DepED
DEPARTMENT OF EDUCATION

COMPUTER PROGRAMMING
(ICT)

Grade 11

QUARTER 2

MODULE 10

JavaScript Variables, Operators and Data Types

TECHNICAL VOCATIONAL LIVELIHOOD

DIVISION OF PASIG CITY
1994
PASIG CITY METRO MANILA

SAGISAG NG LUNGSOD NG PASIG, KALAKHANG MAYNILA
1573

Computer Programming (ICT) – Grade 11

Quarter 2 – Module 10: JavaScript Variables, Operators and Data Types.

Published by the Department of Education Division of Pasig City

Development Team of the Module

Writer : Magiel L. Boncayao

Editor : Ma.Lerma Cantanero

Reviewer : Rowena O. Dimagiba

Illustrator:

Layout Artist:

Management Team: Ma. Evalou Concepcion A. Agustin
OIC-Schools Division Superintendent
Dr. Aurelio G. Alfonso
OIC-Assistant Schools Division Superintendent
Dr. Victor Javena
OIC – Chief Curriculum Implementation Division

Education Program Supervisors
Librada L. Agon EdD (EPP/TLE/TVL/TVE)
Liza A. Alvarez (Science/STEM/SSP)
Bernard R. Balitao (AP/HUMSS)
Joselito E. Calios (English/SPFL/GAS)
Norlyn D. Conde EdD (MAPEH/SPA/SPS/HOPE/A&D/Sports)
Wilma Q. Del Rosario (LRMS/ADM)
Ma. Teresita E. Herrera EdD (Filipino/GAS/Piling Larang)
Perlita M. Ignacio PhD (EsP)
Dulce O. Santos PhD (Kindergarten/MTB-MLE)
Teresita P. Tagulao EdD (Mathematics/ABM)

Printed in the Philippines by _____
Department of Education – Division of Pasig City
Office Address: Caruncho Avenue, San Nicolas, Pasig City
Telefax:
E-mail Address:

Introductory Message

For the Facilitator:

Welcome to the Computer Programming for the ICT Module on JavaScript Variables, Operators and Data Types.

This module was collaboratively designed, developed, and reviewed by educators from Schools Division Office of Pasig City headed by its Officer-In-Charge Schools Division Superintendent, Ma. Evalou Concepcion A. Agustin in partnership with the Local Government of Pasig through its mayor, Honorable Victor Ma. Regis N. Sotto. The writers utilized the standards set by the K to 12 Curriculum using the Most Essential Learning Competencies (MELC) while overcoming their personal, social, and economic constraints in schooling.

This learning material hopes to engage the learners into guided and independent learning activities at their own pace and time. Further, this also aims to help learners acquire the needed 21st century skills especially the 5 Cs namely: Communication, Collaboration, Creativity, Critical Thinking and Character while taking into consideration their needs and circumstances.

In addition to the material in the main text, you will also see this box in the body of the module:

Notes to the Teacher

This contains helpful tips or strategies that will help you in guiding the learners.

As a facilitator you are expected to orient the learners on how to use this module. You also need to keep track of the learners' progress while allowing them to manage their own learning. Moreover, you are expected to encourage and assist the learners as they do the tasks included in the module.

Computer Programming

11

Quarter 2

Self Learning Module 10

JavaScript Variables, Operators and Data Types

Writer: Magiel L. Boncayao
Editor: Ma. Lerma I. Cantanero
Validator/Reviewer: Rowena O. Dimagiba

For the Learner:

Welcome to the Computer Programming for the ICT Module on JavaScript Variables, Operators and Data Types.

The hand is one of the most symbolized part of the human body. It is often used to depict skill, action and purpose. Through our hands we may learn, create and accomplish. Hence, the hand in this learning resource signifies that you as a learner is capable and empowered to successfully achieve the relevant competencies and skills at your own pace and time. Your academic success lies in your own hands!

This module was designed to provide you with fun and meaningful opportunities for guided and independent learning at your own pace and time. You will be enabled to process the contents of the learning material while being an active learner.

