

WAPH - Web Application Programming and Hacking

Instructor: Dr. Phu Phung

Student

Name : Sai Krishna Barupai

Email : barupasa@mail.uc.edu



Profile Pic :

Repository information for Lab-2

Repository URL : <https://github.com/barupasa/waph-barupasa/blob/main/labs/lab2>

In completing this Lab 2 Assignment - This lab focuses on Front-End Development, providing a comprehensive overview of fundamental

technologies such as HTML, JavaScript, Ajax, CSS, the JQuery library, and the integration of web APIs. In the first part of the lab, a HTML webpage is designed, incorporating basic tags and forms to create a foundation for user interaction. Next, JavaScript is introduced in four different ways: inline JS, JS within script tags, external JS files, and JS code retrieved from a remote repository. These approaches showcase the versatility of incorporating JavaScript into web pages. To enhance the visual appeal of the webpage, CSS is integrated using three methods: inline CSS, internal CSS, and external CSS. Each approach contributes to the overall design and aesthetics of the webpage. Furthermore, JQuery is implemented to facilitate AJAX (Asynchronous JavaScript and XML) calls to an ‘echo.php’ file, demonstrating how JQuery simplifies the process of making asynchronous requests. The lab also involves the integration of two web services using JQuery Ajax and the fetch method. One service generates a random joke, while the other allows users to guess a person’s age. Finally, the lab utilizes Pandoc to convert the README.md file into a PDF format, providing a convenient way to share documentation and information. This comprehensive approach to front-end development equips developers with a well-rounded understanding of essential technologies and their practical application in building interactive and visually appealing web pages.

Task 1: Basic HTML with forms, and JavaScript

*** For this assignment, I initiated by crafting a straightforward HTML file titled waph-barupasa.html. Through this task, I gained insights into integrating essential HTML tags like

,
,

, and others. Additionally, I incorporated a headshot image using the tag and formulated a form employing the

tag. This experience enhanced my comprehension of HTML structure and the incorporation of multimedia and interactive features within a webpage.

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8">
<title>WAPH- Sai Krishna Barupati</title>
</head>
<body>
<div >
```

```

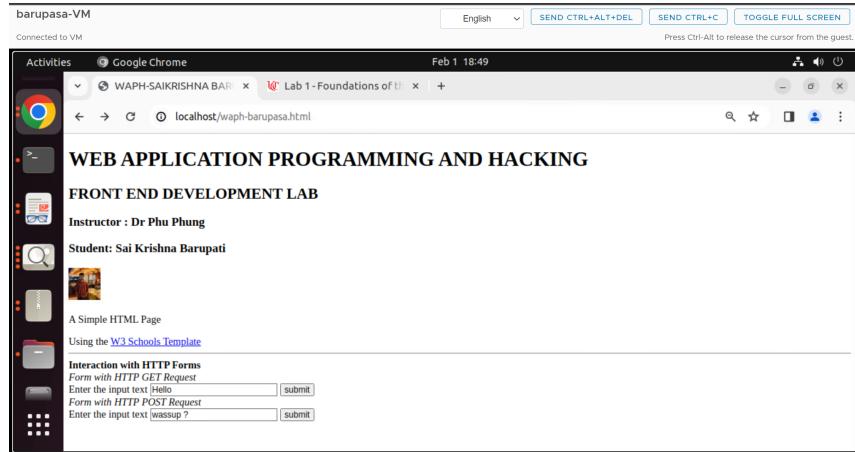
<div id="top">
    <h1>Web Application Programming and Hacking</h1>
    <h2>Front End Development Lab </h2>
    <h3>Instructor : Dr Phu Phung</h3>
</div>
<div>
    <div id="menubar">
        <h3> Student: Sai Krishna Barupati </h3>
        <div id="email" onclick="showOrHideEmail()">Display my Email</div>
            <script type="text/javascript" src="email.js"></script>

            
    </div>

    <div id="main">
        <p>A Simple HTML Page</p>Using the <a href="https://www.w3schools.com/htm...
        <hr>
        <b>Interaction with HTTP Forms</b>
        <div>
            <i>Form with HTTP GET Request</i>
            <form action="/echo.php" method="GET">
                <label for="data">Enter the input text</label>
                <input type="text" name="data" onkeyup="console.log('you have clicked a Key')">
                <input type="submit" value="submit">
            </form>
        </div>

        <div>
            <i>Form with HTTP POST Request</i>
            <form action="/echo.php" method="POST">
                <label for="data">Enter the input text</label>
                <input type="text" name="data" onkeyup="console.log('you have clicked a Key')">
                <input type="submit" value="submit">
            </form>
        </div>
    </div>
</body>
</html>

```



*** Simple JavaScript

This task offers a basic overview of JavaScript syntax and different approaches to incorporating JavaScript code into an HTML file. Inline JavaScript code is utilized to exhibit the presentation of the current date and time upon clicking, as well as to record the click event in the console.

