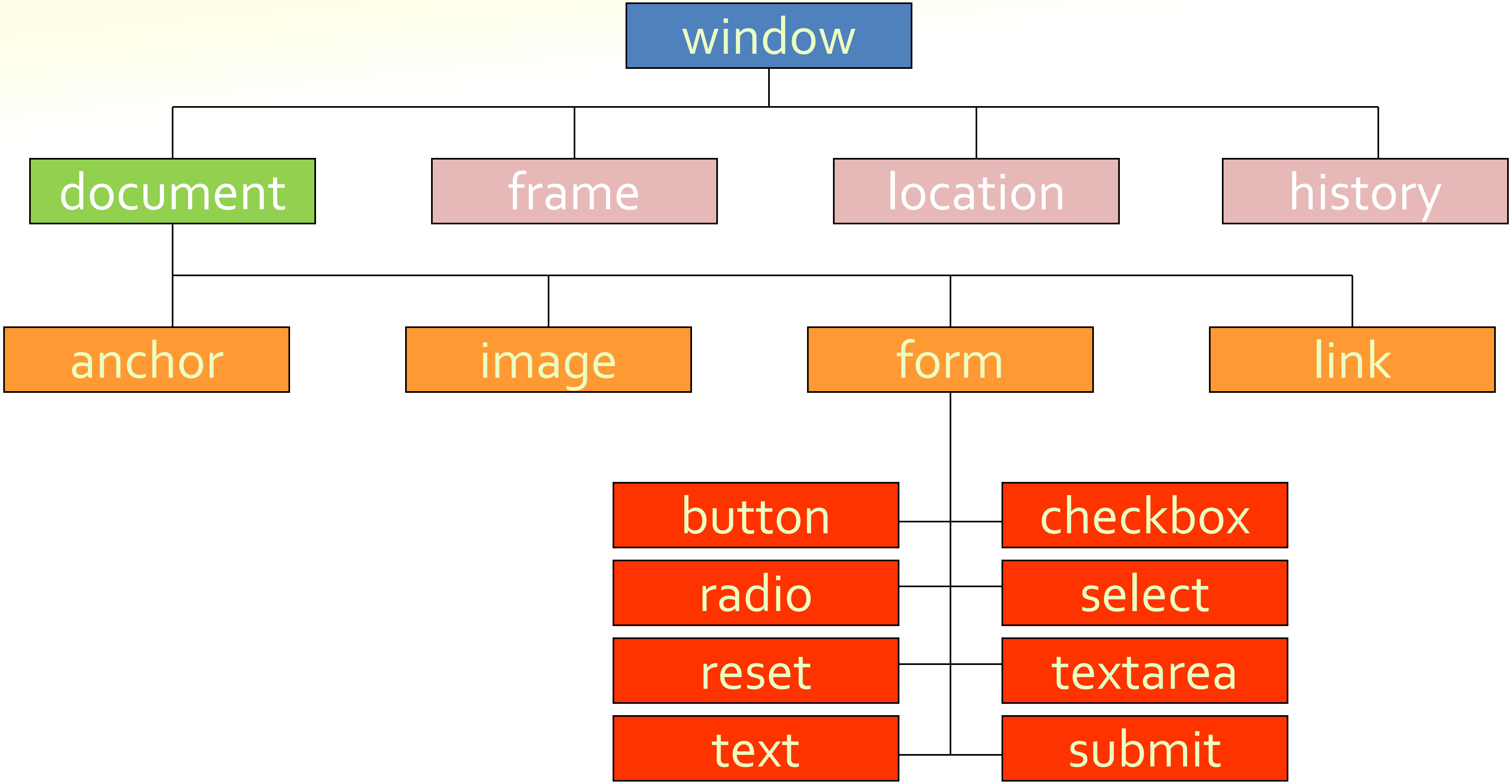
# DOM & Selection Methods

Dr. Jenila Livingston L.M.
Professor
VIT Chennai

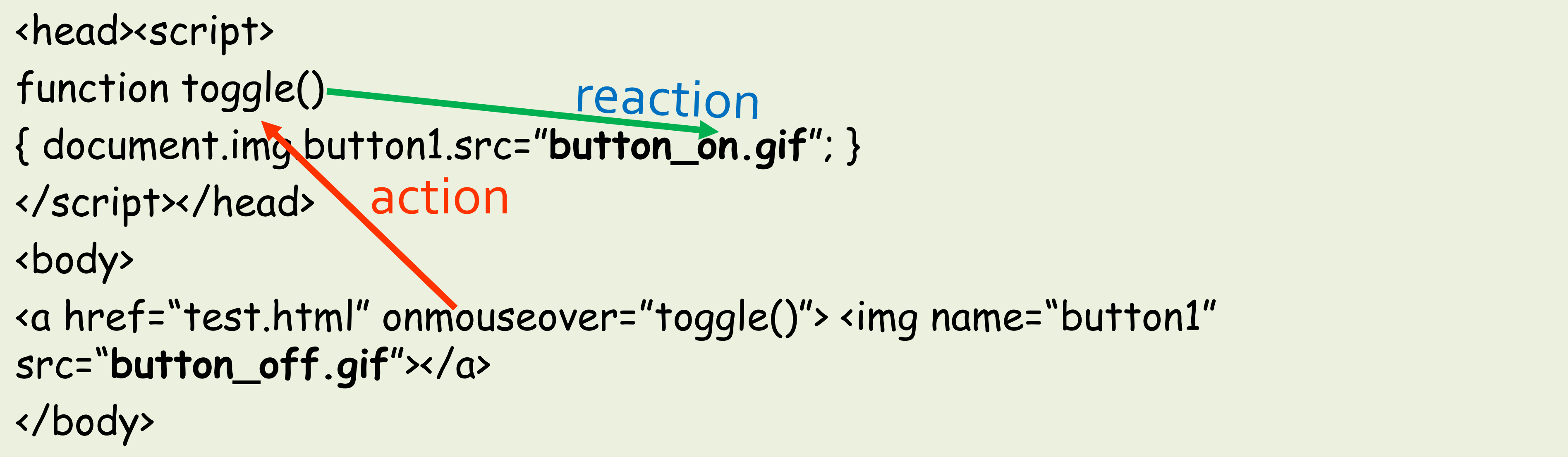
# DOM

- JavaScript also provides you with **objects that can control and manipulate the displays** of browsers.
  - More dynamic and interactive.
- DOM is an object-oriented model that describes **how all elements** in an HTML page **are arranged**.
- The HTML DOM is a standard for how to **get, change, add, or delete** HTML elements.

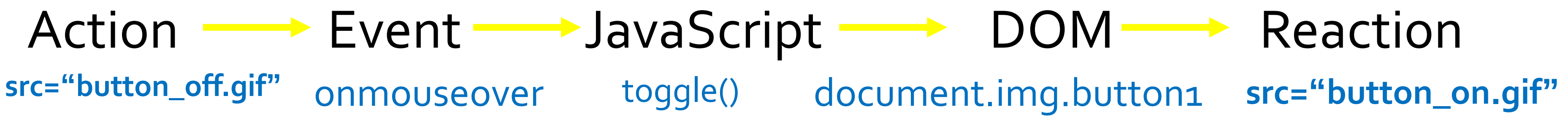
# DOM


window
document
frame
location
history
anchor
image
form
link
button
checkbox
radio
select
reset
textarea
text
submit

# How the DOM works?



```
<head><script>
function toggle()
{ document.img.button1.src=“button_on.gif”; }
</script></head>
<body>
<a href=“test.html” onmouseover=“toggle()”> <img name=“button1”
src=“button_off.gif”></a>
</body>
```



| Action → | Event → | JavaScript → | DOM → | Reaction |
|---|---|---|---|---|
| **src=“button_off.gif”** | onmouseover | toggle() | document.img.button1 | **src=“button_on.gif”** |

1) User moves mouse over object
2) Event senses that something happened to the object
3) JavaScript tells the object what to do (Even handler)
4) Locates object on the web page
5) Object’s image source is changed