This module has the following parts and corresponding icons:

Expectation - These are what you will be able to know after completing the lessons in the module

Pre-test - This will measure your prior knowledge and the concepts to be mastered throughout the lesson.

Recap - This section will measure what learnings and skills that you understand from the previous lesson.

Lesson- This section will discuss the topic for this module.

Activities - This is a set of activities you will perform.

Wrap Up- This section summarizes the concepts and applications of the lessons.

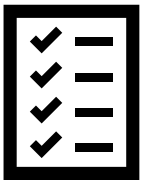
Valuing-this part will check the integration of values in the learning competency.

Post-test - This will measure how much you have learned from the entire module. Ito po ang parts ng module.

EXPECTATION

At the end of the module the learner is expected to:

1. understand the JavaScript Variables, Operators and Data Types;
2. recognize how JavaScript Variables, Operators and Data Types works, and
3. apply JavaScript Variables, Operators and Data Types in an HTML Document.



PRE–TEST

Instructions: Select the letter that corresponds to the correct answer.

1. Which of the following JavaScript are the containers for storing data values?

   A. Statement B. Comments C. Variables D. Operators

2. Which of the following JavaScript that are case-sensitive and must be unique names?

   A. Identifiers B. Variable C. Values D. Language

3. Which of the following is an example of assignment Operator?

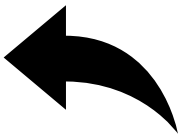
   A. - B. = C. + D. *

4. Which of the following data types is called text values?

   A. Data Types B. Numbers C. String D. Variables

5. What do you call a variable that declared without a value?

   A. Non-variable B. Undefined C. No Value D. No string



RECAP

Last module we discussed, JavaScript Syntax, Statement, Variables, Keywords, Operators, Expression, Values, Identifiers and Comments. We learned the different syntax of it. In this activity give an example syntax for the following JavaScript.

JavaScript Expressions _____

JavaScript Operators _____

JavaScript Keywords _____

JavaScript Statement _____

JavaScript Variables _____

LESSON

JavaScript Variables, Operators and Data types

JavaScript variables are containers for storing data values.

In this example, x, y, and z, are variables:

```
var x = 10;
var y = 20;
var z = x + y;
```

From the example above, you can expect:

- x stores the value 10
- y stores the value 20
- z stores the value 30

Example syntax:

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

<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>

</body>
</html>
```

Result:

JavaScript Variables

In this example, x, y, and z are variables.

The value of z is: 11

Much Like Algebra

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

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

- In programming, just like in algebra, we use variables (like price1) to hold values.
- In programming, just like in algebra, we use variables in expressions (total = price1 + price2).
- From the example above, you can calculate the total to be 11.

JavaScript Identifiers

All JavaScript **variables** must be **identified** with **unique names**. These unique names are called **identifiers**. Identifiers can be short names (like x and y) or more descriptive names (age, sum, totalVolume). Identifiers is case sensitive, also you can't use Keyword as a Identifiers or name.

The Assignment Operator

In JavaScript, the equal sign (=) is an "assignment" operator, not an "equal to" operator. This is different from algebra. The following does not make sense in algebra:

Example:

```
x = x + 5
```

- In JavaScript, however, it makes perfect sense: it assigns the value of x + 5 to x.
- (It calculates the value of x + 5 and puts the result into x. The value of x is incremented by 5.)

The "equal to" operator is written like == in JavaScript.

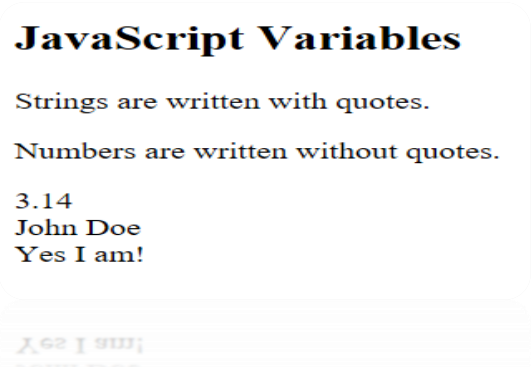
JavaScript Data Types

JavaScript variables can hold numbers like 100 and text values like "John Doe". In programming, text values are called text strings. JavaScript can handle many types of data, but for now, just think of numbers and strings. Strings are written inside double or single quotes. Numbers are written without quotes, and If you put a number in quotes, it will be treated as a text string

Example:

```
 <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>
```

Result:



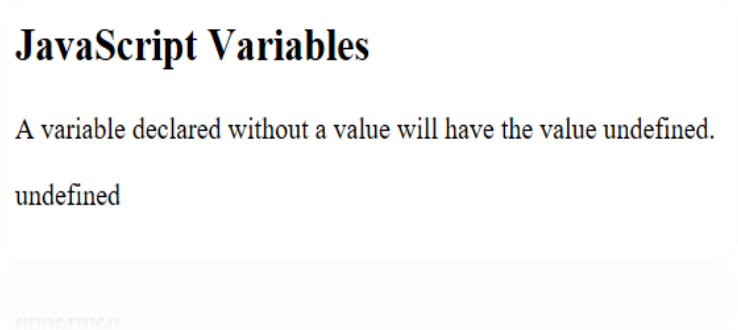
Value = Undefined

In computer programs, variables are often declared without a value. The value can be something that has to be calculated, or something that will be provided later, like user input. A variable declared without a value will have the value undefined. The variable carName will have the value undefined after the execution of this statement:

Example:

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

<h2>JavaScript Variables</h2>
<p>A variable declared without a
value will have the value undefined.</p>

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

<script>
var carName;
document.getElementById("demo")
.innerHTML = carName;
</script>

</body>
</html>
```

Result:



Re-Declaring JavaScript Variables

The variable carName will still have the value "Volvo" after the execution of these statements: If you re-declare a JavaScript variable, it will not lose its value.

JavaScript Arithmetic

As with algebra, you can do arithmetic with JavaScript variables, using operators like = and +:

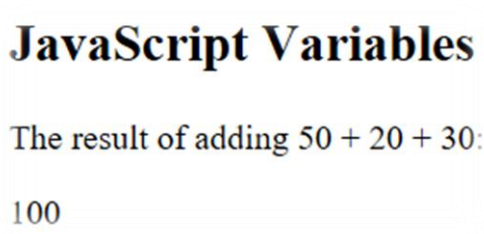
Example:

var x = 5 + 2 + 3;

Example:

```
<p id="demo"></p>
<script>
var x = 50 + 20 + 30;
document.getElementById("demo")
.innerHTML = x;
</script>
```

Result:



You can also add strings, but strings will be concatenated:

Example:

var x = "John" + " " + "Doe";

If you put a number in quotes, the rest of the numbers will be treated as strings, and concatenated.

Examples:

Try this: **var x = "5" + 2 + 3;**	**Result:** **523**
How about this: **var x = 2 + 3 + "5";**	**Result;** **55**

JavaScript Operators

JavaScript Arithmetic Operators

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

A typical arithmetic operation operates on two numbers. The two numbers can be literals:

var x = 100 + 50; or *variables*: var x = a + b;

or *expressions*: var x = (100 + 50) * a;

Example:

Adding

```
var x = 5;
var y = 2;
var z = x + y;
```

Subtracting

```
var x = 5;
var y = 2;
var z = x - y;
```

Multiplying

```
var x = 5;
var y = 2;
var z = x * y;
```

Dividing

```
var x = 5;
var y = 2;
var z = x / y;
```

Remainder

The modulus operator (%) returns the division remainder.

```
var x = 5;
var y = 2;
var z = x % y
```

Incrementing

The **increment** operator (++) increments numbers.

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

Decrementing

The **decrement** operator (--) decrements numbers.

```
var x = 5;
x--;
var z = x;
```

Operator Precedence

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

```
var x = 100 + 50 * 3;
```

Example of PMDAS

```
<!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>
```

Result:

Multiplication has precedence over addition.

