

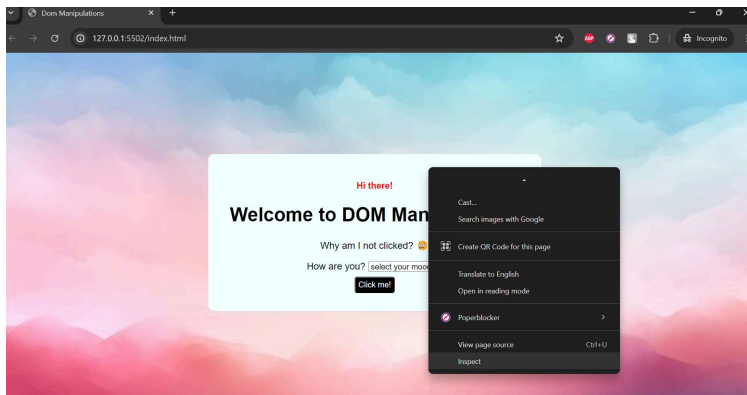
# Using Chrome Dev Tools - POC

Using Chrome DevTools for JavaScript development can significantly enhance your **debugging** and **development workflow**

## Opening Chrome DevTools

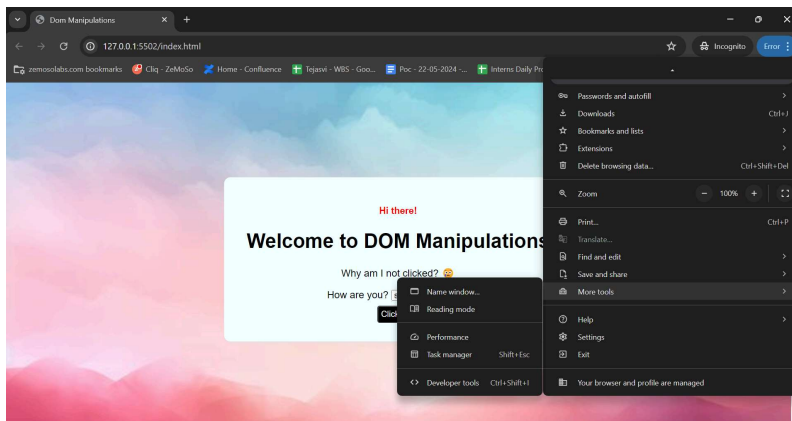
### Method-1

-Left mouse click on page you want to inspect and go to inspect



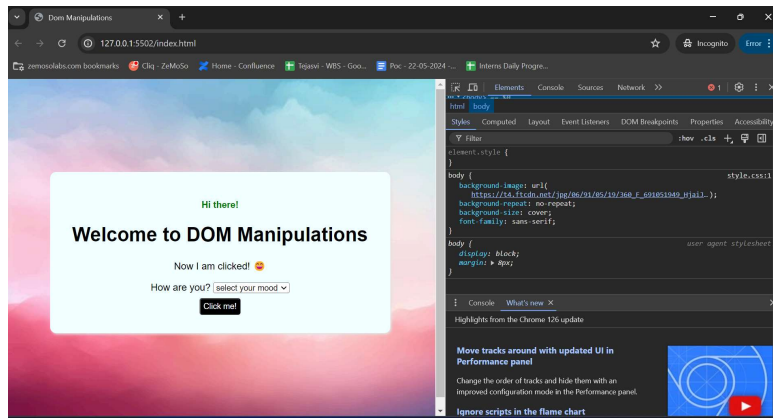
### Method-2

-Click on 3 dots on chrome tab there go to more tools where you can go to developer tools