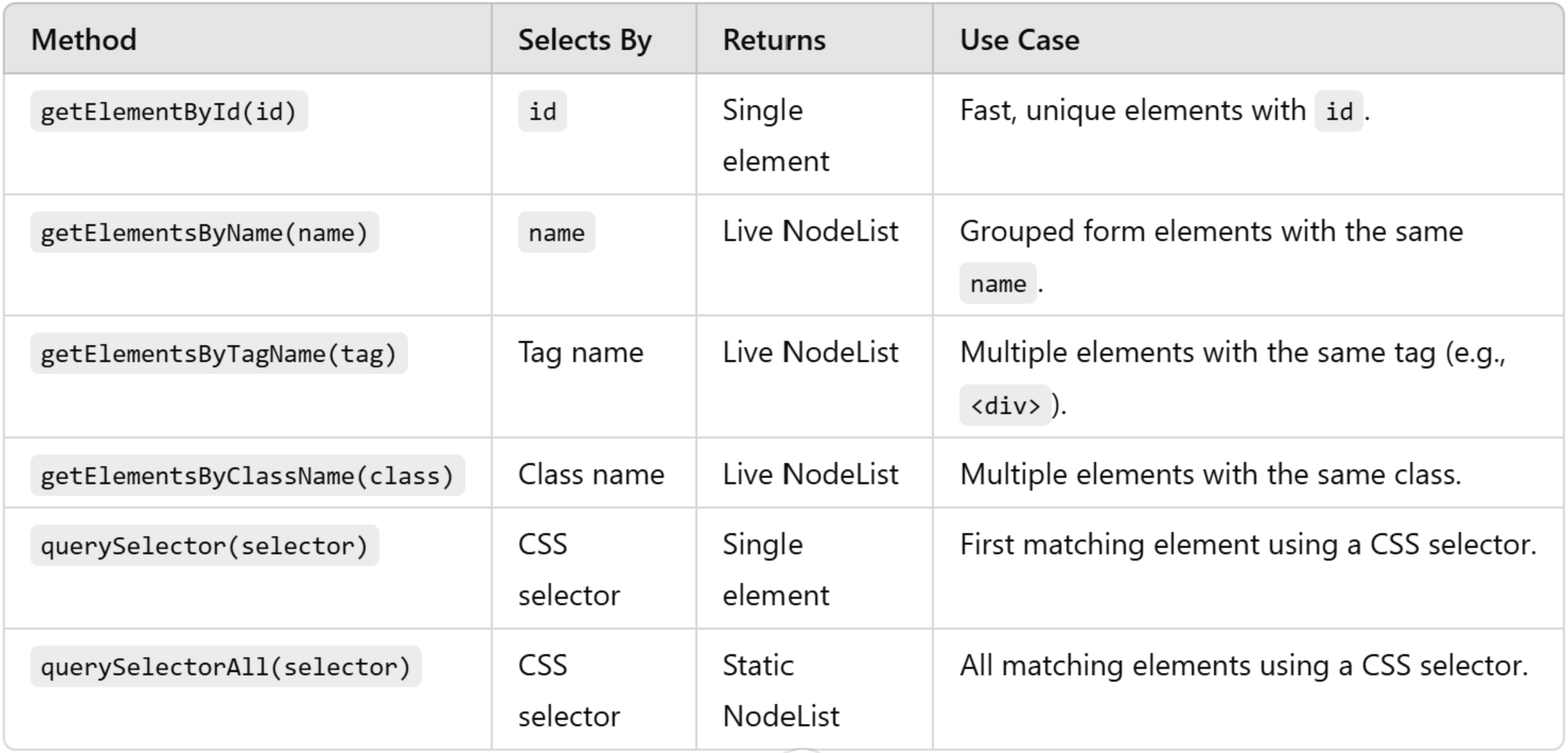
# HTML DOM

- HTML DOM **methods (fuction)** are **actions** you can perform (on HTML Elements).
- HTML DOM **properties** are **values** (of HTML Elements) that you can set or change.

