DepED
DEPARTMENT OF EDUCATION

COMPUTER PROGRAMMING
(ICT)

Grade 11

QUARTER 2

MODULE 9

JavaScript Syntax, Statement and Comments

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 9: JavaScript Syntax, Statement and Comments.

Published by the Department of Education Division of Pasig City

Development Team of the Module

Writer : Magiel L. Boncayao

Editor : Ma. Lerma I. 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 Syntax, Statement and Comments.

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 9

JavaScript Syntax, Statement and Comments

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

For the Learner:

Welcome to the Computer Programming for the ICT Module on JavaScript Syntax, Statement and Comments.

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

The students should be able to:

- understand the JavaScript, Syntax, Statement and Comments,
- recognize how JavaScript Syntax, Statement and Comments works,
- apply JavaScript Syntax, Statement and Comments in an HTML Document.

PRE–TEST

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

1. Which of the following JavaScript syntax that are composed of Values, Operators, Expressions, Keywords and Comments?

   A. Statement B. Comments C. Values D. Language

2. Which of the following JavaScript syntax that defines two types of values: Fixed values and variable values?

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

3. Which of the following JavaScript syntax that are used to store data values?

   A. Statement B. Variable C. Values D. Expressions

4. Which of the following JavaScript syntax that combines of Values, Operators, Expressions, Keywords and Comments to evaluate

   A. Statement B. Variable C. Values D. Expressions

5. Which of the following JavaScript syntax that can be a statement but not executed?

   A. Statement B. Comments C. Values D. Language

RECAP

Last module we discussed JavaScript, we learned the introduction of JavaScript, we learned some basic syntax and the placement of JavaScript. We Learned also that like CSS, JavaScript can also be a external files.

For this activity students will get 5 points. All they need to do is to enumerate the 5 things that JavaScript can do.

1. _____

2. _____
3. _____
4. _____
5. _____

LESSON

JavaScript Syntax, Statements and Comments

JavaScript syntax

is the set of rules, how JavaScript programs are constructed.

JavaScript Programs

A **computer program** is a list of "instructions" to be "executed" by the computer.

In a programming language, these program instructions are called **statements**.

JavaScript is a **programming language**.

JavaScript statements are separated by **semicolons**

Example:

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

In HTML, JavaScript programs are executed by the web browser.

JavaScript Statements

JavaScript statements are composed of: Values, Operators, Expressions, Keywords, and Comments.

JavaScript Values

The JavaScript syntax defines two types of values: Fixed values and variable values. Fixed values are called **literals**. Variable values are called **variables**.

JavaScript Literals

The most important rules for writing fixed values are:

Numbers are written with or without decimals:

Example:

```
10.50 (not like this)

1001 (it should be like this)
```

Strings are text, written within double or single quotes:

Example:

```
"John Doe"

'John Doe'
```

JavaScript Variables

In a programming language, **variables** are used to **store** data values. JavaScript uses the **var** keyword to **declare** variables. An **equal sign** is used to **assign values** to variables.

In this example, x is defined as a variable. Then, x is assigned (given) the value 6:

Example:

```
var x;

x = 6;
```

JavaScript Operators

JavaScript uses **arithmetic operators** (+ - * /) to **compute** values:

Example:

```
(5 + 6) * 10
```

JavaScript uses an **assignment operator** (=) to **assign** values to variables:

Example:

```
var x, y;
x = 5;
y = 6;
```

JavaScript Expressions

An expression is a combination of values, variables, and operators, which computes to a value. The computation is called an evaluation.

For example, 5 * 10 evaluates to 50:

<body>

<h2>JavaScript Expressions</h2>

```
<p>Expressions compute to values.</p>
<p id="demo"></p>
<script>
document.getElementById("demo").innerHTML = 5 * 10;
</script>
</body>
</html>
```

Expressions can also contain variable values:

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

Both Result looks like this

The values can be of various types, such as numbers and strings.

For example, "John" + " " + "Doe", evaluates to "John Doe":

```
<body>
<h2>JavaScript Expressions</h2>
<p>Expressions compute to values.</p>
<p id="demo"></p>
<script>
document.getElementById("demo").innerHTML = "John" + " "  + "Doe";
</script>
```

JavaScript Keywords

JavaScript **keywords** are used to identify actions to be performed.

The **var** keyword tells the browser to create variables:

```
<p id="demo"></p>
<script>
var x, y;
x = 5 + 6;
y = x * 10;
document.getElementById("demo").innerHTML = y;
</script>
```

JavaScript Comments

Not all JavaScript statements are "executed".

Code after double slashes **//** or between **/*** and ***/** is treated as a **comment**.

Comments are ignored, and will not be executed

Example:

```
<p id="demo"></p>
<script>
var x;
x = 5;
// x = 6; I will not be executed
document.getElementById("demo").innerHTML = x;
</script>
```

JavaScript Identifiers

Identifiers are names.

