

Web Programming

1

Web Concepts

- Web Programming
- Http Protocol
- Web Client
- Web Server
- Markup Language
 - Elements / Tags
 - Attributes / Property

Elements of HTML

- Heading
- Paragraph
- Anchor / Link
- Tables
- List
- Images / Pictures
- Media Object (Audio/Video)
- Canvas (Drawing)

Layout & Design Html Page

- Body Container
- Division Element
- Height / Width
- Alignment (Left/Right/Center)
- Horizontal / Vertical Alignment
- Position Property
- Visibility Property
- Float

HTML Form Element

- Input Text
- Text Area
- Radio / Checkbox
- Select
- Button and Submit Button
- Number

HTML 5

- New Semantic Element

- Header
- Footer
- Section
- Article

- New Form Element

- Datetime
- Number
- Email
- Url

- Web Storage

- Session Storage
- Local Storage

- Web SQL

- Drag & Drop

- Audio/Video

- Geo Location

- Canvas

- WebSocket

- Microdata

CSS

- CSS Syntax (Rule)
 - Element Selector
 - Class Selector
 - Id Selector
 - Descendent Selector
 - Child Selector
 - Pseudo Selector
- Inline Styling
- Internal Style sheet
- External Style Sheet

CSS Continue

➤ Border

- Style
- Width
- Color

➤ Margin

- Top
- Right
- Bottom
- Left

➤ Padding

- Top
- Right
- Bottom
- Left

➤ Color

- RGB
- Hex Code

➤ Background

- Background-color
- Background-image

➤ Texts

- Color
- Line-spacing
- Word spacing
- Line Height

➤ Fonts

- Family
- Size
- Weigth

CSS Continue

- Display

- Block

- Inline

- None

- Position

- Static

- Relative

- Fixed

- Absolute

- List

- Table

- Float

- Align

Bootstrap Classes

- GRID System Understand
- Table Classes
- Form Classes
- Button Classes
- Image Classes
- Contextual Classes
- Glyphicons
- Navs
- Label
- Jumbotron
- Alert
- List Group
- Panels
- Modal Component
- Carousel Component

jQuery DOM Manipulation

➤ Get / Set values of Html Element

- `val()`
- `html()`
- `text()`

➤ Add Css Class

- `addClass()`
- `removeClass()`
- `toggleClass()`

➤ Add Style

- `css()`

➤ Add New Element

- `append()`
- `prepend()`
- `after()`
- `before()`

➤ Remove Element

- `remove()`
- `empty()`

➤ Show/Hide

- `show()`
- `hide()`
- `toggle()`

jQuery Traversing

- **Ancestor**
 - `parent()`
- **Descendent**
 - `children();`
- **Sibling**
 - `prev()`
 - `next();`
 - `siblings();`
- **Filter**
 - `first()`
 - `last()`

PHP

- Hypertext Preprocessor.
- Server Side Programming Language.
- Generates dynamic web content (html, json, xml)
- Easily connects with database, and perform sql operation.
- Get/Set Session and Cookie at client.

- Install WAMP/LAMP/MAMP to run your php application.

PHP Syntax

- `<?php`
 `echo 'Hello World';`
- `?>`

PHP Variables

- Variables starts with \$ sign, followed by variable name.
- Variables are loosely coupled.
- Variables are case-sensitive.
- Scope
 - Global
 - Local

PHP Special Variable

- Special Variable
 - \$GLOBALS

 - \$_SERVER
 - \$_REQUEST
 - \$_GET
 - \$_POST
 - \$_SESSION
 - \$_COOKIE

 - \$_ENV
 - \$_FILES

Data Types

- ▶ String
 - ▶ `$name = "Hello World";`
- ▶ Integer
 - ▶ `$id = 1;`
- ▶ Float
 - ▶ `$price = 1.024;`
- ▶ Boolean
 - ▶ `$x = true;`
- ▶ Array
 - ▶ `$days = array();`
- ▶ Null
- ▶ Object
 - ▶ `$abc -> property;`

Strings in PHP

- **strlen**

- `echo strlen("Hello World");`

- **str_word_count**

- `echo str_word_count("Hello World");`

- **strrev**

- `echo strrev("Hello World");`

- **str_replace**

- `echo str_replace("world", "Hello World");`

- Ref-
http://www.w3schools.com/php/php_ref_string.asp

- **trim**

- Remove white space.