# DOM Selection methods

| Method | Description |
|---|---|
| getElementById(id) | Find an element-by-element id |
| getElemnetsBYName(name) | Find elements by Name |
| getElementsByTagName(tag) | Find elements by tag name |
| getElementsByClassName(class_name) | Find elements by class name |
| querySelector(selector) | Find element by CSS selector |
| querySelectorAll(selector) | Find elements by CSS selector<br>Alternate for getElementsByName |

# DOM Selection methods



| Method | Selects By | Returns | Use Case |
|---|---|---|---|
| `getElementById(id)` | `id` | Single element | Fast, unique elements with `id`. |
| `getElementsByName(name)` | `name` | Live NodeList | Grouped form elements with the same `name`. |
| `getElementsByTagName(tag)` | Tag name | Live NodeList | Multiple elements with the same tag (e.g., `<div>`). |
| `getElementsByClassName(class)` | Class name | Live NodeList | Multiple elements with the same class. |
| `querySelector(selector)` | CSS selector | Single element | First matching element using a CSS selector. |
| `querySelectorAll(selector)` | CSS selector | Static NodeList | All matching elements using a CSS selector. |

**Note: NodeList:** Array like Collection. First element index is 0

A **Static NodeList** does **not** update when the document changes.

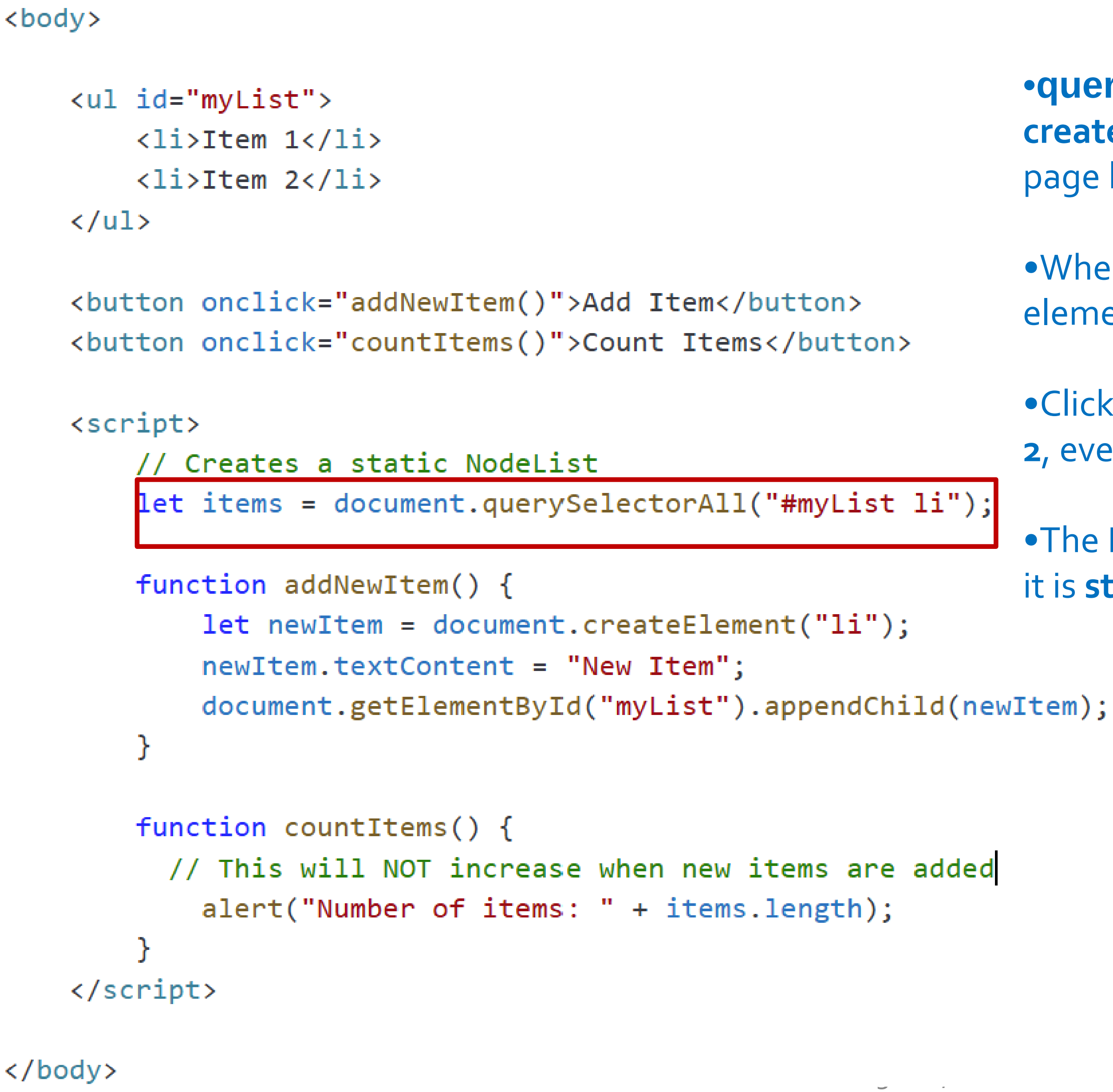
# Which One Should You Use?