In JavaScript, identifiers are used to name variables (and keywords, and functions, and labels). The rules for legal names are much the same in most programming languages. In JavaScript, the first character must be a letter, or an underscore (_), or a dollar sign ($). Subsequent characters may be letters, digits, underscores, or dollar signs.

Numbers are not allowed as the first character. This way JavaScript can easily distinguish identifiers from numbers.

JavaScript is Case Sensitive

All JavaScript identifiers are **case sensitive**. The variables **lastName** and **lastname**, are two different variables.

Example:

```
var lastname, lastName;
lastName = "Doe";
lastname = "Peterson";
```

JavaScript does not interpret **VAR** or **Var** as the keyword **var**.

JavaScript and Camel Case

Historically, programmers have used different ways of joining multiple words into one variable name:

Hyphens:

first-name, last-name, master-card, inter-city. However, Hyphens are not allowed in JavaScript. It is reserved for subtractions.

Underscore:

first_name, last_name, master_card, inter_city.

Upper Camel Case (Pascal Case):

FirstName, LastName, MasterCard, InterCity.

JavaScript programmers tend to use camel case that starts with a lowercase letter:

firstName, lastName, masterCard, interCity.

Semicolons;

Semicolons separate JavaScript statements. Add a semicolon at the end of each executable statement:

Example:

```
var a, b, c;
a = 5;
b = 6;
c = a + b;
```

When separated by semicolons, multiple statements on one line are allowed, On the web, you might see examples without semicolons. Ending statements with semicolon is not required, but highly recommended.

JavaScript Line Length and Line Breaks

For best readability, programmers often like to avoid code lines longer than 80 characters. If a JavaScript statement does not fit on one line, the best place to break it, is after an operator:

JavaScript Code Blocks

JavaScript statements can be grouped together in code blocks, inside curly brackets {...}. The purpose of code blocks is to define statements to be executed together. One place you will find statements grouped together in blocks, is in JavaScript functions:

Example:

```
function myFunction() {
    document.getElementById("demo1").innerHTML = "Hello Dolly!";
    document.getElementById("demo2").innerHTML = "How are you?";
}
```

JavaScript Keywords

JavaScript statements often start with a **keyword** to identify the JavaScript action to be performed. JavaScript keywords are reserved words. Reserved words cannot be used as names for variables.

Example:

Function - Declares a function

if ... else - Marks a block of statements to be executed, depending on a condition

Single Line Comments

Single line comments start with //.

Any text between // and the end of the line will be ignored by JavaScript (will not be executed).

This example uses a single-line comment before each code line:

Example

```
// Change heading:
document.getElementById("myH").innerHTML = "My First Page";
// Change paragraph:
document.getElementById("myP").innerHTML = "My first paragraph.";
```

Multi-line Comments

Multi-line comments start with /* and end with */.

Any text between /* and */ will be ignored by JavaScript.

This example uses a multi-line comment (a comment block) to explain the code:

Example

```
/*
The code below will change
the heading with id = "myH"
and the paragraph with id = "myP"
in my web page:
*/
document.getElementById("myH").innerHTML = "My First Page";
document.getElementById("myP").innerHTML = "My first paragraph.";
```

ACTIVITIES

JavaScript

The following table lists are JavaScript, all you need to do is to explain the function and give syntax of the following. For this activity, the student will get 20 points

JavaScript	Functions	Syntax
JavaScript Statement		
JavaScript Variables		
JavaScript Value		
JavaScript Operator		
JavaScript Expressions		
JavaScript Keywords		
JavaScript Comments		
JavaScript Identifiers		
JavaScript Code Blocks		
Multi-line Comments		

WRAP–UP

In this 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 the syntax of the following JavaScript.

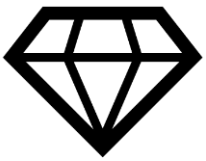
Single Comments _____

Multi-line Comments _____

String Value _____

Number Value _____

Operators _____



VALUING

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

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

POST TEST

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

1. Which of the following JavaScript syntax that combines of Values, Operators, Expressions, Keywords and Comments to evaluate
   A. Statement B. Variable C. Values D. Expressions

2. Which of the following JavaScript syntax are composed of Values, Operators, Expressions, Keywords and Comments?
   A. Statement B. Comments C. Values D. Language

3. Which of the following JavaScript syntax can be a statement but not executed?
   A. Statement B. Comments C. Values D. Language

4. Which of the following JavaScript syntax defines two types of values: Fixed values and variable values?

   A. Statement B. Variable C. Values D. Language
5. Which of the following JavaScript syntax are used to store data values?
   A. Statement B. Variable C. Values D. Expressions

KEY TO CORRECTION

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

REFERENCES

Websites

- Accessed September, 19 2020 1:10 am
  https://www.w3schools.com/