Practical No. 4

Study of Java Script and DOM.

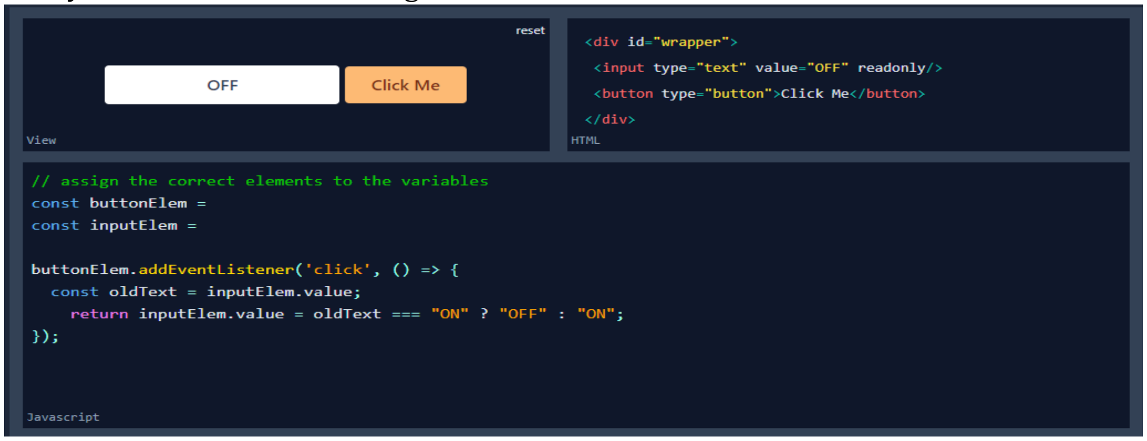
Perform following problem statements for DOM using Javascript

Problem Statement 0: Basics of DOM

- What is the DOM?
- What is DOM Tree Structure? Elaborate its elements with example (you may create DOM tree structure of previously created web pages)
- Give examples for following:
    - Accessing the DOM
    - Manipulating the DOM
    - Event Handling
    - Traversing the DOM
- What are Performance Considerations while implementing the DOM and can DOM supports all browsers?
- Elaborate Common Methods and Properties of DOM

Problem Statement 1: DOM selector methods

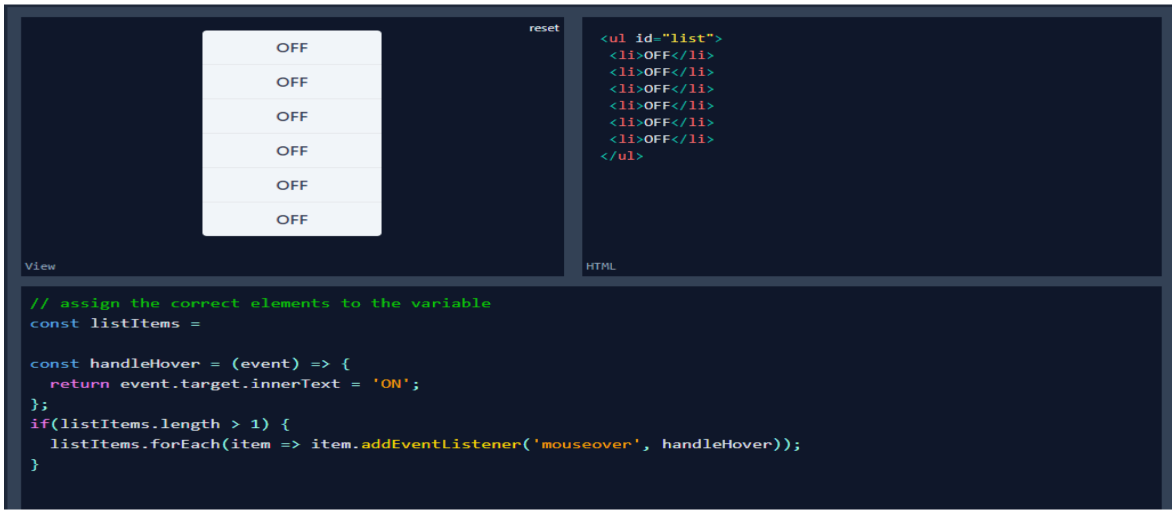
- Here, the existing code expects the variables 'buttonElem' and 'inputElem' to represent the button and input elements in the example UI. Assign the respective elements to the variables. In this case, the two elements do not have unique identifiers - like for example an id. Instead they are direct descendents of a div element with id 'wrapper'. Use an appropriate selector method! Click the button to verify that the code is working.



```html
<div id="wrapper">
  <input type="text" value="OFF" readonly/>
  <button type="button">Click Me</button>
</div>
```

```javascript
// assign the correct elements to the variables
const buttonElem =
const inputElem =

buttonElem.addEventListener('click', () => {
  const oldText = inputElem.value;
    return inputElem.value = oldText === "ON" ? "OFF" : "ON";
});
```