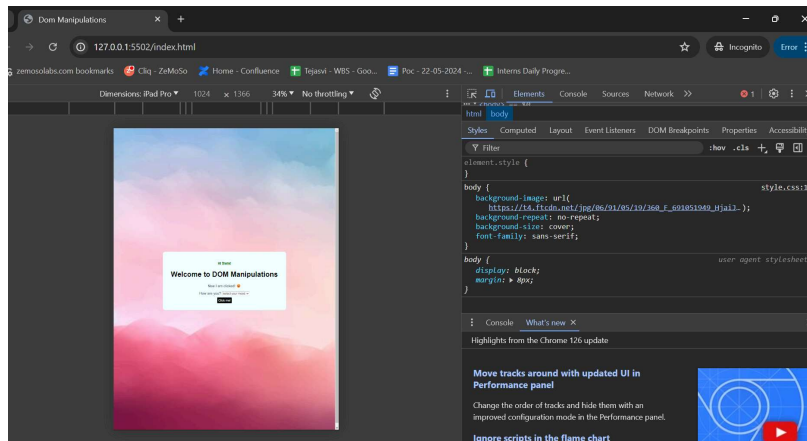


## Method-3

-Or shortcut **ctrl+shift+j**

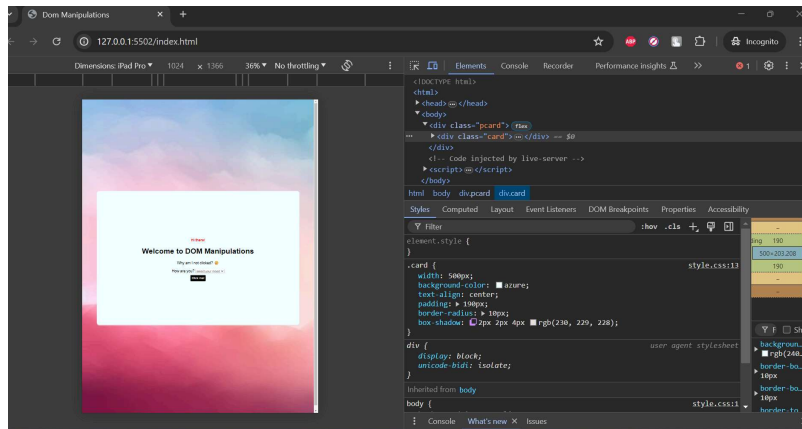


We can check the responsiveness by inspecting the page , we can inspect dimensions with predefined device dimensions or we can make adjustments with our custom dimensions

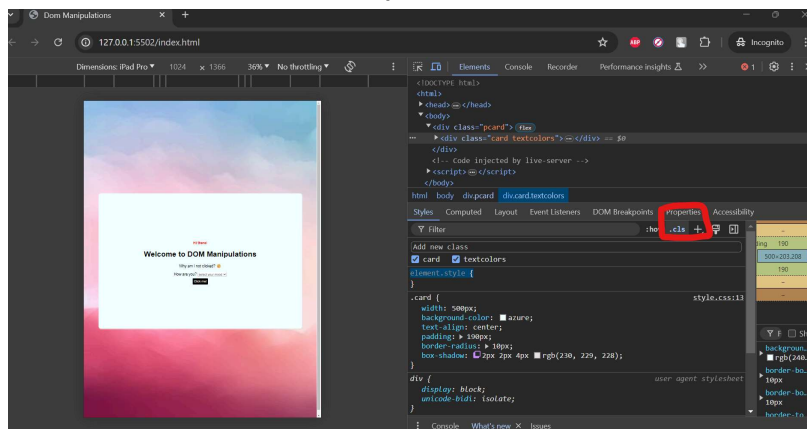


## Elements Tab

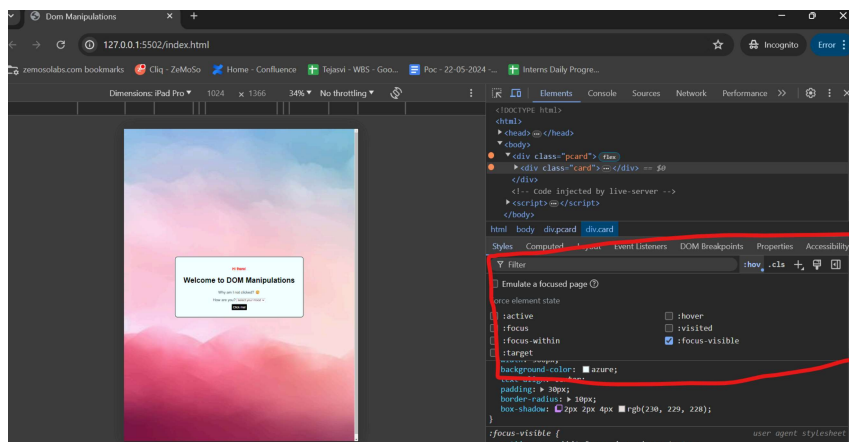
- The Elements tab allows you to inspect and modify the DOM and CSS.
- **Live Editing**: Directly edit HTML and CSS, and see changes reflected in real-time.
- **Event Listeners**: View and manage event listeners attached to DOM elements.