```
<div>
<hr>
<b>Experiments with Javascript</b> <br>
<i>Inlined JavaScript</i>
<div id="date" onclick="document.getElementById('date').innerHTML= Date()"> </div>
```

Date&Time: DigitalClock: Analog Clock:

Code snippet:

```
<div id="digit-clock"></div>
<canvas id="analog-clock" width="150" height="150" style="background-color:#999"></canvas>
<script src="https://waph-uc.github.io/clock.js"></script>
<script type="text/javascript">
    function displaytime()
    {
        document.getElementById('digit-clock').innerHTML = "Current time: " + new Date()
    }
    setInterval(displaytime,500);
</script>
</div>
```

Email: Code snippet
`var shown =false;`

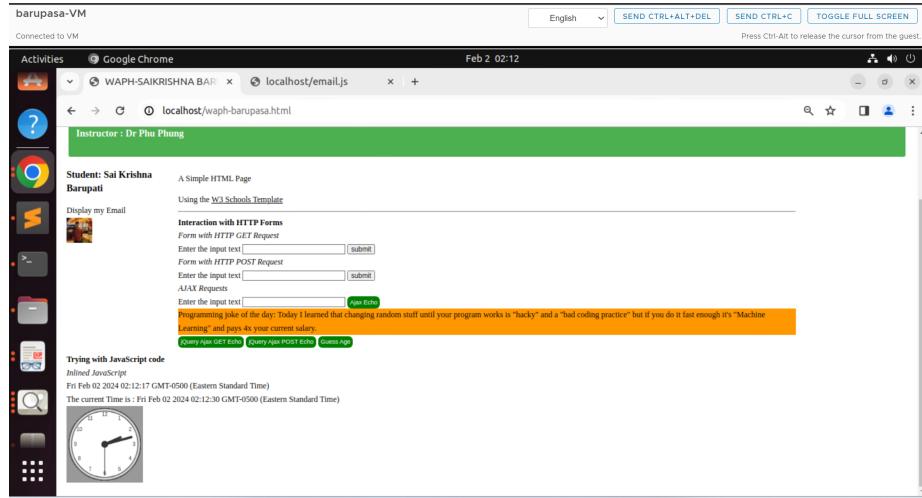
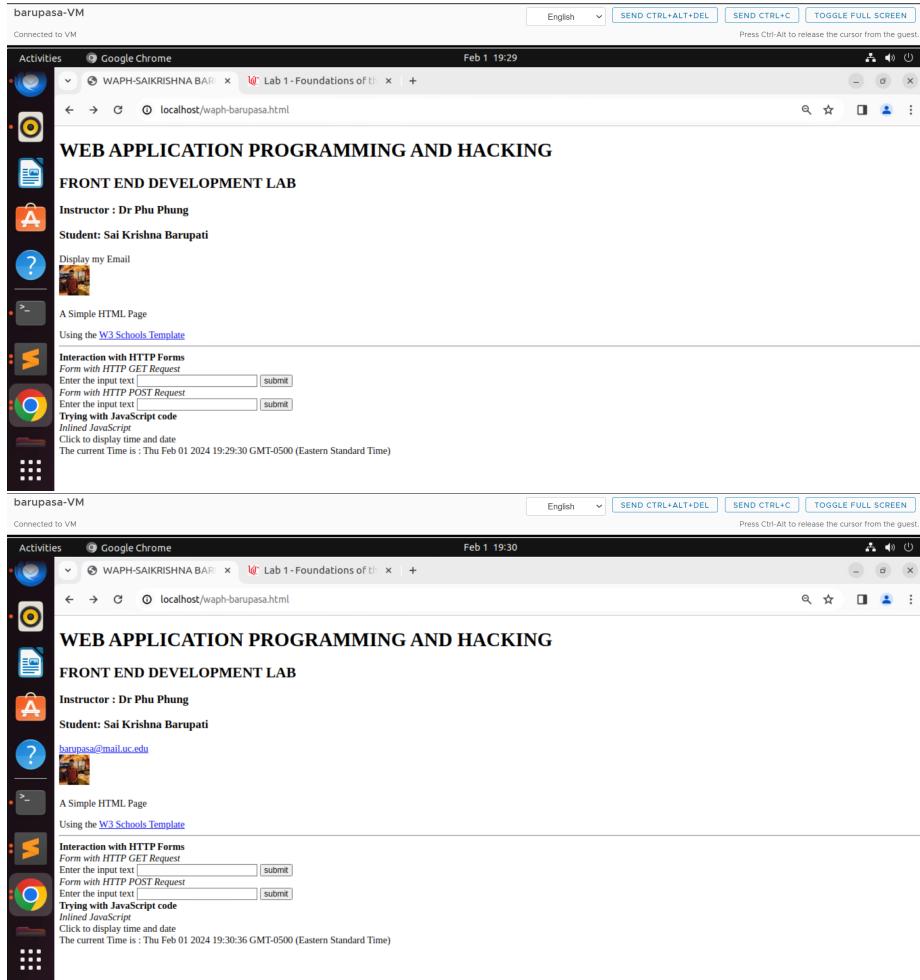


Figure 1: Task1.b