## *Live NodeLists or Static NodeLists*

- **Use Live NodeLists** (getElementsByName(), etc.) when you want to track changes dynamically.

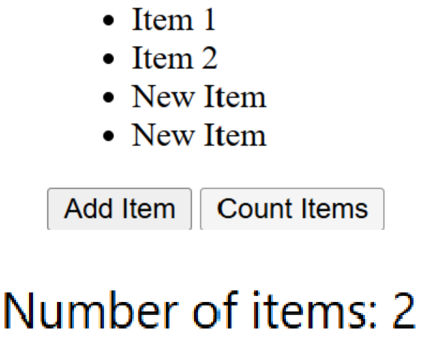
- **Use Static NodeLists** (querySelectorAll()) when you don’t want automatic updates (better performance in large documents).

# Static NodeList



```
<body>

    <ul id="myList">
        <li>Item 1</li>
        <li>Item 2</li>
    </ul>

    <button onclick="addNewItem()">Add Item</button>
    <button onclick="countItems()">Count Items</button>

    <script>
        // Creates a static NodeList
        let items = document.querySelectorAll("#myList li");

        function addNewItem() {
            let newItem = document.createElement("li");
            newItem.textContent = "New Item";
            document.getElementById("myList").appendChild(newItem);
        }

        function countItems() {
          // This will NOT increase when new items are added
            alert("Number of items: " + items.length);
        }
    </script>

</body>
```