- In this scenario, we are looking for a list of elements gathered in one variable - rather than only one element. Assign the list items in the view to the variable 'listItems' by using an appropriate selector method. Once you have completed the code below, verify it by hovering over the list items until all items have the value 'ON'

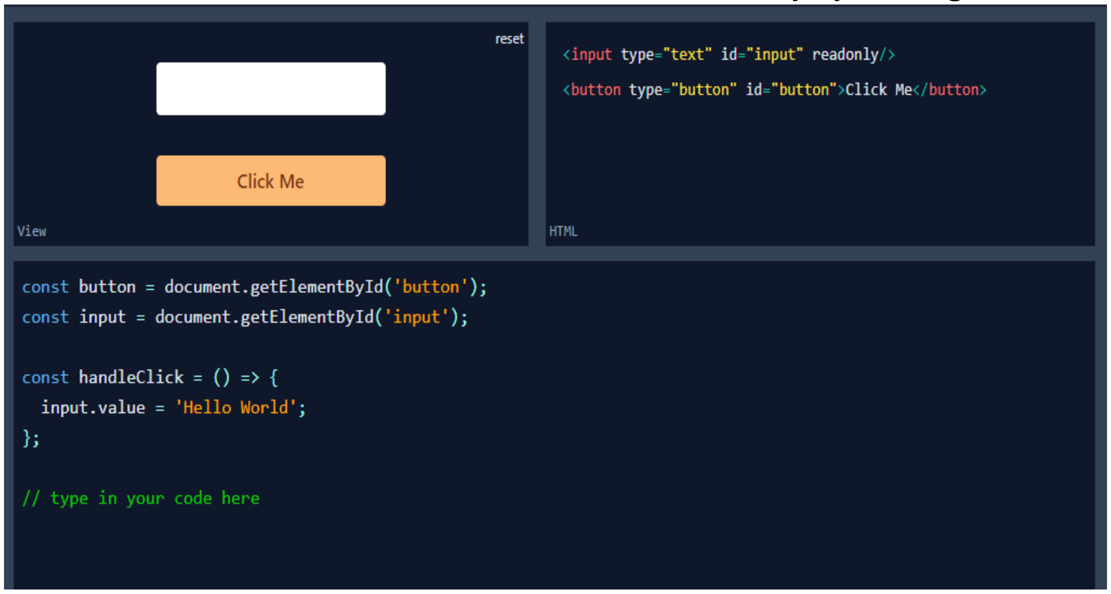


```
<ul id="list">
  <li>OFF</li>
  <li>OFF</li>
  <li>OFF</li>
  <li>OFF</li>
  <li>OFF</li>
  <li>OFF</li>
</ul>
```

```
// assign the correct elements to the variable
const listItems =

const handleHover = (event) => {
  return event.target.innerText = 'ON';
};
if(listItems.length > 1) {
  listItems.forEach(item => item.addEventListener('mouseover', handleHover));
}
```

Problem Statement 2: Events and user interactions

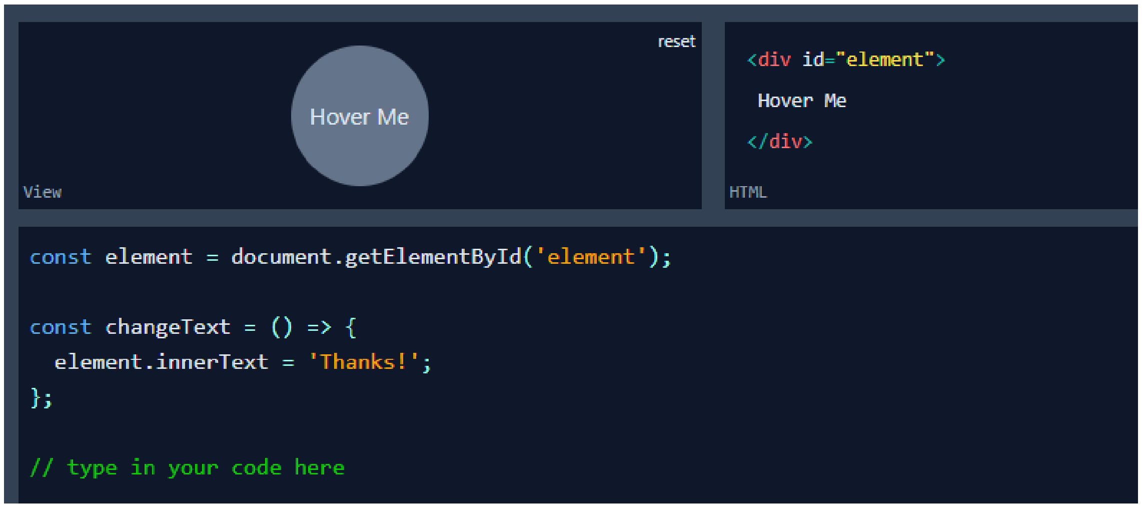
- The Javascript function handleText fills the input field with the words Hello World. But, there is no code to execute this function. Complete the existing code below such that the function is called when the button is clicked. Verify by clicking the button.



```
<input type="text" id="input" readonly/>

<button type="button" id="button">Click Me</button>
```

```
const button = document.getElementById('button');
const input = document.getElementById('input');

const handleClick = () => {
  input.value = 'Hello World';
};

// type in your code here
```