```

function showhideEmail()
{
    if (shown)
    {
        document.getElementById('email').innerHTML = "show my email";
        shown = false;
    }
    else
    {
        var myemail = "<a href= 'mailto:barupasa"+ "@" + "mail.uc.edu' > barupasa" + "@" +
        document.getElementById('email').innerHTML = myemail;
        shown = true;
    }
}

```



Task 2: Ajax, CSS, jQuery, and Web API integration

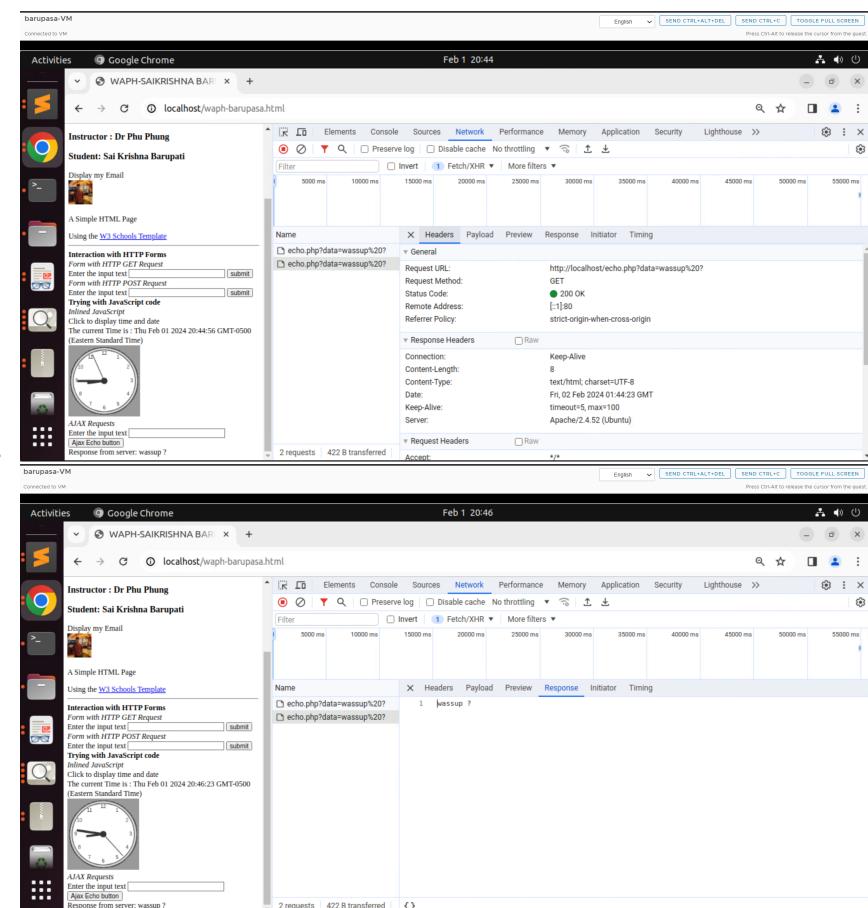
- Ajax The HTML code is crafted to capture user input, initiating a GET request to echo.php through AJAX. The ensuing response is showcased within a designated div. Given the nature of the GET call, the input is transmitted as a path variable in the URL to the webserver. The analysis of the Ajax call's response is conducted in the inspect view, revealing a GET request method, a status code of 200 OK, and the inclusion of input data within the URL.

