

Javascript, PHP

H. Sahibzada

(in)Visible Networks
Hasibullah.sahibzada@uni-weimar.de
Bauhaus University Weimar

2nd May

Topics

1 JavaScript

2 PHP

JavaScript

- JavaScript is the programming language of HTML and the Web.
- HTML to define the content of web pages
- CSS to specify the layout of web pages
- JavaScript to program the behavior of web pages.

JavaScript

Basics

- JavaScript Can Change HTML Content.

```
document.getElementById("demo").innerHTML = "Hello  
JavaScript";
```

- JavaScript Can Change HTML Attributes.

```
document.getElementById("demo").width = "100";
```

- JavaScript Can Change HTML Styles (CSS).

```
document.getElementById("demo").style.fontSize = "25px";
```

- JavaScript Can Validate Data. (Often)

JavaScript

How to use?

- It separates HTML and code.
- It makes HTML and JavaScript easier to read and maintain
- Cached JavaScript files can speed up page loads
- Inside `<script>` Tag (in head, or body anywhere)
- External JavaScript, (myscript.js) in body or head tags
 - `<script src=" myScript.js" ></script>`

JavaScript

Output

- Alert box, `window.alert()`.
- writing in HTML output, (`document.write()`)
- Changing HTML content, `innerHTML`.
- Writing in browser console, `console.log()`

JavaScript

Syntax

- Computer program is a list of "instructions", (Statements)
- Statements ends with semicolons (;)
- Composed of **Values**, **Operators**, **Expressions**, **Keywords**, and **Comments**.

JavaScript

Values

- Literals (fixed values) "25, john"
- Variable values (Variables)
 - variables are used to store data values.
 - use *var* to declare variables.
 - An equal sign (=) for assigning value to variable.
 - *var age;*
 - *age = 25*
 - case sensitive.

JavaScript

Operators

- Assignment operator. (=, +=, *=,)
- arithmetic operators (+, -, *, /, %, ++, -).
- *var result;*
- *result = (5 + 6) * 10;*

JavaScript

Expressions

- An expression is a combination of values, variables, and operators, which computes to a value.
- *result = (5 + 6) * 10;*
- *var x =10;*
- *result = (5 + 6) * x;*
- *"John" + " " + "Doe"*

JavaScript

Keywords

- JavaScript keywords are used to identify actions to be performed.
- *var, void, if, int, float, class, char, function, for ..*

JavaScript

Comments

- To note something inside the code,
- Use slashes, `//` or `/* */`

JavaScript

Data types

- To be able to operate on variables, it is important to know something about the type.
- Without data types, a computer cannot safely do operations.
- Sequence of operations on variables produce different results.
- Number, *var length = 16;*
- String, *var lastName = "Johnson";*
- Booleans, *var x = true, / false*
- Array, *var cars = ["Saab", "Volvo", "BMW"];*
- Objects, *var x = {firstName:" John", lastName:" Doe"};*
- typeof operator, Finds the type of the variable.
- A variable without a value, has the value undefined. The typeof is also undefined.

JavaScript

Functions

- Function is a block of code designed to perform a particular task.
- A JavaScript function is executed when "something" invokes it (calls it).
- You can reuse code many times.
- Function has, Name, parentheses, parameters, body, (return statements)

```
function myFunction(p1, p2){  
return p1 * p2;  
}
```

JavaScript

Conditions

- You want to perform different actions for different decisions.
- **if condition:**
if (condition) {
block of code to be executed if the condition is true
}
- **else condition:**
if (condition) {
block of code to be executed if the condition is true
} else {
block of code to be executed if the condition is false
}
- **else if condition:**
- **switch condition:**

JavaScript

Switch

- Use the switch statement to select one of many blocks of code to be executed.

```
switch(expression) {  
    case n:  
        code block  
        break;  
    case n:  
        code block  
        break;  
    default:  
        default code block  
}
```


JavaScript

Events

- An HTML event can be something the browser does, or something a user does.
- HTML allows event handler attributes, with JavaScript code, to be added to HTML elements.
- `<some-HTML-element some-event='some JavaScript'>`
- `<button
onclick='getElementById("demo").innerHTML=Date()>The
time is?</button>`

JavaScript

HTML Events

- onchange (An HTML element has been changed)
- onclick (The user clicks an HTML element)
- onmouseover (The user moves the mouse over an HTML element)
- onmouseout (The user moves the mouse away from an HTML element)
- onkeydown (The user pushes a keyboard key)
- onload, (The browser has finished loading the page)

JavaScript

Debuggers

- Searching for errors in programming code is called code debugging.
- all modern browsers have a built-in debugger.
- you can also set breakpoints (places where code execution can be stopped)
- debugger keyword

Break

Break !