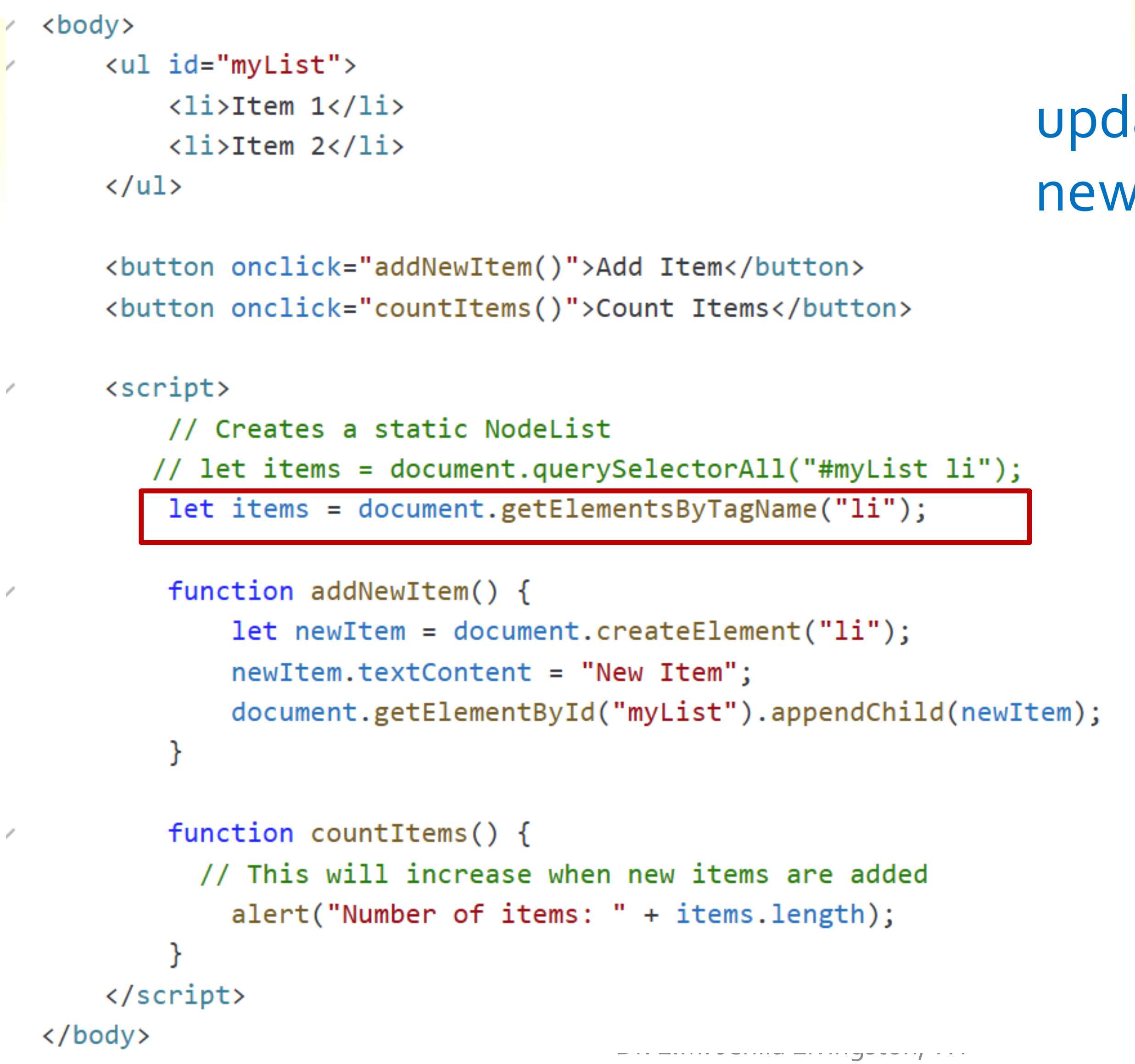
- **querySelectorAll("#myList li") creates a static NodeList** when the page loads.
- When clicking "Add Item", a new <li> element is added dynamically.
- Clicking "Count Items" will always show **2**, even after adding new items.
- The NodeList does not update because it is **static**.



- Item 1
- Item 2
- New Item
- New Item

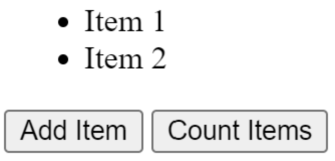
Number of items: 2

# Dynamic NodeLIst

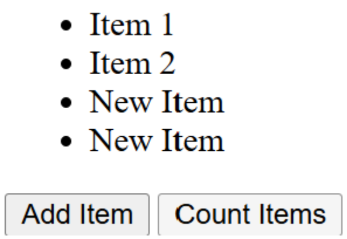


```
<body>
    <ul id="myList">
        <li>Item 1</li>
        <li>Item 2</li>
    </ul>

    <button onclick="addNewItem()">Add Item</button>
    <button onclick="countItems()">Count Items</button>

    <script>
        // Creates a static NodeList
       // let items = document.querySelectorAll("#myList li");
        let items = document.getElementsByTagName("li");

        function addNewItem() {
            let newItem = document.createElement("li");
            newItem.textContent = "New Item";
            document.getElementById("myList").appendChild(newItem);
        }

        function countItems() {
          // This will increase when new items are added
            alert("Number of items: " + items.length);
        }
    </script>
</body>
```

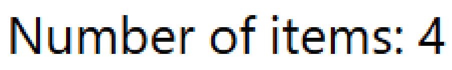
updates the count when new items are added