250

Example of PMDAS

```
var x = (100 + 50) * 3;
```

x = 450

JavaScript Assignment Operators

Operator	Example	Same As
=	x = y	x = y
+=	x += y	x = x + y

Example:

```
var x = 10;
x += 5;
```

```
var x = 10;
x /= 5;
```

```
var x = 10;
x -= 5;
```

```
var x = 10;
x %= 5;
```

```
var x = 10;
x *= 5;
```

JavaScript String Operators

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

```
var txt1 = "John";
var txt2 = "Doe";
var txt3 = txt1 + " " + txt2;
```

The result of txt3 will be: John Doe

When used on strings, the + operator is called the concatenation operator, and if you add a number and a string, the result will be a string!

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

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.

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

Example:

```
var x = 16 + 4 + "Volvo";
```

Result: 20Volvo

Example:

```
var x = "Volvo" + 16 + 4;
```

Result: Volvo164

Example:

```
var x = 5 + “2” + 3;
```

Result: 523

JavaScript Strings

A string (or a text string) is a series of characters like "John Doe". Strings are written with quotes. You can use single or double quotes.

```
var carName = "Volvo XC60";     // Using double quotes
var carName = 'Volvo XC60';     // Using single quotes
```

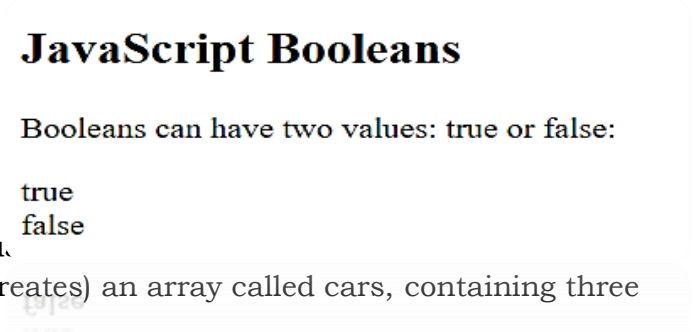
JavaScript Booleans

Booleans can only have two values: true or false. Booleans are often used in conditional testing.

Example:

```
var x = 5;
var y = 5;
var z = 6;
(x == y)       // Returns true
(x == z)       // Returns false
```

Result:



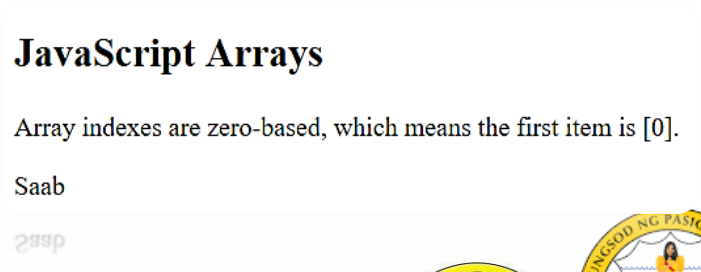
JavaScript Arrays

JavaScript arrays are written with squ
commas. The following code declares (creates) an array called cars, containing three items (car names).

Example:

```
<script>
var cars = ["Saab","Volvo","BMW"];
document.getElementById("demo")
.innerHTML = cars[0];
</script>
```

Result:



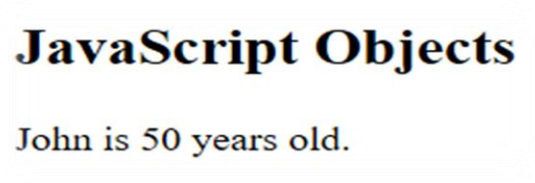
JavaScript Objects

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

Example:

```
<script>
var person = {
    firstName : "John",
    lastName  : "Doe",
    age           : 50,
    eyeColor   : "blue"
};
document.getElementById("demo").innerHTML =
person.firstName + " is " + person.age + " years old.";
</script>
```

Result:



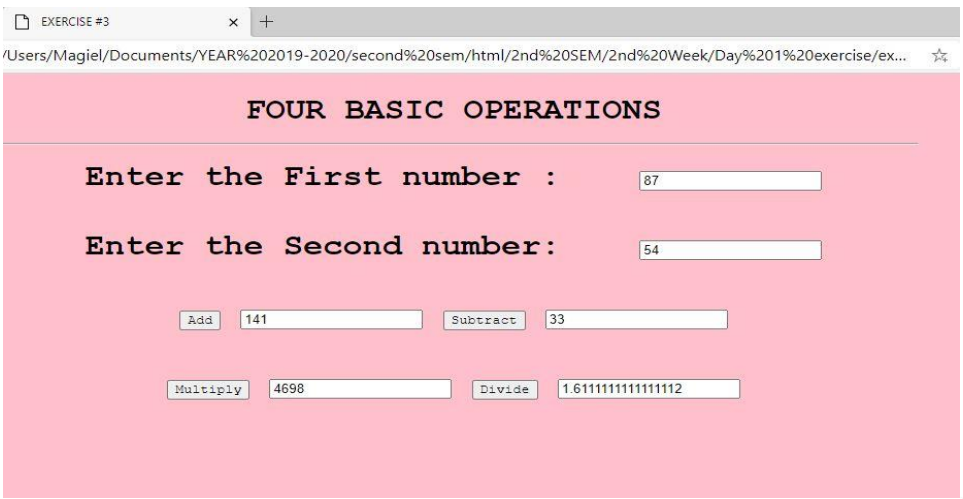
ACTIVITIES

JavaScript

Create a simple calculator that will only compute the 4-basic operation like Addition, subtraction, multiplication, and division. Make sure that you also include to use external JavaScript in this activity.

1. Create a HTML documents and use External JavaScript.
2. Put Title “Four Basic Operations” align it in center.
3. Set any background color you want.
4. Use HTML Button style and Input type also use the onclick button.
5. Use the Courier as a Font Family.

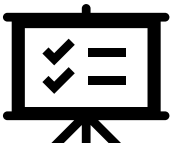
Example:



Rubrics:

For Activity 10 the students will get total of 50 points.

Items	Points
1. Title, and HTML document	10
2. External JavaScript	10
3. Background and font-family	10
4. HTML Onclick Button and input type	10
5. JavaScript syntax	10
TOTAL	**50 points**

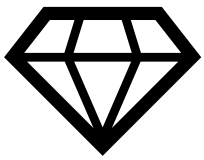


WRAP–UP

In this module we discussed JavaScript Variables, Operators, and Data Types. We learned the different syntax that we can use to make our website functionable.

In this activity student will create a variable **w, x, y**, and **z** assign the 1st value to "**Happy**", the 2nd value put your "**age**", 3rd value to **"Birthday"**, then the last value just add the first 3 values and don't forget to display it.

```
<body>
<p id="demo">Display the result here.</p>
<script>
// Create the variable here
_____________________________________________
</script>
</body>
```



VALUING

Instructions: Carefully read the following questions and provide two to three sentences answer to each number.

1. Why do you think JavaScript Variable is important?
   _____
   _____
2. What do you think are the advantages of knowing the operators in making a website?
   _____
   _____

POST TEST

INSTRUCTIONS: Select the letter that corresponds to the correct answer.

1. Which of the following JavaScript that are case-sensitive and must be unique names?
   a. A. Identifiers B. Variable C. Values D. Language

2. What do you call a variable that declared without a value?

   a. A. Non-variable B. Undefined C. No Value D. No string

3. Which of the following data types is called text values?

   a. A. Data Types B. Numbers C. String D. Variables

4. Which of the following JavaScript are the containers for storing data values?
   a. A. Statement B. Comments C. Variables D. Operators

5. . Which of the following is the example of increment Operator?

   a. A. **--** B. **==** C. **++** D. **%**

KEY TO CORRECTION

Pre-test:	Post-test:
1. C	1. A
2. A	2. B
3. B	3. C
4. C	4. C
5. B	5. C

REFERENCES

Websites

- Accessed September 24 2020 1:20 am
  https://www.w3schools.com/