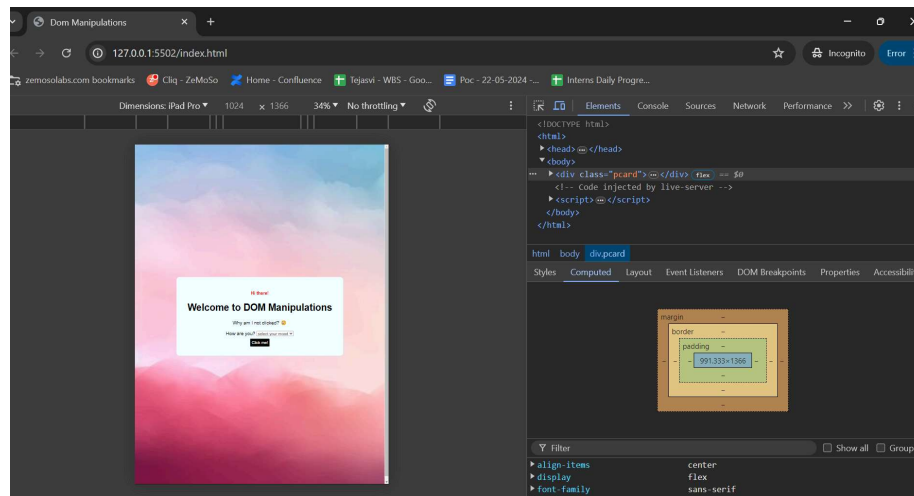
We can add a new class in styles.



Can add event listeners



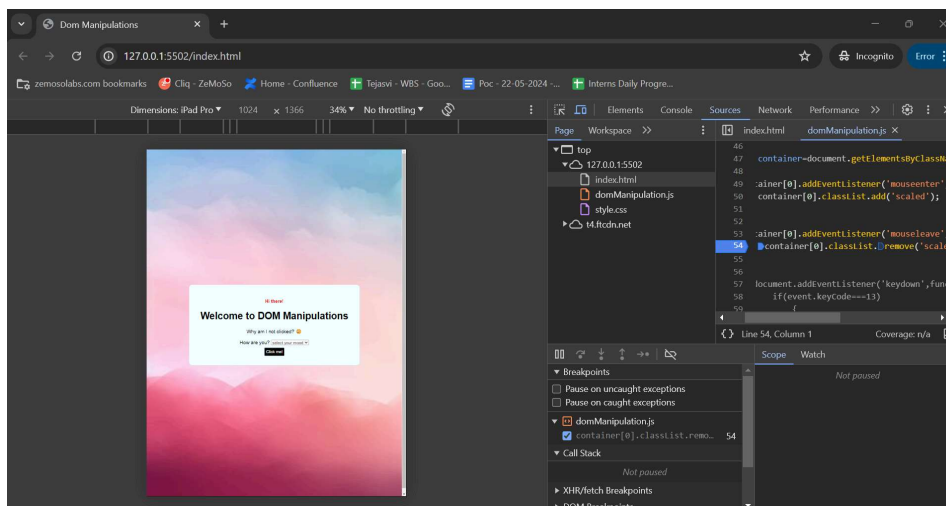
We can also inspect css box model



## Sources Tab

The Sources tab allows you to view and debug JavaScript code

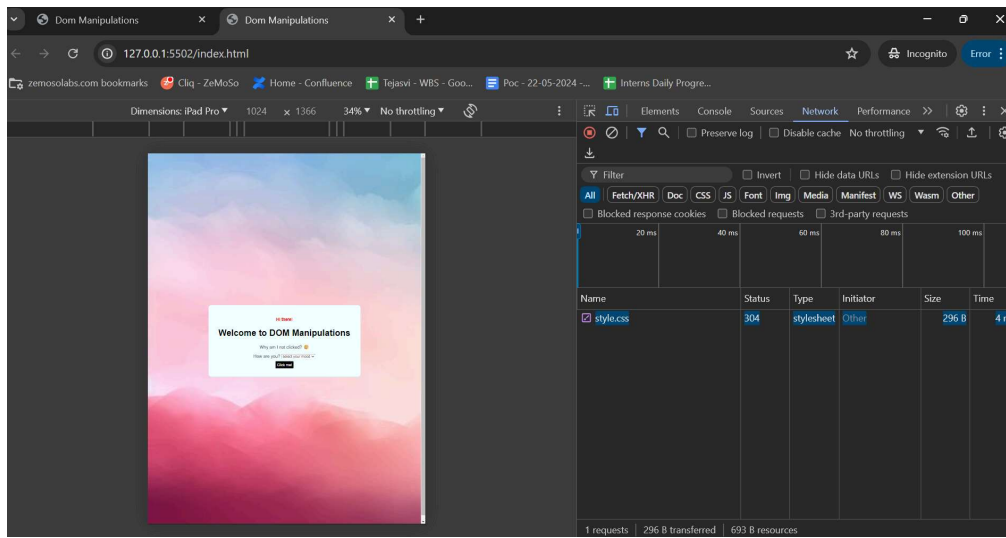
- **Breakpoints:** Click on the line number in the code to set a breakpoint. The code execution will pause at this line.
- **Step Through Code:** Use the controls to step over, step into, or step out of functions while debugging.
- **Watch Expressions:** Add variables or expressions to the "Watch" section to monitor their values during code execution.



