Experiment No. 06

Title: Event handling using JavaScript to explore web browser environment.

Batch:A2 **Roll No.:16010421063** **Experiment No.:6**

Aim: Event handling using JavaScript to explore web browser environments.

Resources needed: Notepad++, Web Browser
Theory:

An HTML event can be something the browser does, or something a user

does. Here are some examples of HTML events:

An HTML web page has finished
loading An HTML input field was
changed
An HTML button was clicked
Often, when events happen, you may want to do

something. JavaScript lets you execute code when events

are detected.

HTML allows event handler attributes, with JavaScript code, to be added to HTML elements.

What can JavaScript Do?
Event handlers can be used to handle, and verify, user input, user actions, and browser actions:

- Things that should be done every time a page loads
- Things that should be done when the page is closed
- Action that should be performed when a user clicks a button
- Content that should be verified when a user inputs data

Many different methods can be used to let JavaScript work with events:

- HTML event attributes can execute JavaScript code directly
- HTML event attributes can call JavaScript functions
- You can assign your own event handler functions to HTML elements
- You can prevent events from being sent or being handled

Syntax:

```
<element event='some JavaScript'>
<element event="some JavaScript">
```

Example

In the following example, an onclick attribute (with code), is added to a <button> element:

```
<!DOCTYPE html>
<html>
<body>
<button onclick="document.getElementById('demo').innerHTML=Date()">The
time is?</button>

<p id="demo"></p>

</body>
</html>
```

JavaScript HTML DOM Events

HTML DOM allows JavaScript to react to HTML events:

A JavaScript can be executed when an event occurs, like when a user clicks on an HTML element. To execute code when a user clicks on an element, add JavaScript code to an HTML event attribute:

onclick=JavaScript

EXAMPLE

```
<!DOCTYPE html>
<html>
<body>

<h1 onclick="this.innerHTML='Ooops!'">Click on this text!</h1>

</body>
</html>
```

Activity:

Apply following JS events on your web pages Input Events

- Onblur
- onreset

Mouse Events

- Onmouseover
- Onmousedown

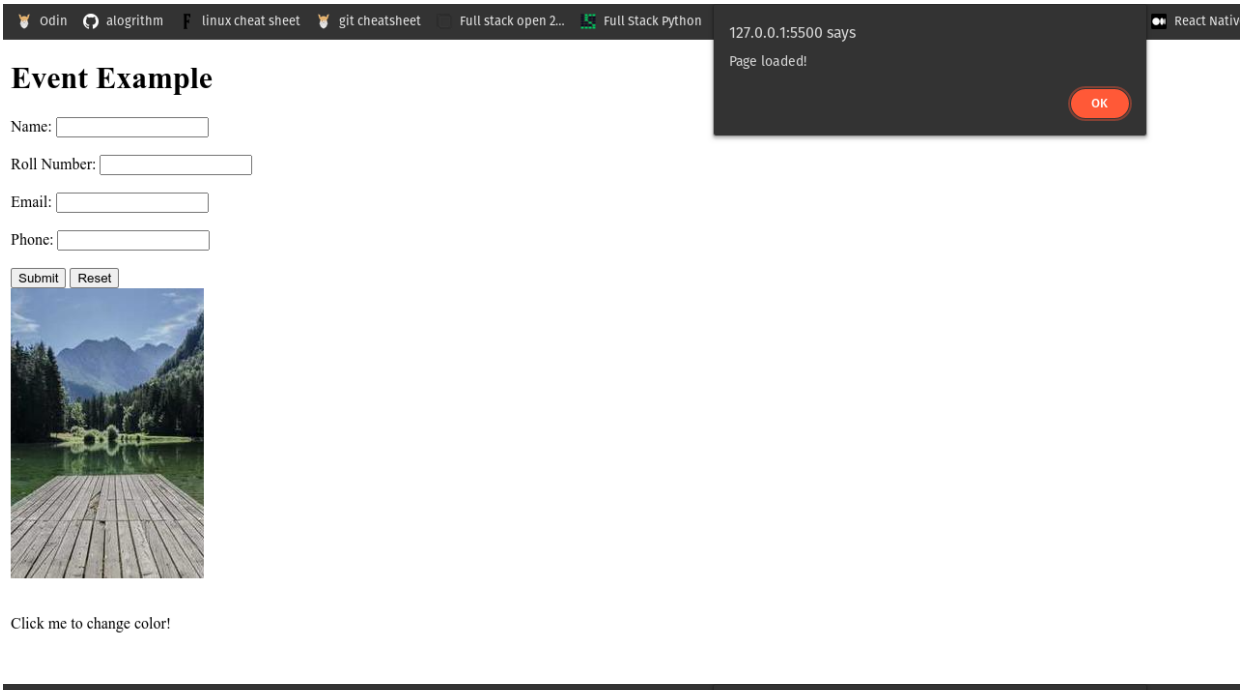
Click Events

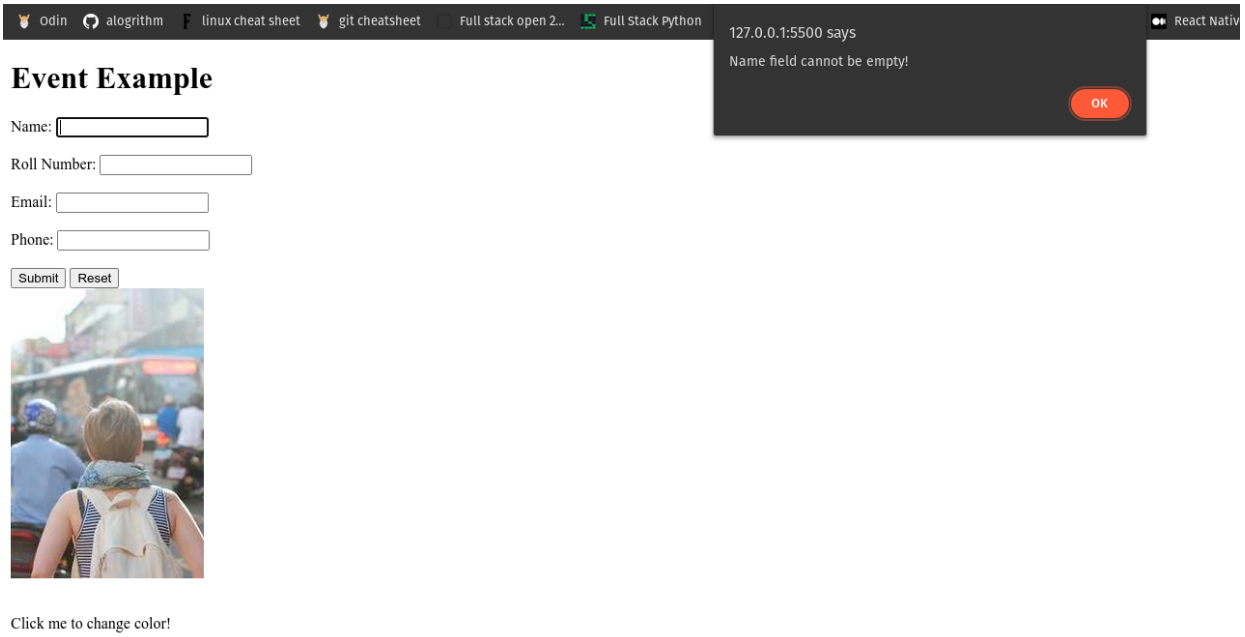
- Onclick
- Ondbclick

Apply following DOM events to your webpages

- Onload
- Onchange
- onmouseover

Results: (Program printout with output)





127.0.0.1:5500 says
Form reset!
OK

Event Example

Name: Arya

Roll Number:

Email:

Phone:

Submit Reset

Click me to change color!

Odin alogrithm linux cheat sheet git cheatsheet Full stack open 2... Full Stack Python React Nativ

127.0.0.1:5500 says
Mouse over image!
OK

Event Example

Name:

Roll Number:

Email:

Phone:

Submit Reset

Click me to change color!

Event Example

Name:

Roll Number:

Email:

Phone:

Submit Reset

Click me to change color!

