❖ AJAX in php

- AJAX stands for asynchronous JavaScript and xml.
- Using AJAX we can create fast and dynamic web pages.
- AJAX allows web pages to be updated asynchronously by exchanging small amounts of data with the server behind the scenes. This means that it is possible to update parts of a web page, without reloading the whole page.
- Classic web pages, (which do not use AJAX) must reload the entire page if the content should change.
- AJAX is based on internet standards, and uses a combination of:
    - XMLHttpRequest object (to exchange data asynchronously with a server)
    - JavaScript/DOM (to display/interact with the information)
    - CSS (to style the data)
    - XML (often used as the format for transferring data)
- AJAX applications are browser- and platform-independent.

- **The XMLHttpRequest Object**
    - All modern browsers support the XMLHttpRequest object.
    - The XMLHttpRequest object can be used to exchange data with a server behind the scenes. This means that it is possible to update parts of a web page, without reloading the whole page.
    - Create an XMLHttpRequest Object
    - All modern browsers (Chrome, Firefox, IE7+, Edge, Safari Opera) have a built-in XMLHttpRequest object.
    - Syntax for creating an XMLHttpRequest object:

```
variable = new XMLHttpRequest();
```

- **AJAX - Send a Request To a Server**
    - The XMLHttpRequest object is used to exchange data with a server.
    - Send a Request To a Server
    - To send a request to a server, we use the open() and send() methods of the XMLHttpRequest object:

```
xhttp.open("GET", "ajax_info.txt", true);
xhttp.send();
```

- **AJAX - Server Response**

    - The onreadystatechange Property
    - The readyState property holds the status of the XMLHttpRequest.
    - The onreadystatechange property defines a function to be executed when the readyState changes.
    - The status property and the statusText property holds the status of the XMLHttpRequest object.
    - The onreadystatechange function is called every time the readyState changes.
    - When readyState is 4 and status is 200, the response is ready

- Here are some example of AJAX with php

- **1. We can load other file's data into our main page without refreshing the page.**
- Example :

Include.php

```php
<?php

echo '<h1 style="color:powderblue;">Hello, this is the content from fil
e <br> thank you </h1>';

?>
```

Index.php

```html
<html lang="en">

<head>
    <title>ajax</title>
    <script type="text/javascript">
        function load(div, file) {
            var xmlhttp = new XMLHttpRequest();
            xmlhttp.onreadystatechange = function() {
                if (this.readyState == 4 && this.status == 200) {
                    document.getElementById(div).innerHTML = this.respo
nseText;
                }
            };
            xmlhttp.open("GET", file, true);
            xmlhttp.send();
        }
    </script>
</head>
```

```
<body>
    <input type="submit" onclick="load('content' , 'include.php' );"> <
br><br>
    <div id="content"><small>this content will be changed with file con
tent</small></div>
</body>
</html>
```

- In this example, include.php file's content will be included on index.php file without refreshing the page.

- **2. We can make suggestion application using AJAX with php**
- Example :

Include.php

```
<?php

if(isset($_GET['search_text'])){
    $search_text = $_GET['search_text'];
}

if(!empty($search_text)) {
    if (@$link = new mysqli('localhost','root','','training')) {
        $sql = "select name from names where name like '".mysqli_real_e
scape_string($link , $search_text)."%'";
        $result = mysqli_query($link , $sql);
        while($row = mysqli_fetch_array($result)) {
            echo $row['name'].'<br>';
        }
    }
}

?>
```

Suggestion_application.php

```
<!DOCTYPE html>

<html lang="en">
<head>
    <title>Auto Suggest App</title>
    <script type="text/javascript">
        function findMatch() {
```

```
            var xmlhttp = new XMLHttpRequest();
            xmlhttp.onreadystatechange = function() {
                if(this.status == 200 && this.readyState == 4) {
                    document.getElementById('results').innerHTML = this
.responseText;
                }
            }

            xmlhttp.open('GET' , 'include_suggest.php?search_text='+doc
ument.search.search_text.value , true);
            xmlhttp.send();
        }
    </script>
</head>
<body>
    <form id="search" name="search">
        Type a name : <br><br>
        <input type="text" name="search_text" onkeyup="findMatch();"> <
br> <br>

        <div id="results"></div>
    </form>
</body>
</html>
```

- **3. We can handle get and post request and perform database operations using ajax**
- Example :

Include.php

```
<?php

$link = new mysqli('localhost','root','','training');

if(isset($_POST['text'])) {
    $text = $_POST['text'];

    if(!empty($text)) {
        $sql = "INSERT into names (name) VALUES ('" . mysqli_real_escap
e_string($link , $text) . "')";
        if(mysqli_query($link, $sql)) {
            echo 'Name inserted Successfully';
        }
        else {
            echo 'sorry, operation failed';
```

```
        }
    }
    else {
        echo 'please fill some value';
    }
}

?>
```

Posting_data.php

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Posting data</title>
    <script type="text/javascript">

    function insert() {
        var xmlhttp = new XMLHttpRequest();

        xmlhttp.onreadystatechange = function() {
            if(this.readyState == 4 && xmlhttp.status == 200) {
                document.getElementById('message').innerHTML = xmlhttp.
responseText;
            }
        }

        var parameters = 'text='+document.getElementById('text').value;

        xmlhttp.open('POST', 'post_include.php', true);
        xmlhttp.setRequestHeader('Content-type' ,'application/x-www-
form-urlencoded');
        xmlhttp.send(parameters);
    }

    </script>
</head>
<body>

Insert :  <input type="text" id="text">  <input type="submit" value="In
sert" onclick="insert();"> <br><br>

<div id="message"></div>

</body>
</html>
```

❖ Namespace in php

- Namespaces are qualifiers that solve two different problems
  - They allow for better organization by grouping classes that work together to perform a task
  - They allow the same name to be used for more than one class
- Namespaces are declared at the beginning of a file using the namespace keyword
- A namespace can have constants , classes and functions only.

Example :

```php
<?php

namespace Product;
    const NUM = 45;
    class Laptop {
        function __construct()
        {
            echo '<h1>product namespace - Laptop class</h1>';
        }
    }
    function display() {
        echo '<h1>product namespace - display function</h1>';
    }

?>
```

- So if we want to create object of class , use any function or constant which is inside the namespace then it is fine to call them as we call regular function or constant.

Example :

```php
    echo NUM;
    display();
    $obj = new Laptop();
```

- But if we want to access them from outside the namespace then we have to use namespace\ attached to it.

Example :

```php
require 'product.php';
```

```
    echo Product\NUM;
    Product\display();
    $obj = new Product\Laptop();
```

- We can also give alias to namespace as well

  Example:

```
use Product as pro;
require 'product.php';

    echo pro\NUM;
    pro\display();
    $obj = new pro\Laptop();
```