## Network Tab:

The Network tab helps you monitor and debug network requests.

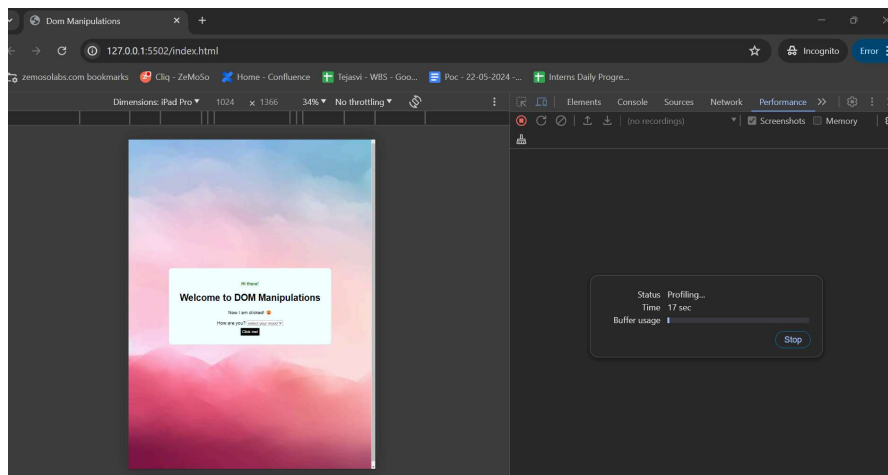
- **Inspect Requests:** View details of network requests such as headers, payload, response, and timing.
- **Filter Requests:** Use filters to focus on specific types of requests (e.g., XHR, JS, CSS).



## Performance Tab:

The Performance tab is used to record and analyze runtime performance.

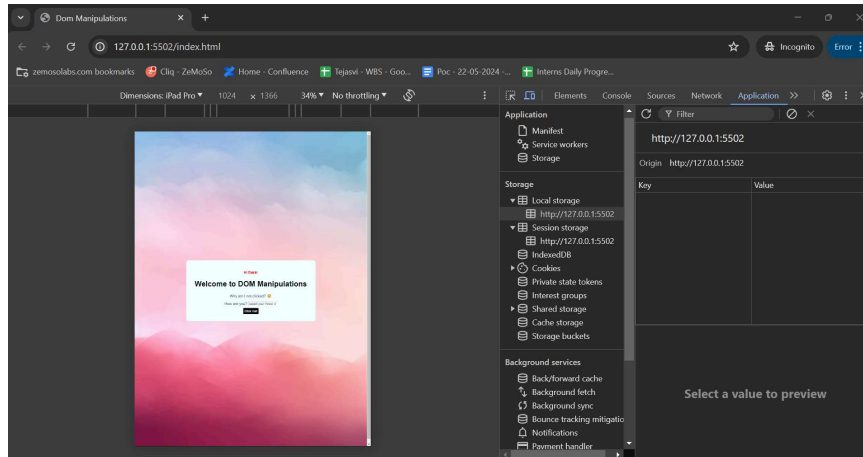
**Profile Your Code:** Record a session to see where your application might be running slowly and optimize accordingly.



## Application Tab

The Application tab gives access to storage, service workers, and other application-related resources.