```
<div>
    <i> AJAX Requests </i><br>
    <label for="data">Enter the input text</label>
    <input type="text" name="data" id="data">
```

```

<input class ="button round" type="submit" value="Ajax Echo" onclick="getEcho()
<input class="button round" type="submit" value="Ajay Jquery GET" onclick="getJQueryGet()
<input class="button round" type="submit" value="Ajay Jquery POST" onclick="getJQueryPost()
<input class ="button round" type="submit" value="Guess Age" onclick="guessAge()
<div ID="response"></div>

```



b. CSS

I have applied Inline ,internal and external CSS to style the html. CSS:

```

<h1 style="color:blue;">WEB APPLICATION PROGRAMMING AND HACKING</h1>

<link rel="stylesheet" type="text/css" href="https://waph-uc.github.io/style1.css">
<style>
.button{
    background-color:green;
    border:none;
}

```

```

        color:white;
        padding:5px;
        text-align:center;
        text-decoration:none;
        display:inline-block;
        font-size:12px;
        margin:4px2px;
        cursor:pointer;
    }
    .round{
        border-radius:8px;
    }
    #response{
        background-color:#ff9800;
    }
</style>


- ! [Task2.a.Internal] (images/Task2.a.Internal.png)
- ! [Task2.a.Inline] (images/Task2.a.Inline.png)
- ! [Task2.a.External] (images/Task2.a.External.png)

```

c. jQuery

I integrated the jQuery library into my webpage and developed HTML and JavaScript code to send Ajax GET requests to echo.php. The use of jQuery significantly simplified DOM manipulation and communication with the server. By incorporating buttons to initiate Ajax requests and exhibit response content upon click events, I further honed my skills in leveraging jQuery for effective front-end development.

```

function getJqueryAjax(){
    var input=$("#data").val();
    if(input.length==0)
        return;
    $.get("echo.php?data="+input,function(result){
        printResult(result);
    });
    $("#data").val("");
}
function getJqueryAjaxPost(){
    var input=$("#data").val();
    if(input.length==0)
        return;
    $.post("echo.php",{data:input},function(result){
        printResult(result);
    });
    $("#data").val("");
}

```

JQUery:

The screenshots illustrate the use of jQuery to interact with a PHP echo page. The first screenshot shows a GET request to echo.php?data=Hello, and the second shows a POST request to echo.php. Both requests return a 200 OK status with a response time of approximately 100ms.

Request	Method	URL	Status	Response Time (ms)
echo.php?data=Hello	GET	http://localhost/echo.php?data=Hello	200 OK	~100
echo.php	POST	http://localhost/echo.php	200 OK	~100

d. Web API integration

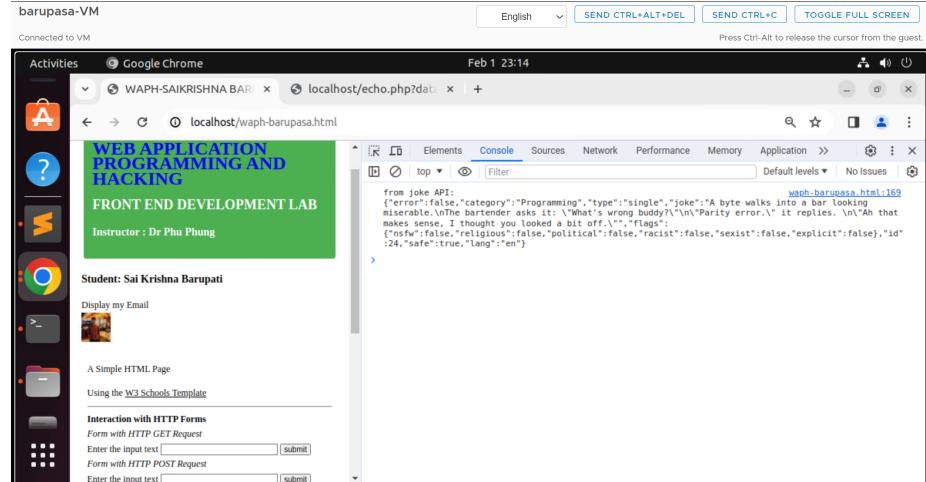
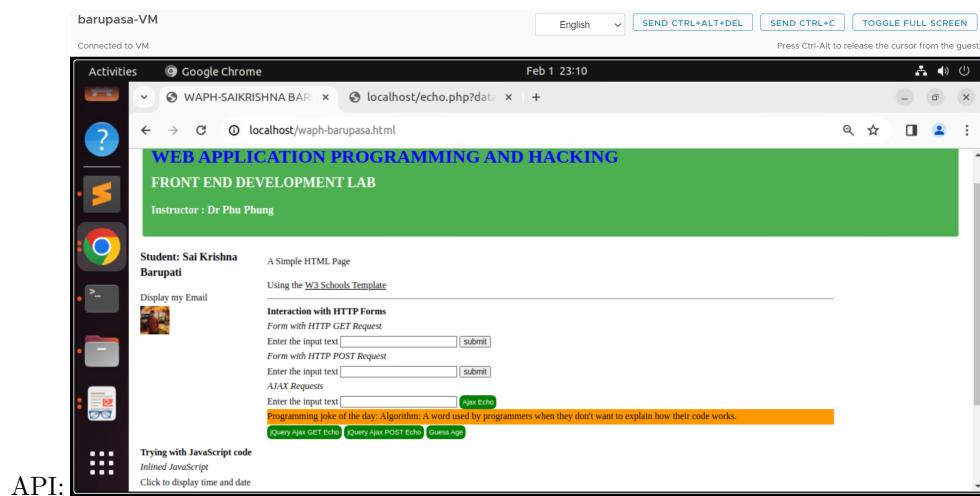
By employing Ajax on <https://v2.jokeapi.dev/joke/Programming?type=single>, I crafted JavaScript code utilizing jQuery Ajax to initiate a request and manage the response, presenting a random joke upon page loading. Examining the network through browser inspection allowed me to scrutinize both the request and response, enriching my comprehension of incorporating external APIs.

Similarly, leveraging the fetch API on <https://api.agify.io/?name=input> entailed the creation of HTML and JavaScript code to invoke the API using user input and exhibit the resulting response. Analyzing the network connections offered valuable insights into the request and response mechanism of the fetch API.

