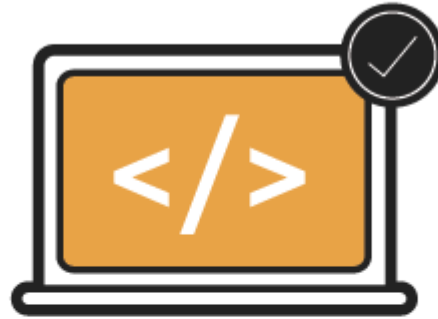


LEARN. DO. EARN

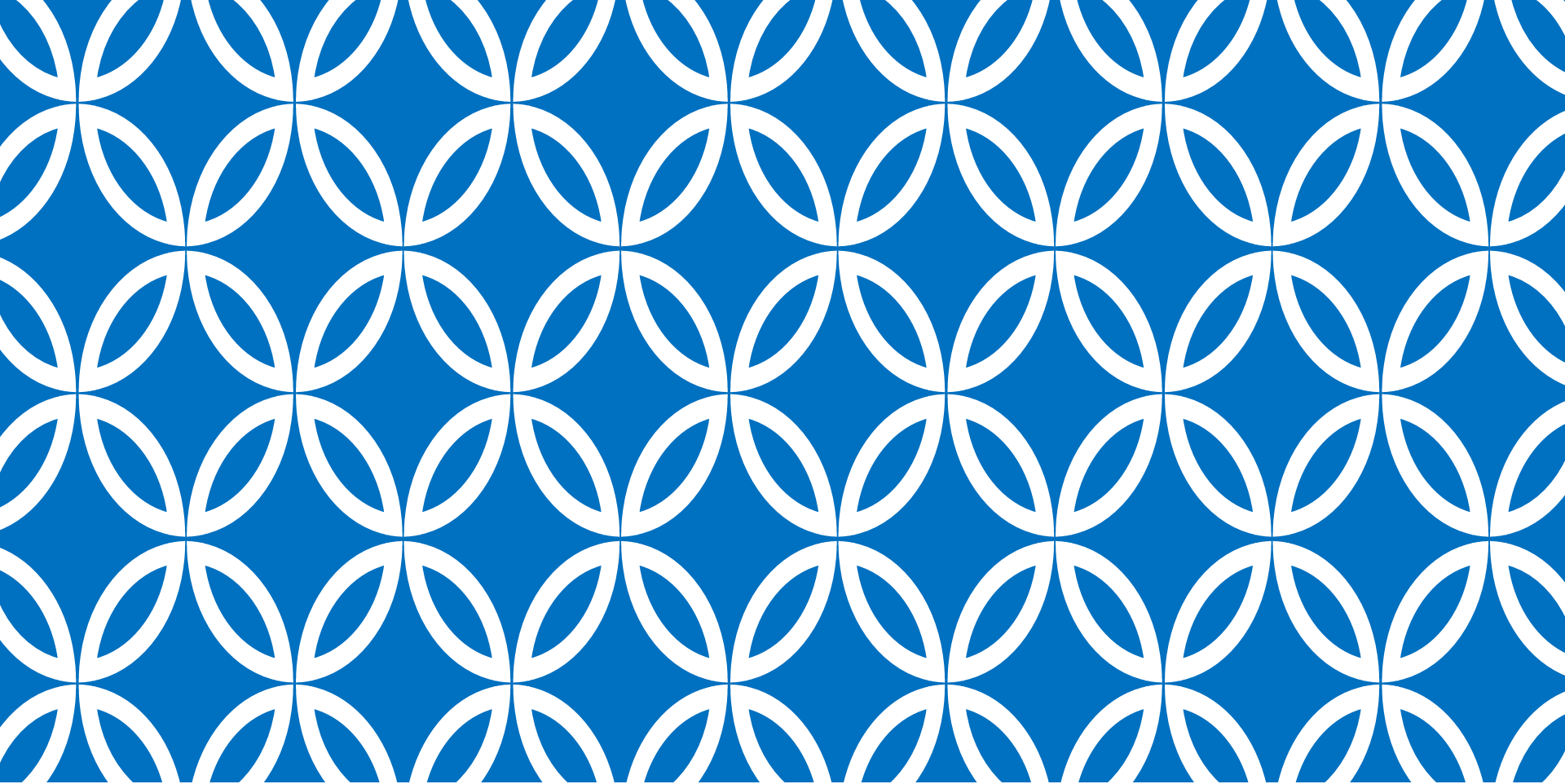
ACADGILD



FRONT END WEB DEVELOPMENT FUNDAMENTALS

Website : <http://www.acadgild.com>
LinkedIn : <https://www.linkedin.com/company/acadgild>
Facebook : <https://www.facebook.com/acadgild>