Odin alogrithm linux cheat sheet git cheatsheet Full stack open 2... Full Stack Python UI Design Daily |... USACO Bronze To... Submissions - Co... React Nativ

Event Example

Name:

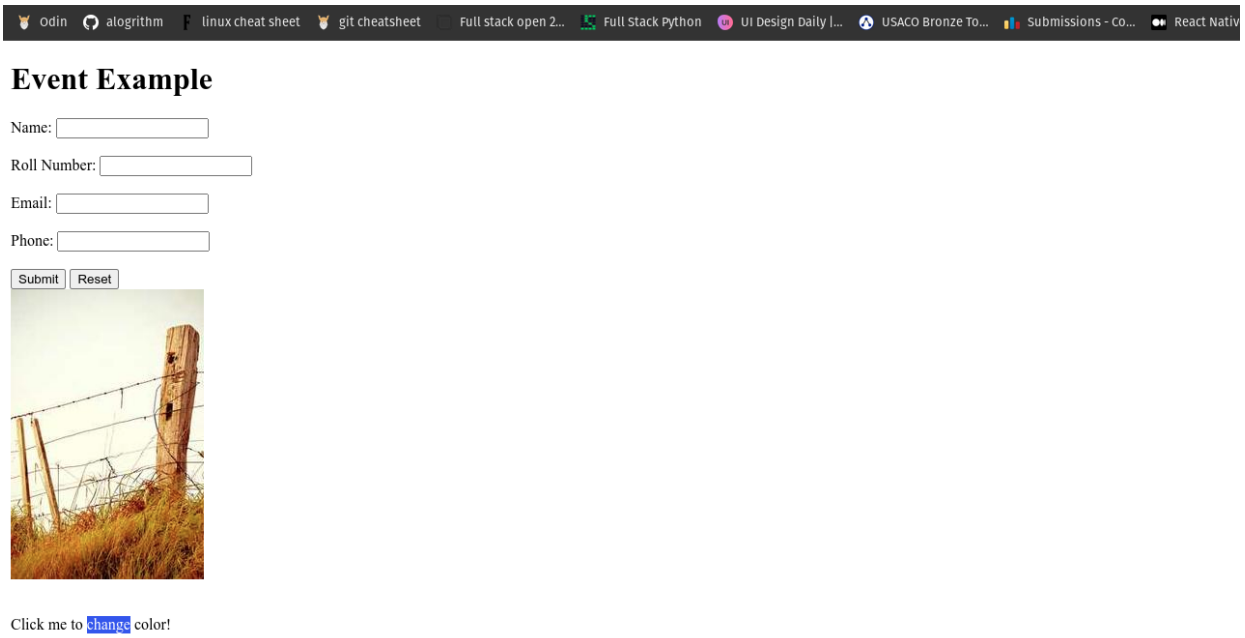
Roll Number:

Email:

Phone:

Submit Reset

Click me to change color!



Questions:

Q1) What are the different types of load events

There are two main types of load events in
web development:

1. DOMContentLoaded: triggered when the HTML is parsed and the DOM is constructed, allowing JavaScript to start interacting with the page.
2. Load: triggered when all resources of the web page (including images, scripts, and stylesheets) have finished loading.

Q2) Explain Onkeypress , onkeyup events

Both onkeypress and onkeyup are JavaScript events that are triggered when a key is pressed and released respectively.

The onkeypress event is fired when a key is pressed down and it provides the Unicode value of the pressed key. This event is useful for capturing user input in real-time, like in search fields or forms.

The onkeyup event is fired when a key is released after being pressed down. It provides the same information as onkeypress, but it's triggered after the user releases the key, making it useful for performing actions based on a specific key combination (like Ctrl+S to save).

In summary, onkeypress is triggered when a key is pressed down, and onkeyup is triggered when a key is released after being pressed down.

Outcomes:

CO 3 Apply JavaScript and JSON for Web Application development

Conclusion: (Conclusion to be based on the outcomes

achieved) Grade: AA / AB / BB / BC / CC / CD /DD

Implemented various methods on javascript DOM

Signature of faculty in-charge with date

References:

Books/ Journals/ Websites:

- "Web technologies: Black Book", Dreamtech Publications
- http://www.w3schools.com