```

function printResult(result){
    $("#response").html("Response From Server: " + result);
}
$.get("https://v2.jokeapi.dev/joke/Programming?type=single",function(result){
    $("#response").html("Programming joke of the day: " +result.joke);
});
async function guessAge(name){
    const response= await fetch("https://api.agify.io/?name="+name);
    const result= await response.json();
    $("#response").html("Hello "+name+" ,your age should be "+result.age);
}

```



barupasa-VM

Connected to VM

Activities Google Chrome Feb 1 23:11

localhost/echo.php?data=

WEB APPLICATION PROGRAMMING AND HACKING
FRONT END DEVELOPMENT LAB
Instructor : Dr Phu Phung

Student: Sai Krishna Barupati

A Simple HTML Page
Using the W3 Schools Template

Interaction with HTTP Forms

Form with HTTP GET Request
Enter the input text submit

Form with HTTP POST Request
Enter the input text submit

AJAX Requests
Enter the input text Ajax echo
Hello Sai Krishna,your age should be undefined
Query Ajax GET Echo Query Ajax POST Echo Guess Age

Trying with JavaScript code
Inlined JavaScript
Click to display time and date

barupasa-VM

Connected to VM

Activities Google Chrome Feb 1 23:14

localhost/waph-barupasa.html

WEB APPLICATION PROGRAMMING AND HACKING
FRONT END DEVELOPMENT LAB
Instructor : Dr Phu Phung

Student: Sai Krishna Barupati

A Simple HTML Page
Using the W3 Schools Template

Interaction with HTTP Forms

Form with HTTP GET Request
Enter the input text submit

Form with HTTP POST Request
Enter the input text submit

Console