© copyright ACADGILD



Session 4 – JavaScript





Agenda – JavaScript

1. **What is JavaScript**
2. **Why Learn JavaScript**
3. **First JavaScript Program**
4. **Variable**
5. **Identifiers**
6. **Keywords**
7. **Data Types**
8. **Example of Data Types**
9. **Operators**
10. **Where To Place JavaScript**





What is JavaScript

- **JavaScript** is a full-fledged programming language that can be applied to an HTML document to create **dynamic interactivity on websites**.
- It was invented by **Brendan Eich**, co-founder of the Mozilla project, the Mozilla Foundation and the Mozilla Corporation.





Why Learn JavaScript

- **JavaScript** is one of the **3 languages** all web developers **must** learn:
 - **HTML** to define the structure/content of web pages
 - **CSS** to specify the layout/design of web pages
 - **JavaScript** to program the behavior of web pages
- It's free and easy to set up.
- It's easy to debug.
- It's powerful.
- It's execution speed is fast.
- Is widely used.
- In Demand and Thriving User Community.





First JavaScript Program

- **JavaScript** is written in html file inside a script tag.
- alert is a function to display popup message to user.
- The plus (+) operator is used to concatenate strings.

```
<html>
  <head>
    <script type = text/javascript">
      Var name = "Smith";
      Var age = 29;
      alert("The name is "+name + "And age is "+age);
    </script>
  </head>
  <body> </body>
</html>
```





Variable

- **Variables** are containers that you can store values in.
- Declaring a variable with the var keyword, followed by any name you want to call it:

Syntax : `var variableName;`

- **Rule :** They must begin with a letter or the underscore character.
- **JavaScript** is an **un-typed programming language**.
- Semicolons in JavaScript terminates a statement.





JavaScript Identifiers

- **Identifiers** are used to name variables, keywords, functions and labels.
- The rules for legal names are much the same in most programming languages.
- **Rules:**
 - First character must be a letter, an underscore (_), or a dollar sign (\$).
 - Subsequent characters may be letters, digits, underscores, or dollar sign.
 - Numbers are not allowed as the first character.
 - Reserved words (like JavaScript keywords) cannot be used as names.
- All JavaScript identifiers are **case sensitive**.
- **Hyphens are not allowed** in JavaScript. *It is reserved for subtractions.*





JavaScript Keywords

- **JavaScript** keywords are reserved words.
- *Reserved words cannot be used as names for variables.*

Keywords	Description
break	Terminates a switch or a loop
continue	Jumps out of a loop and starts at the top of the loop
debugger	Stops the execution of JavaScript and calls (if available) the debugging function
do...while	Executes a block of statements, and repeats the block, while a condition is true
for	Marks a block of statements to be executed, as long as a condition is true
function	Declares a function
if...else	Marks a block of statements to be executed, depending on a condition
return	Exits a function
switch	Marks a block of statements to be executed, depending on different cases
try...catch	Implements error handling to a block of statements
var	Declares a variable





Data Types

- JavaScript allows the same variable to contain different types of data values.
- **Primitive data types:**
 - **Number:** integer & floating-point numbers
 - **Boolean:** logical values “true” or “false”
 - **String:** a sequence of alphanumeric characters
- **Composite data types (or Complex data types):**
 - **Object:** a named collection of data
 - **Array:** a sequence of values
- **Special data types:**
 - **Null:** an initial value is assigned
 - **Undefined:** the variable has been created but not yet assigned a value





Examples of Data Types

Variable Data Types	Explanation	Example
String	A string of text. To signify that the variable is a string, you should enclose it in quote marks.	<code>var myVariable = 'Bob';</code>
Number	A number. Numbers don't have quotes around them.	<code>var myVariable = 10;</code>
Boolean	A True/False value. true/false are special keywords in JS, and don't need quotes.	<code>var myVariable = true;</code>
Array	A structure that allows you to store multiple values in one single reference.	<code>var myVariable = [1,'Bob','Steve',10];</code> Call each member of the array like this: <code>myVariable[0],myVariable[1], etc.</code>
Object	Everything in JavaScript is an object, and can be stored in a variable.	<code>var myVariable = document.querySelector('h1');</code> Note : All of the above examples too.





