DATA STRUCTURES

What is our GOAL for this MODULE?

We continued our coding journey tackling very advanced coding concepts like data structures. We learned to perform the sorting method on an array.

What did we ACHIEVE in the class TODAY?

- Got values from user input, stored and sorted them in an array.

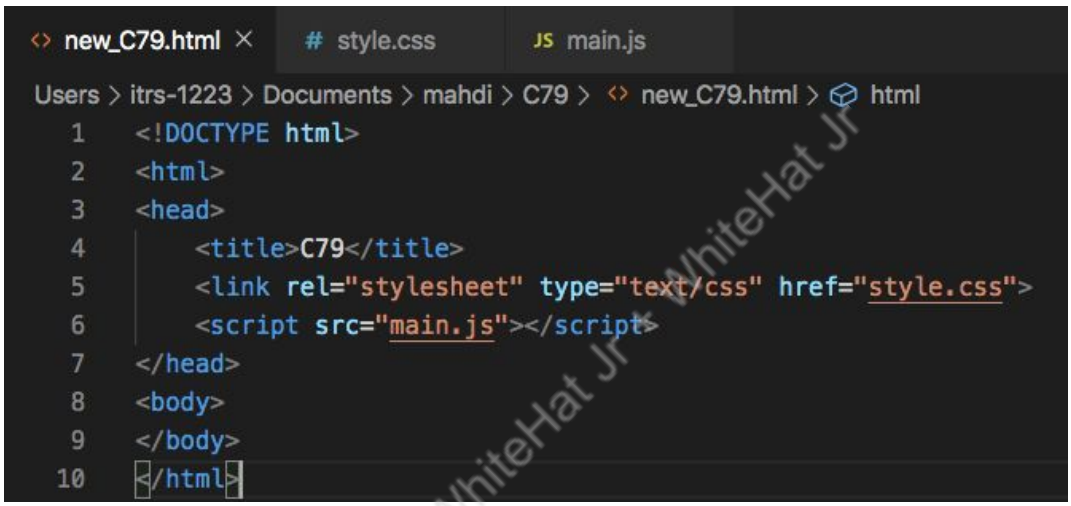
Which CONCEPTS/ CODING did we cover today?

- Got values from user input.
- Stored user input one by one inside an array.
- Sorted and displayed an array on a web page.

How did we DO the activities?

We began with a basic HTML code consisting of the **HTML**, **Head**, and **Title** tags including **CSS** and **JS** links in it.

HTML and CSS code:



Steps to create the heading for the web page:

1. Initiate a **center** tag to have everything in the center.
2. Use **h3** tag inside the **center** tag to hold the heading of the page and set the **style** attribute to **margin-top:20px and margin-bottom:20px** for the heading to get a 20-pixel space above and below.

```
<center>
<h3 style="margin-top: 20px; margin-bottom: 20px; ">
    List of the students in my class
</h3>
```

Steps to hold all the input boxes for the name of the students:

1. Open a **div** tag.
2. Use a **span** tag to hold the serial number.
3. Use an **input** tag to get the name of the first student and set the following attributes:
    - **placeholder** as **Name of the Student** to display a HINT for the input box
    - **id** as **name_of_student_1** to denote this is the first input box for the first

student

4. Break the line using a **br** tag.

```
<div>
<span>1</span>
<input type="text" placeholder="Name of the Student" id="name_of_the_student_1">
<br>
```

5. Provide a margin of 10 pixels by adding a CSS inside **style.css** for input boxes as we want them to look clean.

```
input
{
    margin: 10px;
}
```

6. Now, use another **input** tag to get the name of the second student and set the following attributes:
    - **placeholder** as **Name of the Student** to display a HINT for the input box
    - **id** as **name_of_student_2** to denote this is the second input box for the second student
7. Break the line using a **br** tag.

```
<span>2</span>
<input type="text" placeholder="Name of the Student" id="name_of_the_student_2">
<br>
```

8. Add two more input boxes by following the steps similar to the preceding two steps.

```
<span>3</span>
<input type="text" placeholder="Name of the Student" id="name_of_the_student_3">
<br>

<span>4</span>
<input type="text" placeholder="Name of the Student" id="name_of_the_student_4">
```

Steps to create a submit button:

1. Set an **onclick** function to the button to call the **submit()** function.
2. Give an **id** to reference this button.
    - **submit()** will be defined in JavaScript

```
<button onclick="submit()" id="submit_button">
    Submit
</button>
```