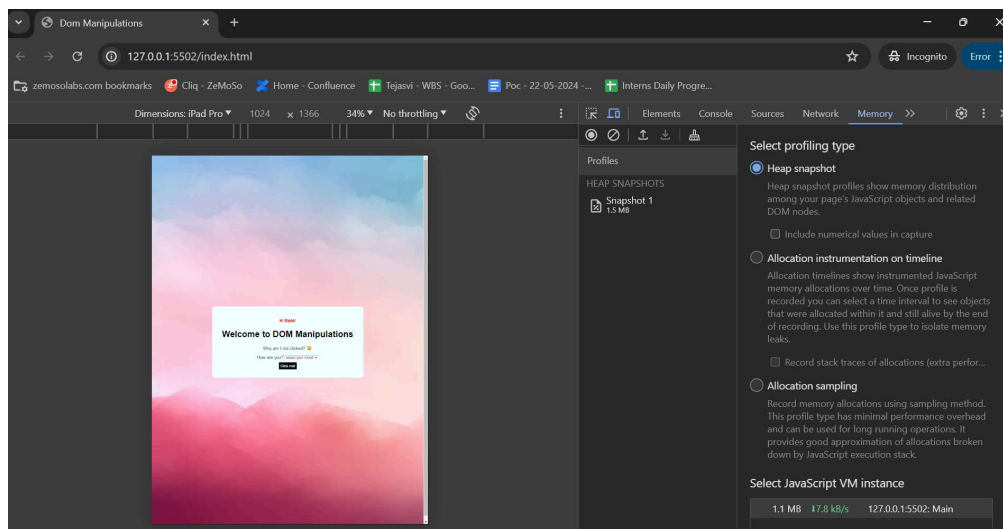
**Local Storage and Cookies:** View, edit, and delete local storage, session storage, and cookies.



## Memory Tab:

The Memory tab helps you diagnose memory leaks and optimize memory usage.

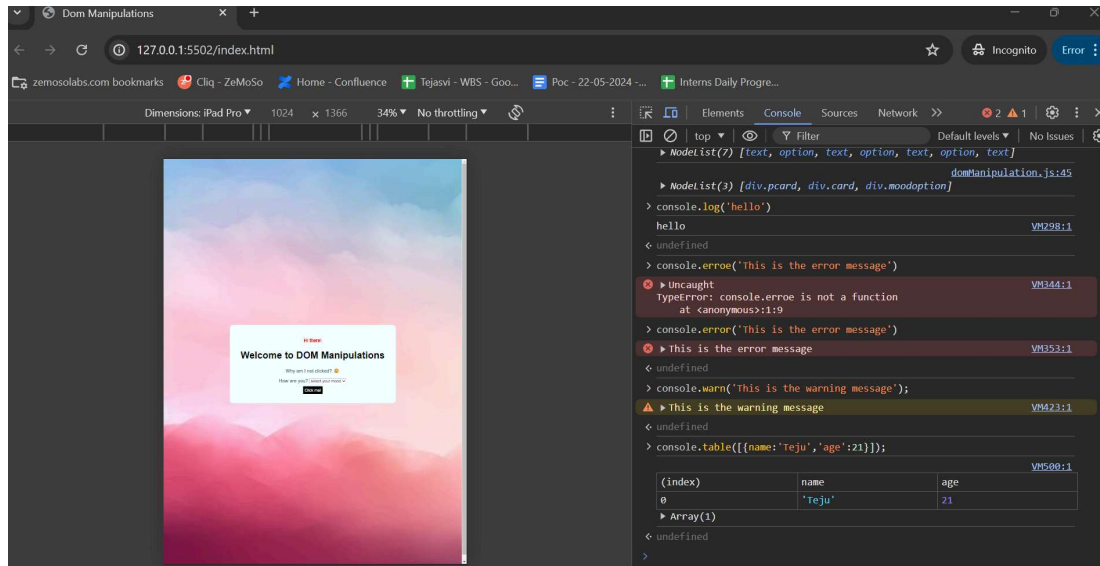
- **Heap Snapshot:** Take snapshots to inspect the memory heap and track memory usage over time.
- **Allocation Timeline:** Record memory allocations to identify and troubleshoot memory leaks.



## Console Tab:

The Console tab allows you to run JavaScript code, log output, and interact with the webpage's JavaScript context.

Logging: Use **console.log()**, **console.error()**, **console.warn()**, **console.table()**, etc., to output messages.



```
< undefined
> console.group('User Details');
  console.log('Name: Alice');
  console.log('Age: 25');
  console.log('Occupation: Engineer');
  console.groupEnd();
▼ User Details VM504:1
  Name: Alice VM504:2
  Age: 25 VM504:3
  Occupation: Engineer VM504:4
< undefined
> console.time('myTimer');

// Code block to measure
for (let i = 0; i < 1000000; i++) {
  // Some time-consuming operations
  Math.sqrt(i);
}

console.timeEnd('myTimer');
myTimer: 3.699951171875 ms VM508:9
< undefined
>
```