Operators

- **An operator** is basically a mathematical symbol that can act on two values (or variables) and produce a result.

Arithmetic operators	<code>+</code> , <code>-</code> , <code>/</code> , <code>*</code> , <code>%</code>
Logical operators	<code>&&</code> , <code> </code> , <code>!</code>
Comparison operators	<code>==</code> , <code>===</code> , <code>>=</code> , <code><=</code>
String operators	<code>+</code>
Bit-wise operators	<code>&</code> , <code>!</code> , <code>>></code> , <code><<</code>
Assignment operators	<code>+=</code> , <code>-=</code> , <code>/=</code> , <code>*=</code>





Where to Place JavaScript

- Scripts can be placed in 3 places in Web page:

- In the <head> section of an HTML page

```
<head>  
  <script>  
    alert("Hello");  
  </script>  
</head>
```

- In the <body> section of an HTML page

```
<body>  
  <script>  
    alert("Hello");  
  </script>  
</body>
```

- Import as an external file

```
<script src="Path_to_JavaScript_File"></script>
```





Agenda – JavaScript

SI No	Agenda Title
1	Conditionals
2	if Statement
3	if...else Statement
4	else if Statement
5	if/if...else Statement
6	switch Statement
7	Debugging





Conditionals

- **Conditionals** are code structures that allow you to test whether an expression returns true or not, and then run different code depending on the result.

These are:

- “if” statement
- “if ... else” statement
- “else if” statement
- “if/if ... else” statement
- “switch” statement





if Statement

- It is the main conditional statement in JavaScript.
- The keyword “if” always appears in lowercase.
- The condition yields a logical true or false value.
- If the condition is true then statements are executed.

Syntax : `if (condition) { statements; }`





if...else Statement

- You can include an “else” clause in an if statement when you want to execute some statements in case the condition is false.
- In other words, else statement is used to specify a block of code to be executed if the condition is false.

Syntax :

```
if (condition) { statements; }
```

```
else { statements; }
```





else if Statement

- Allows you to test for multiple expression for one true value and executes a particular block of code.
- else if statement is used to specify a new condition if the first condition is false.

Syntax:

```
if (condition) { statement; }  
  else if (condition) { statement; }  
    else { statement; }
```





if/if ... else statement

Syntax :

```
if (condition) {  
    if (condition) { statements; }  
    else { statements; }  
}
```





switch Statement

- Allows you to merge several evaluation tests of the same variable into a single block of statements.
- The **switch** statement is used to perform different actions based on different conditions.

Syntax:

```
switch (expression) {  
    case label1:  
        statements; break;  
    default:  
        statements;  
}
```





Debugging in JavaScript

- With the recent boom of JavaScript, all major browsers come with their own debug tools.

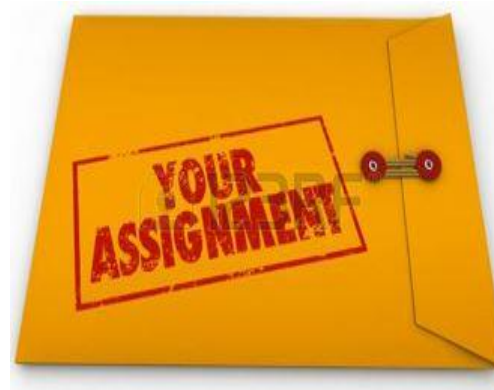
Examples:

- Chrome has **Chrome DevTools**. You can access it by shortcut key Ctrl+Shift+Alt.
- For Firefox you can use **firebug extension**.





Lets Discuss Assignments



Assignment