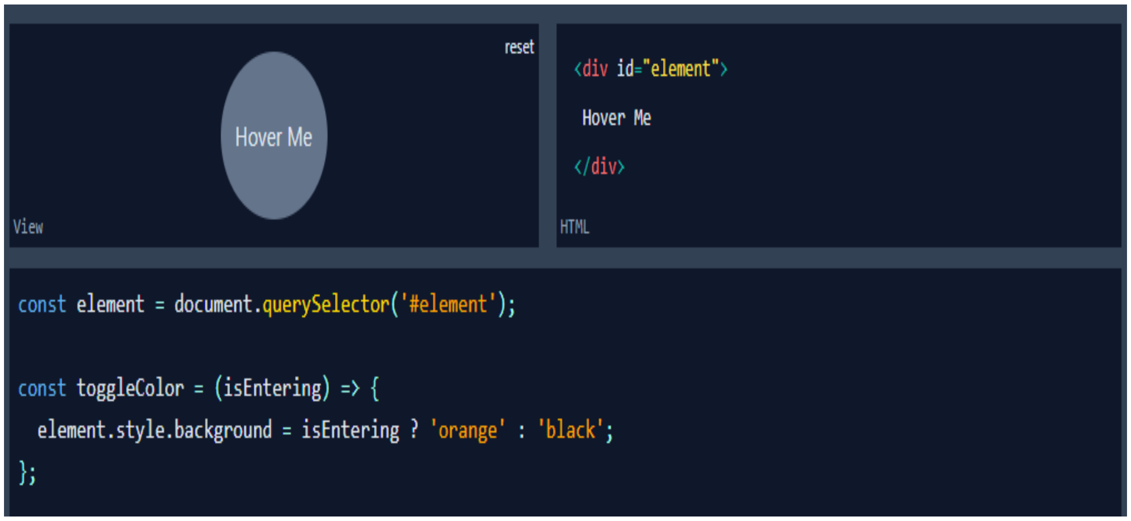
- The Javascript function changeText changes the text inside the circle. But again, there is no code to execute this function. Complete the existing code below such that the function is called when the cursor moves onto the circle. Verify that your code works by hovering over the circle.



```
<div id="element">
  Hover Me
</div>
```

```
const element = document.getElementById('element');

const changeText = () => {
  element.innerText = 'Thanks!';
};

// type in your code here
```

- In this scenario we want the color of the circle to change depending on the type of cursor movement. Use the function toggleColor to turn the circle orange when the cursor moves onto it. Reuse the same function to turn it black when the cursor leaves it. The tricky part is that you have to call toggleColor with different values for the parameter isEntering. Verify that your code is working by hovering the circle with the mouse cursor and leaving it again.



```
<div id="element">
  Hover Me
</div>
```

```
const element = document.querySelector('#element');

const toggleColor = (isEntering) => {
  element.style.background = isEntering ? 'orange' : 'black';
};
```

Problem Statement 3: DOM manipulation with JavaScript

- Remove element from the DOM. Create t2 circles red and green and a button clickme. Place them such a way that red circle hides the green circle. Add the function removeRedCircle to remove the circle with id red from the DOM when clicked on clickme button. Make sure that you really remove the element instead of just hiding it.

Problem Statement 4: DOM fundamentals

- Create JavaScript code to interact with the displayed HTML elements. Create a checkbox and a button. Once you click the button, the checkbox should be checked.
- Create 3 textboxes and a button. First 2 checkboxes contain first name and last name respectively. When the button is clicked, combine the names of the first two input fields. Insert the full name in the third input field (textbox). Check if your code still works if you change the first or last name.
- Create three buttons. One button displays value of 0. Other two buttons are for increment and reset. By clicking increment button each time, increase the value of the button by 1. By clicking the reset button set the value of button to 0. Confirm your code by clicking the buttons.
- create a dynamic input filter with JavaScript. Type a search term in the input field. The displayed items in the list should match your search term. The rest of the list elements should be hidden.
- Create 10 balloons as shown below. Every time you hover over a balloon, it should become invisible. Your goal is to pop all the balloons one after the other. Create a refresh button. After clicking refresh button it will again display all the balloons.

Problem Statement 5: Recursive functions

- Create a function move that moves the button 1px to the left or the right. It is recursive because it calls itself again and again. This keeps the button moving. Extend the JavaScript code. Once you click the button, it should stop moving. When you click it again, it should move again.

Note:

1. Create a **document** of the above website with screenshots.
2. Scan the document and **create a pdf file** with **"ExamSeatNum_P#PS#" as its name**.
3. Upload the file on the **WCE Moodle** before the given deadline.