ST0523 Fundamentals of Programming

Topic 1c :

Introduction to JavaScript Programming & Data Types

SP Singapore Polytechnic

Introduction To JavaScript Programming & Data Types

- To know the basic syntax of a JavaScript program.
- To apply the good programming practices.
- To identify the primitive data types of JavaScript.

A Simple JavaScript Program

Example 1: Guess what does the program do?

```
console.log( “Hello World” );
```

This statement will display
“Hello World” in the terminal.

The **console.log()** is a function in JavaScript which is used to print any message (encapsulated within the inverted comma) or any variables defined before in it to the user.

Escape Sequence

JavaScript uses the \ (backslash) as an escape characters for:

Description	***Escape Sequence***
Backspace	\b
Tab	\t
Linefeed	\n
Blackslash	\\
Single Quote	\’
Double Quote	\”

```
console.log( “Hello\nWorld” );
```

Output:

```
Hello
World
```

Comments

- Make programs more readable and easier to work on.
- Ignored by compiler and does not affect program output.
- Single line comment:

```
// some text, are preceded by two slashes (//) in a line
```

When the compiler sees //, it ignores all text after // in the same line.

- Multi-line comment:

```
/*
     some text, are enclosed between /* and */ in one
      or multiple lines
*/
```

When the compiler sees /*, it scans for the next */ and ignores any text between /* and */.

Comment Example

```
// This line will not be executed.
// console.log( “Hello World” );
```

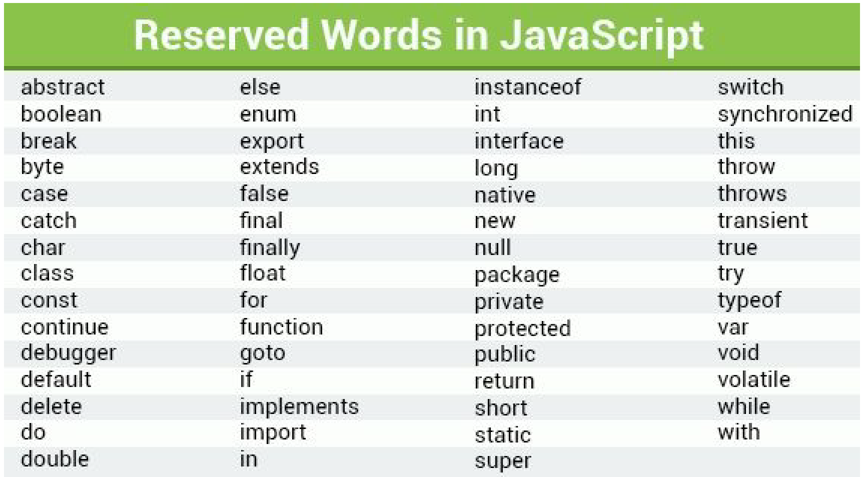
This statement will not be executed as it’s commented off.

```
/*
    Every line within /* */ will not be executed.
    console.log( “Happy Birthday” );
    console.log( “To You!” );
 */
```

Reserved Words

Reserved words or keywords are the words that have already been taken by the JavaScript Engine. They have a specific meaning to the compiler and cannot be used for other purposes.

Therefore, you are not allowed to use any of the words to name your variables/ functions/ classes.



Reserved Words in JavaScript

abstract	else	instanceof	switch
boolean	enum	int	synchronized
break	export	interface	this
byte	extends	long	throw
case	false	native	throws
catch	final	new	transient
char	finally	null	true
class	float	package	try
const	for	private	typeof
continue	function	protected	var
debugger	goto	public	void
default	if	return	volatile
delete	implements	short	while
do	import	static	with
double	in	super	

E.g. console is a prebuilt class and log is a built-in function).

Programming Example

Say for example we want to display the area of triangle, how can we write a program to do so? Let’s take a look at the following codes.

```
var area;    // Variable declaration

area = 6;    // Assign an area

console.log( "The area is " + area );
```

Variables

A variables can be thought of as named containers.

You can put data into these containers.

You can refer (or retrieve) the data in the container simply by mentioning the name of the container in your program code.

Declaring a variable

Many languages require that a variable be declared (**defined**) before it is first used.

Variables are declared with the **var** keyword as follows:

```
var area;     // Variable declaration
```

You can also declare multiple variables with the same var keyword as follows:

```
var areaOfSquare, areaOfTriangle;
```

Declaring a variable

Although JavaScript allows variable declaration, it does not require it - except in the case when we want to declare a variable being local (more on local variables in the later topics).

However, it is **good programming practice** to declare variables before using them

Identifiers

Identifiers are names. They are used to name variables (functions, keywords and labels).

An identifier is a sequence of characters that consist of letters, digits, underscores (_), and dollar signs ($).

An identifier must start with a letter, an underscore (_), or a dollar sign ($). It cannot start with a digit. Subsequent characters may be letters, digits, underscores, or dollar signs.

An identifier cannot be a reserved word such as true, false, null, var.

An identifier can be of any length and not have any spaces.

Identifier Example

In the below example, area is the identifier.

```
var area;      // Variable declaration

area = 6;      // Assign an area

console.log( "The area is " + area );
```

Initializing variables

Storing a value in a variable for the first time is called variable initialization.

You can do variable initialization at the time of variable creation using equal sign (=) as follow:

```
var area = 5;     // Declare and initialize
```

or you can do it at a later point in time e.g.

```
var area;          // Variable declaration
area = 5;          // Assign an area
```

Re-initializing variables

A variable’s value can be re-initialized and we will always look at the latest value of the variable.

```
var area = 5;    // Declare and initialize
area = 8;        // Re-initialize area to 8
```

Now, the current value of area is 8.