3. Provide a margin of 20 pixels at the top and bottom as the CSS for the **submit_button** inside **style.css**.

```
#submit_button
{
    margin-top: 20px;
    margin-bottom: 20px;
}
```

4. Now, have another **div** and give **id** to it so that we can later reference it in JavaScript.
    - This div will be used to show the names of the students and we will update this div with the names of the students.

```
<div id="display_name">

</div>
```

Steps to create a sort button:

1. Set an **onclick** function to the button to call the **sorting()** function.
2. Give an **id** to reference this button.
    - **sorting()** will be defined in JavaScript.

```
<button onclick="sorting()" id="sort_button">
    Sort
</button>
```

3. Provide a margin of 20 pixels at the top and bottom as the CSS for **sort_button** inside the **style.css** to make it look clean.
4. Set **display: none** to make the sort button display only when the submit button is clicked.

```
#sort_button
{
    display: none;
    margin-top: 20px;
    margin-bottom: 20px;
}
```

JavaScript Code in main.js:

```
    name_of_the_student_array = [];

    function submit()
    {

        var name_1 = document.getElementById("name_of_the_student_1").value;
        var name_2 = document.getElementById("name_of_the_student_2").value;
        var name_3 = document.getElementById("name_of_the_student_3").value;
        var name_4 = document.getElementById("name_of_the_student_4").value;

        name_of_the_student_array.push(name_1);
        name_of_the_student_array.push(name_2);
        name_of_the_student_array.push(name_3);
        name_of_the_student_array.push(name_4);

        console.log(name_of_the_student_array);

        document.getElementById("display_name").innerHTML = name_of_the_student_array;
        document.getElementById("submit_button").style.display = "none";
        document.getElementById("sort_button").style.display = "inline-block";

    }

function sorting()
{
    name_of_the_student_array.sort();
    console.log(name_of_the_student_array);
    document.getElementById("display_name").innerHTML = name_of_the_student_array;
}
```

Steps to achieve the **main.js** code given above:

1. Declare an empty array for storing names.

```
    name_of_the_student_array = [];
```

2. Write the code to declare the **submit** function.

```
function submit()
{
```

3. Get the names of the students by using **id** and store them in a variable.
   ***Note:** We will have 4 variables as we have 4 students (4 input boxes).

```
var name_1 = document.getElementById("name_of_the_student_1").value;
var name_2 = document.getElementById("name_of_the_student_2").value;
var name_3 = document.getElementById("name_of_the_student_3").value;
var name_4 = document.getElementById("name_of_the_student_4").value;
```

4. Now, push all these variables inside the array one by one.

```
name_of_the_student_array.push(name_1);
name_of_the_student_array.push(name_2);
name_of_the_student_array.push(name_3);
name_of_the_student_array.push(name_4);
```

5. Update the div which has the **id="display_name"** with these array values.

```
document.getElementById("display_name").innerHTML = name_of_the_student_array;
```

6. Hide the submit button and show the sort button.
   - **.style** - will add style to the selected element from the id
   - **.display** - is the CSS property
   - **none** - is the value of the property

```
document.getElementById("submit_button").style.display = "none";
document.getElementById("sort_button").style.display = "inline-block";
```

7. Write the code to declare the **sorting** function.

```
function sorting()
{
```

8. Sort the array which contains the name of the student using the **sort()** function.

```
name_of_the_student_array.sort();
```

9. Update the div which has the **id="display_name"** with the sorted array.

```
document.getElementById("display_name").innerHTML = name_of_the_student_array;
```

What's NEXT?

We will learn how to integrate loops and arrays. We will also learn how to create HTML elements dynamically using JS.

EXTEND YOUR KNOWLEDGE

Here are some Best References we've compiled together to enhance your knowledge and understanding of the concepts we learned today in the class. This will help you become pro at coding and creating industry-grade tech products!

Short Videos: Watch these Short Videos to understand the application of the concepts learned in class in real-world applications.

1. Reverse a String: https://www.youtube.com/watch?v=k7zzWvQaEhc
2. Learn Box Model: https://www.youtube.com/watch?v=rIO5326FgPE
3. HTML Input Types: https://www.youtube.com/watch?v=MKSQYsLLFEo

Coding Playground: Try out these code examples to get more practice in making Websites and Playstore ready apps.

1. 

: https://www.w3schools.com/jsref/jsref_reverse.asp
2. 

: https://www.w3schools.com/css/css_margin.asp
3. 

: https://www.w3schools.com/html/html_form_attributes.asp