```
from joke API:
{"error":false,"category":"Programming","type":"single","joke":"A byte walks into a bar looking
at the menu. The bartender says: 'What can I get you?' The byte replies: '\n'Ah that
makes sense, I thought you looked a bit off.\n','flags":{},"nsfw":false,"religious":false,"political":false,"racist":false,"sexist":false,"explicit":false}, "id":24,"safe":true,"lang":"en"}
```

barupasa-VM

Connected to VM

Activities Google Chrome Feb 2 17:00

localhost/waph-barupasa.html

WEB APPLICATION PROGRAMMING AND HACKING
FRONT END DEVELOPMENT LAB
Instructor : Dr Phu Phung

Student: Sai Krishna Barupati

A Simple HTML Page
Using the W3 Schools Template

Interaction with HTTP Forms
Form with HTTP GET Request
Enter the input text submit

Form with HTTP POST Request
Enter the input text submit

AJAX Requests
Enter the input text

Programming joke of the day: "Can I tell you a TCP joke?" "Please tell me a TCP joke." "OK, I'll tell you a TCP joke."

Copy Ajax GET Click Copy Ajax POST Click Clear

Trying with JavaScript code
Inlined JavaScript
Click to display time and date
The current Time is : Fri Feb 02 2024 17:00:57 GMT-0500 (Eastern Standard Time)

barupasa-VM

Connected to VM

Activities Google Chrome Feb 2 17:02

localhost/waph-barupasa.html

WEB APPLICATION PROGRAMMING AND HACKING
FRONT END DEVELOPMENT LAB
Instructor : Dr Phu Phung

Student: Sai Krishna Barupati

A Simple HTML Page
Using the W3 Schools Template

Interaction with HTTP Forms
Form with HTTP GET Request
Enter the input text submit

Form with HTTP POST Request
Enter the input text submit

AJAX Requests
Enter the input text

Programming joke of the day: "Can I tell you a TCP joke?" "Please tell me a TCP joke." "OK, I'll tell you a TCP joke."

Copy Ajax GET Click Copy Ajax POST Click Clear

Trying with JavaScript code
Inlined JavaScript
Click to display time and date
The current Time is : Fri Feb 02 2024 17:01:59 GMT-0500 (Eastern Standard Time)

The Network tab shows the request for "Programming?type=single".

Request URL: https://v2.jokeapi.dev/joke/Programming?type=single
Request Method: GET
Status Code: 200 OK
Remote Address: 104.21.76.251:443
Referrer Policy: strict-origin-when-cross-origin

Response Headers:

- Access-Control-Allow-Origin: *
- Access-Control-Allow-Methods: GET, POST, HEAD, OPTIONS, PUT
- Access-Control-Allow-Headers: Content-Type, Authorization, Accept
- Access-Control-Request-Method: GET
- Access-Control-Request-Headers: Content-Type

Response Payload (JSON):

```

1  {
2     "error": false,
3     "category": "Programming",
4     "type": "single",
5     "joke": "Can I tell you a TCP joke?\nPlease tell me a TCP joke.",
6     "flags": {
7         "nsfw": false,
8         "religious": false,
9         "politic": false,
10        "racist": false,
11        "explicit": false,
12        "nsfw": false
13    },
14    "id": 57,
15    "safe": true,
16    "lang": "en"
17 }

```

barupasa-VM

Connected to VM

Activities Google Chrome Feb 2 17:03

localhost/waph-barupasa.html

WEB APPLICATION PROGRAMMING AND HACKING
FRONT END DEVELOPMENT LAB
Instructor : Dr Phu Phung

Student: Sai Krishna Barupati
Using the W3 Schools Template

Interaction with HTTP Forms

Form with HTTP GET Request
Enter the input text:

Form with HTTP POST Request
Enter the input text:

AJAX Requests
Enter the input text:

Hello Sai Krishna, your age should be undefined

Trying with JavaScript code

Inlined JavaScript
Click to display time and date
The current Time is : Fri Feb 02 2024 17:03:26 GMT-0500 (Eastern Standard Time)

barupasa-VM

Connected to VM

Activities Google Chrome Feb 2 17:04

localhost/waph-barupasa.html

WEB APPLICATION PROGRAMMING AND HACKING
FRONT END DEVELOPMENT LAB
Instructor : Dr Phu Phung

Student: Sai Krishna Barupati
Using the W3 Schools Template

Interaction with HTTP Forms

Form with HTTP GET Request
Enter the input text:

Form with HTTP POST Request
Enter the input text:

AJAX Requests
Enter the input text:

Hello Sai Krishna, your age should be undefined

Trying with JavaScript code

Inlined JavaScript
Click to display time and date
The current Time is : Fri Feb 02 2024 17:04:09 GMT-0500 (Eastern Standard Time)

The Network tab in the developer tools shows the following requests:

Name	Headers	Payload	Preview	Response	Initiator	Timing
Programming?type=single						
?name=Sai%20Krishna						

Request URL: https://api.agify.io/?name=Sai%20Krishna
Request Method: GET
Status Code: 429 Too Many Requests
Remote Address: 165.227.126.8:443
Referer Policy: strict-origin-when-cross-origin

Response Headers

Header	Value
Access-Control-Allow	true
Credentials	*
Access-Control-Allow-Origin	*
Access-Control-Expose-Headers	x-rate-limit-limit,x-rate-limit-remaining,x-rate-limit-reset
Cache-Control	max-age=0, private, must-revalidate
Content-Length	33

Response

```
[{"error": "Request limit reached"}]
```