PHP

Introduction

- PHP is an acronym for "PHP: Hypertext Preprocessor"
- Server side language!
- PHP is a widely-used, open source scripting language
- PHP scripts are executed on the server.
- It is deep enough to run the largest social network (Facebook)!
- PHP files can contain text, HTML, CSS, JavaScript, and PHP code
- PHP code are executed on the server, and the result is returned to the browser as plain HTML
- PHP files have extension ".php"

PHP

Introduction

- PHP can generate dynamic page content
- PHP can create, open, read, write, delete, and close files on the server
- PHP can collect form data
- PHP can send and receive cookies
- PHP can add, delete, modify data in your database
- PHP can be used to control user-access
- PHP can encrypt data

PHP

Process

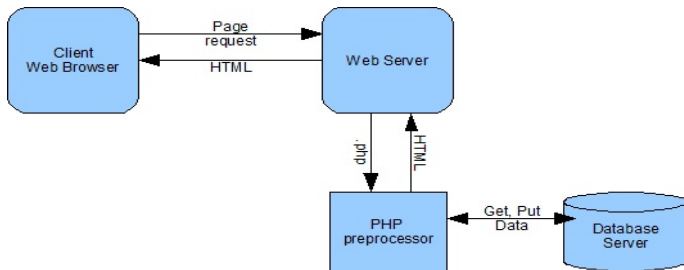


Figure: PHP process

PHP

Syntax

- Code position.

```
<?php  
// PHP code goes here  
?>
```

- Comments (`//`, `/* ... */`, `#`)
- Not case-sensitive.
- Composed of **Variables**, **Operators**, **Expressions**, and **Keywords**

PHP

Variables

- A variable starts with the \$ sign
- PHP automatically converts the variable to the correct data type.

```
<?php
```

```
    $x = 5;
```

```
    $y = 4;
```

```
    echo $x + $y;
```

```
?>
```

PHP

Data Types

- String
- Integer
- Float (floating point numbers - also called double)
- Boolean
- Array
- Object (class)
- NULL
- `var_dump($x);`

PHP

Operators

- Arithmetic operators (+, -, *, /, %)
- Assignment operators (=, +=, *=, /=)
- Comparison operators (==, ===, !=, <>, !==, >, <, >=, <=)
- Increment/Decrement operators (++\$x, \$x++, \$x--, --\$x)
- Logical operators (and, or, xor, &&, —, !)
- String operators (., =) concatenation
- Array operators (+, ==, ===, !=, !==, <>)

PHP

Conditions

- **if condition:**

```
if (condition) {  
    block of code to be executed if the condition is true  
}
```

- **else condition:**

```
if (condition) {  
    block of code to be executed if the condition is true  
} else {  
    block of code to be executed if the condition is false  
}
```

- **else if condition:**

- **switch condition:**

PHP

Switch

- Use the switch statement to select one of many blocks of code to be executed.

```
switch(expression) {  
    case n:  
        code block  
        break;  
    case n:  
        code block  
        break;  
    default:  
        default code block  
}
```

PHP

Loops

- You want the same block of code to run over and over again in a row.
- for loop
- foreach loop
- while loop
- do...while loop

PHP

While loop

- Syntax

```
while (condition is true) {  
    code to be executed;  
}
```

- Example

```
while($x <= 5) {  
    echo "The number is: $x <br >";  
    $x++;  
}
```

PHP

For loop

- Syntax

```
for (init counter; test counter; increment counter) {  
    code to be executed;  
}
```

- Example

```
for ($x = 0; $x <= 10; $x++) {  
    echo "The number is: $x <br >";  
}
```


PHP

Foreach loop

- Syntax

```
foreach ($array as $value) {  
    code to be executed;  
}
```

- Example

```
$colors = array("red", "green", "blue", "yellow");  
foreach ($colors as $value) {  
    echo "$value <br >";  
}
```

PHP

Functions

- Syntax

```
function functionName(arguments) {  
    code to be executed;  
}
```

- Call the function.

```
functionName();
```

- Example

```
function familyName($fname, $year) {  
    echo "$fname Refsnes. Born in $year <br >";  
}
```

- It can have return statement too.

*

PHP

PHP forms

- Create a simple HTML form.
- Connect it to the PHP.
- Show the content back (username and password)
- PHP uses (`$_GET` and `$_POST`) to collect form data.

Relavent links I



Javascript

<http://www.w3schools.com/js/default.asp>



PHP

<http://www.w3schools.com/php/default.asp>