Good Programming Practices

- Include the var keyword when declaring of variables.
- Ending each of your statements with a semi-colon.
- Naming Convention for variables should be meaningful and descriptive.
    - Use lowercase.
    - If the name consists of several words, concatenate all in one, use lowercase for the first word, and capitalize the first letter of each subsequent word in the name. (Camel Case)

Good Programming Practices

- Appropriate comments to be included at the beginning of the program to explain what the program does, its key features, its supporting data structures, and any unique techniques it uses.

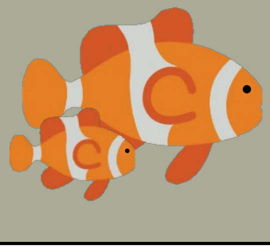
- Proper Indentation & Spaces
  - Indent two spaces.
  - Use blank line to separate segments of the code.

Data Types

- JavaScript variables can hold numerical values like 100 and text values like “Hello World”. These text values are also called text strings.
- We will take a look at the 6 different primitive data types of JavaScript using the example below.

Clownfish

Amphiprioninae



Name:	“Martin”
Age:	6 years old
Weight:	340.12 grams
Extinct:	False
Gender:	Undefined
Horoscope:	Null

Data Types – String

- What values
  - series of characters
  - any length including the zero-length (“”)
- What operations
  - concatenation (+ operator)
- Syntax
  - double or single quotes
    e.g. ‘hello’ or “hello”

Name:	“Martin”
Age:	6 years old
Weight:	340.12 grams
Extinct:	False
Gender:	Undefined
Horoscope:	Null

Data Types – Integer

- What values
  - whole numbers between -9223372036854775808 and 9223372036854775808
- What operations
  - standard mathematical operations (+, -, *, /)
  - special Math functions
- Examples

  20, 30, 5, -20

Name:	“Martin”
Age:	6 years old
Weight:	340.12 grams
Extinct:	False
Gender:	Undefined
Horoscope:	Null

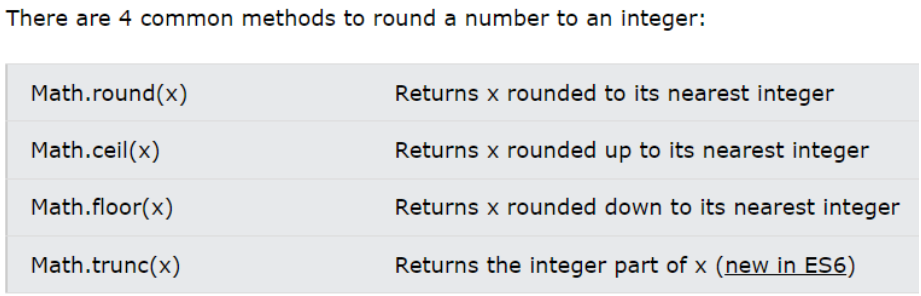
Data Types – Floating-Point

- What values
  - decimal values from ±1.0x10^{308} to ± 1.0x10^{-323}
  - 17 digits of precision (past decimal point)
- What operations
  - standard mathematical operations
  - special Math functions
- Examples
  - 1.5, -0.256, 1.2e3, 103

Name:	"Martin"
Age:	6 years old
Weight:	340.12 grams
Extinct:	False
Gender:	Undefined
Horoscope:	Null

Special Math functions

- JavaScript Math functions/methods allows you to perform mathematical tasks on numbers, for example :



There are 4 common methods to round a number to an integer:

Math.round(x)	Returns x rounded to its nearest integer
Math.ceil(x)	Returns x rounded up to its nearest integer
Math.floor(x)	Returns x rounded down to its nearest integer
Math.trunc(x)	Returns the integer part of x (new in ES6)

- Google to find out more!
  - Math.floor(58.3) results in 58
  - Math.sqrt(100) results in 10

Reference : JavaScript Math Object (w3schools.com)

Data Types – Boolean

- What values
  - true / false
- What operations
  - logical operations (and, or, not)
- Syntax
  - true, false

Name:	“Martin”
Age:	6 years old
Weight:	340.12 grams
Extinct:	False
Gender:	Undefined
Horoscope:	Null

Data Types – Undefined

- What values
  - Not assigned to a value but variable is declared **OR**
  - Variable not declared **OR**
  - Variable doesn’t exist
- Example
  - var gender;

Name:	“Martin”
Age:	6 years old
Weight:	340.12 grams
Extinct:	False
Gender:	Undefined
Horoscope:	Null

Data Types – Null

- What values
  - No value or empty value
- Example
  - var horoscope = null;

- You assign null to a variable when you want to ensure that variable does not contain any data.

Name:	“Martin”
Age:	6 years old
Weight:	340.12 grams
Extinct:	False
Gender:	Undefined
Horoscope:	Null

Data Types – Reference Type

- JavaScript also supports reference or composite data types such as object, array, function.

- Reference data types can contain multiple values or complex types of information as opposed to single values stored in primitive data types.

- More will be discussed in the later topics.

Data Types – Exercise

- Determine the most suitable data types for each of the following categories.

String	Integer	Float	Boolean

Country

Stock Price

Weight

Age

Validity of Email

Admin No.

Postal Code

Initializing variables

Note that JavaScript is untyped language – meaning that JavaScript variable can hold a value of any data type. The value type of a variable can change during the execution of a program. E.g.

```
var x;            // x is undefined here
x = 9999;         // numeric type
console.log(x);   // 9999 is printed

x = "hello";      // string type
console.log(x);   // hello is printed
```

Introduction To JavaScript Programming & Data Types

- To know the basic syntax of a JavaScript program.
- To apply the good programming practices.
- To identify the primitive data types of JavaScript.

SP SCHOOL OF Computing