written with square brackets. Array items are separated by commas. The following code declares (creates) an array called cars, containing three items (car names): Example

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

<h2>JavaScript Arrays</h2>

<p>Array indexes are zero-based, which means the first item
is [0].</p>

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

<script>
var cars = ["Saab","Volvo","BMW"];

document.getElementById("demo").innerHTML = cars[0];
</script>

</body>
</html>
```

JavaScript Arrays

Array indexes are zero-based, which means the first item is [0].

Saab

14.JavaScript Objects

JavaScript objects are written with curly braces. Object properties are written as name:value pairs, separated by commas. Example

```
<!DOCTYPE html>
<html>
<body>
<h2>JavaScript Objects</h2>
<p id="demo"></p>

<script>
var person = {
    firstName : "John",
    lastName  : "Doe",
    age       : 50,
    eyeColor  : "blue"
};

document.getElementById("demo").innerHTML =
person.firstName + " is " + person.age + " years old.";
</script>

</body>
</html>
```

JavaScript Objects

John is 50 years old.

15. The typeof Operator

You can use the JavaScript typeof operator to find the type of a JavaScript variable. The typeof operator returns the type of a variable or an expression: Example

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

<h2>JavaScript typeof</h2>
<p>The typeof operator returns the type of a variable or an
expression.</p>

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

<script>
document.getElementById("demo").innerHTML =
typeof "" + "<br>" +
typeof "John" + "<br>" +
typeof "John Doe";
</script>
</body>
</html>
```

JavaScript typeof

The typeof operator returns the type of a variable or an expression.

string
string
string

```
typeof 0                    // Returns "number"
typeof 314                  // Returns "number"
typeof 3.14                 // Returns "number"
typeof (3)                  // Returns "number"
typeof (3 + 4)              // Returns "number"
```

Excercise

1. Carilah nilai A dari persamaan berikut ini:

$$A = \frac{(1+x)^n = 1 + \frac{nx}{1!} + \frac{n(n-1)x^2}{2!} + \cdots r^2}{\frac{\pi}{2}}$$

   Diketahui, nilai x = 4, n = 2, π = nilai konstanta, r = jari-jari lingkaran, diameter lingkaran = 8
   Output: **Nilai A = ...**
2. Sebuah kelapa jatuh dari ketinggian 8 meter. Hitunglah ketinggian kelapa setelah 0,5 detik diukur dari tanah
   Output: **Jarak yang ditempuh kelapa setelah 0,5 detik adalah 1,25 m, maka ketinggian kelapa dihitung dari tanah adalah 6,75 m**
3. Jika waktu yang dibutuhkan agar sebuah batu menyentuh dasar jurang adalah 10 detik, maka berapakah kedalaman jurang tersebut?
   Output: **Jadi, kedalam jurang tersebut adalah 500 m**
4. Buatlah sebuah puisi untuk Pak Eman dengan ketentuan: puisi terdiri dari 4 bait. Satu bait terdiri dari 4 baris. Ukuran dan ketebalan huruf pada judul puisi harus lebih besar dari huruf pada isi puisi