- **Stripslashes**

- Unquote a string quoted with slashes

- **htmlspecialchars**

- Convert predefined character to html entities.

Constans in PHP

- Constants are variable with fixed value.
 - Value cant be changes.
 - Constants are global.
 - Constant Name **Not** starts with **\$ Sign**.
-
- `define('PI', 3.14);` `// constant declaration.`
 - `echo PI;` `// constant usage.`

Operator in PHP

- Arithmetic Operator
- Assignment Operator
- Comparison Operator
 - `===`
 - `<>`
- Increment Operator
- Logical Operator
 - `and/or/not`
- String Operator
 - `$var1 . $var2;` `// concatenation`
 - `$var1 .= $var2;` `// concatenation`

For loop in php

➤ for (***init***; ***test***; ***increment***) {
 code to be executed;
}

➤ foreach (***\$array*** as ***\$value***) {
 code to be executed;
}

➤ foreach (***\$array*** as ***\$x => \$x_val***) {
 code to be executed;
}

Array in PHP

- Index Array

- `$days = array();`

- `$days[0] = 'Sunday';`

- `$days[1] = 'Monday';`

- Loop through Index array

- Associative Array

- `$days = array();`

- `$days['key1'] = 'Sunday';`

- `$days['key2'] = 'Monday';`

- Loop through Associative array.

Array in PHP

➤ Useful Methods

➤ `sort()`

- Sort array in ascending order.

➤ `rsort()`

- Sort array in descending order

➤ `asort()`

- Sort **associative** array in ascending order, according to **value**.

➤ `arsort()`

- Sort **associative** array in descending order, according to **value**.

➤ `ksort()`;

- Sort **associative** array in ascending order, according to **key**.

➤ `krsort()`;

- Sort **associative** array in descending order, according to **key**.

Function in PHP

- A block of statement, can be used repeatedly in a program.
- Used for logical operation.
- It does not get executed automatically.

```
function functionname(){  
    // logic  
}
```


Include and Require

- Insert the content of php file into another.
- Options
 - include - In case of failure gives only warning.
 - require – In case of failure produce error and stop scripts.

Exception in PHP

- Used to handle logical error.
- It changes the normal flow of script, and display proper message
- Keywords
 - try
 - catch
 - throw

```
try {  
    // block of statment  
} catch(Exception $e) {  
    echo 'Message: ' . $e->getMessage();  
}
```

File Handling

- Open a file
 - `$myfile = fopen("myfile.txt", "r");` // r, w, x, r+, w+, a+
- Close a file
 - `fclose($myfile);`
- Read file
 - `fread($myfile, filesize("myfile.txt"));`
- Write file
 - `$txt = "Hello World";`
`fwrite($myfile, $txt);`

Useful Methods

- **isset**
 - Determine if a variable is set and is not NULL
 - **unset**
 - Unset a given variable
 - **empty**
 - Determine whether a variable is empty
 - **var_dump**
 - Dumps information about a variable
-
- Ref : <http://php.net/manual/en/book.var.php>

Database Connectivity

■ Step 1

- `$conn = new mysqli($url, $username, $password, $dbname);`

■ Step 2

- `$conn->connect_error`

■ Step 3

- `$conn->query($sql) === TRUE`
- `$result = $conn->query($sql);`
- `if($result->num_rows > 0)`
- `while($row = $result->fetch_assoc())`

■ Step 4

- `$conn->close();`

Database Connectivity (Prepared)

■ Step 1

- `$conn = new mysqli($url, $username, $password, $dbname);`

■ Step 2

- `$conn->connect_error`

■ Step 3

- `$pstmt->prepare($sql)`

- `$pstmt->bind_param('ss', $first, $second);` // i-integer, s-string, d-double, b-blob

- `$first = 'abcd';`

- `$second = "efgh";`

- `$pstmt->execute();`

- `$pstmt->errno`

■ Step 4

- `$pstmt->close();`

- `$conn->close();`

Filter in PHP

- **Validation**
 - Check whether data is valid or not
- **Sanitization**
 - Remove illegal character from data
- **`filter_var(arg1, arg2)`**
 - `arg1` : variable we want to filter
 - `arg2` : type check.

Filter Constants

- FILTER_VALIDATE_INT
- FILTER_VALIDATE_FLOAT
- FILTER_VALIDATE_BOOLEAN
- FILTER_VALIDATE_EMAIL
- FILTER_VALIDATE_IP
- FILTER_VALIDATE_URL
- `if(!filter_var($avar, FILTER_VALIDATE_INT) === false)`
- FILTER_SANITIZE_STRING
- FILTER_SANITIZE_EMAIL
- FILTER_SANITIZE_URL

XML

- XML documents must have a **root** element
- XML elements must have a **closing** tag
- XML tags are **case sensitive**
- XML elements must be **properly nested**
- XML attribute values must be **quoted**

DTD

- ▶ Document Type Defination
- ▶ Well Formed & Valid.
- ▶ `<!DOCTYPE user`
 - [
 - `<!ELEMENT user (username, password, email, mobile?)>`
 - `<!ELEMENT username (#PCDATA)>`
 - `<!ELEMENT password (#PCDATA)>`
 - `<!ELEMENT email (#PCDATA)>`
 - `<!ELEMENT mobile (#PCDATA)>`
 - `]>`
- ▶ External DTD
 - ▶ `<!DOCTYPE root_element SYSTEM "DTD_location">`

DTD

```
> <?xml version="1.0" encoding="UTF-8"?>
  <!DOCTYPE user SYSTEM "user.dtd">
  <user>
    <username>test</username>
    <password>12345</password>
    <email>abcd@gmail.com</email>
    <mobile>022-23232323</mobile>
  </user>
```

Cookies in PHP

- A cookie is often used to identify a user.
- Used to store user preference.
- Server script sends a set of cookies to the browser.
- Browser stores this information on **local machine** for future use.
- Next time browser send the cookies with every request to the server.

- **setcookie**(*name, value, expire, path, domain, secure*);

Session in PHP

- ▶ A session is also used to identify a user.
- ▶ Identify multiple request from the same user.
- ▶ Unlike cookie, it store information at server.

- ▶ Session Methods :
 - ▶ **session_start()**
 - ▶ A session is started with the **session_start()** function.
 - ▶ **session_unset();**
 - ▶ Remove all session variable.
 - ▶ **session_destroy();**
 - ▶ Destroy the session.

AJAX

- Asynchronous JavaScript and XML.
- AJAX is a technique to update web page asynchronously, exchanging data with the server behind the scene.
- Ajax is combination of:
 - XMLHttpRequest Object
 - Javascript DOM
- *Steps*
 - `variable = new XMLHttpRequest();`
 - `xhttp.open("GET", "URL", true);`
 - `xhttp.send();`
- The `readyState` property holds the status of the XMLHttpRequest.
 - 0 – Initialized; 1- Connection Established; 2- Req received; 3-Process Request; 4-Done.
- The `onreadystatechange` event is triggered every time the `readyState` changes.

AJAX using jQuery

- `$.get(URL, data, callback);`
- `$.post(URL, data, callback);`
- `$.getJSON(url, data, callback);`

Thank You 😊

Mobile : 91-9323791976

Skype : santooindian

Gmail : santosh.ece06@gmail.com

Linkedin : <http://www.linkedin.com>

Facebook : <http://www.facebook.com>

Twitter : <http://www.twitter.com>