- Item 1
- Item 2

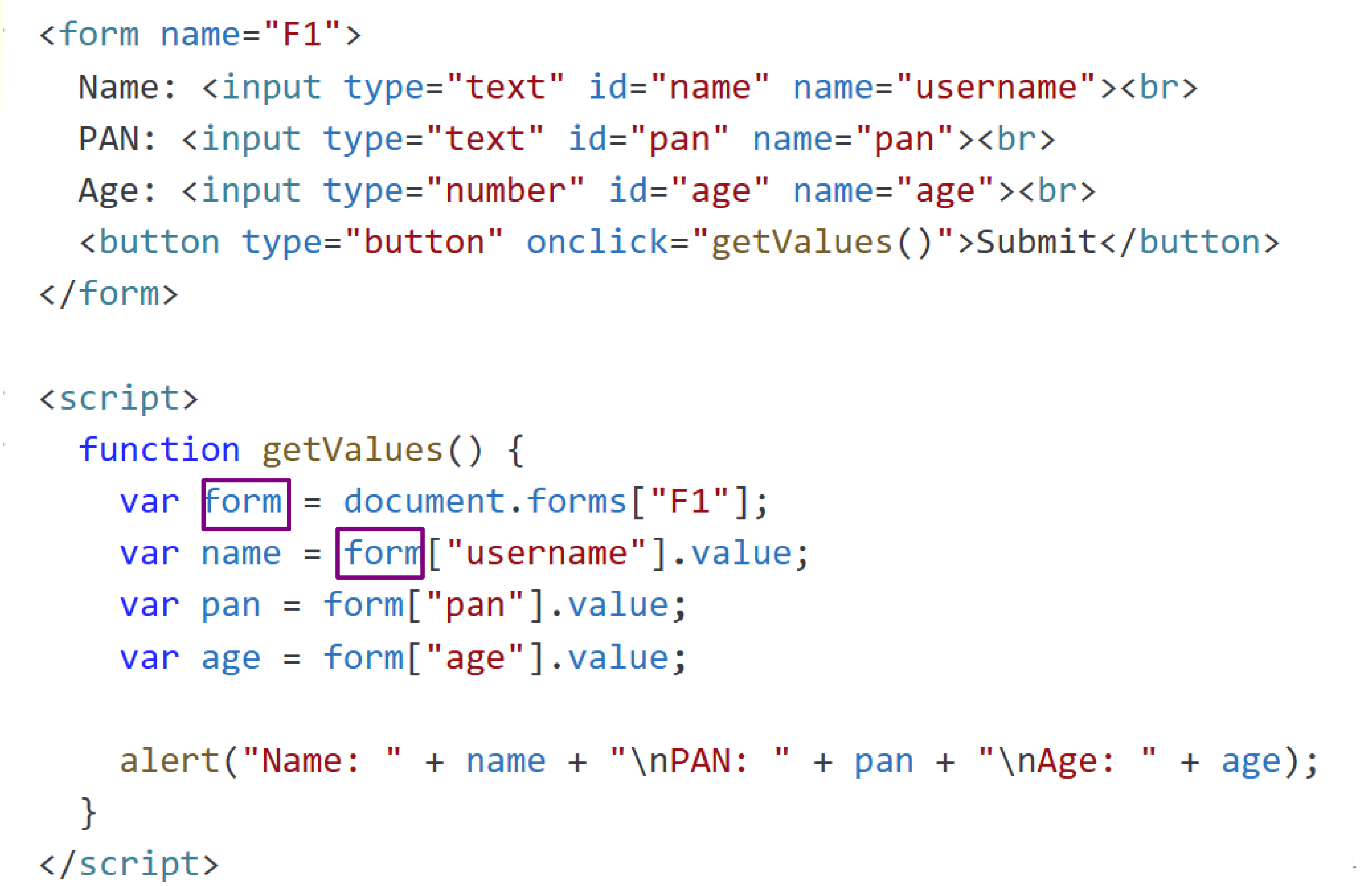


- Item 1
- Item 2
- New Item
- New Item



Number of items: 4

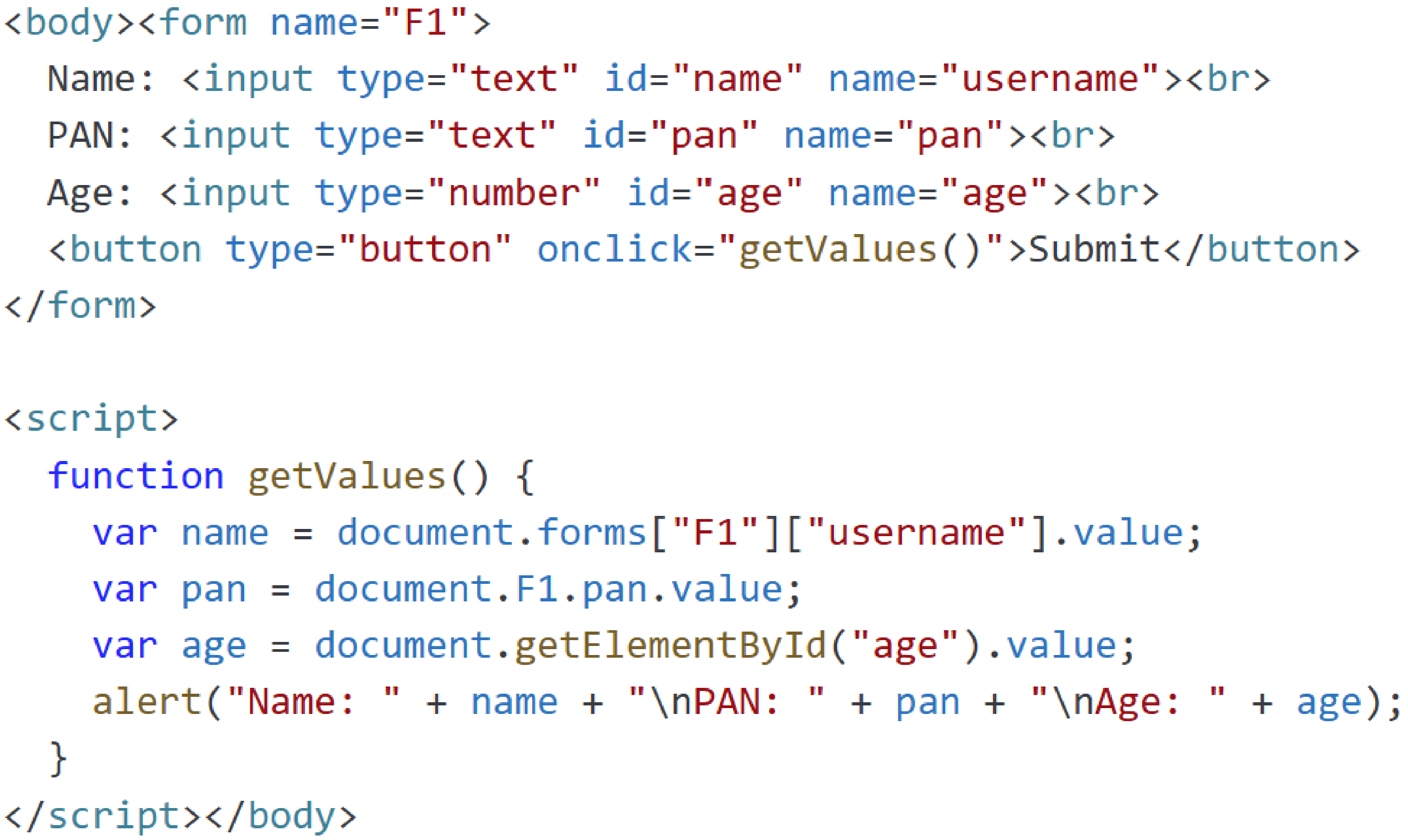
# Accessing Form Fields



```
<form name="F1">
  Name: <input type="text" id="name" name="username"><br>
  PAN: <input type="text" id="pan" name="pan"><br>
  Age: <input type="number" id="age" name="age"><br>
  <button type="button" onclick="getValues()">Submit</button>
</form>

<script>
  function getValues() {
    var form = document.forms["F1"];
    var name = form["username"].value;
    var pan = form["pan"].value;
    var age = form["age"].value;

    alert("Name: " + name + "\nPAN: " + pan + "\nAge: " + age);
  }
</script>
```

# Accessing Input Field Values
## (Text, Number, Password, etc.)

- **document.forms[“formname”][“fieldname”].value**
- **document.formname.filedname.value**
- **getElementById**

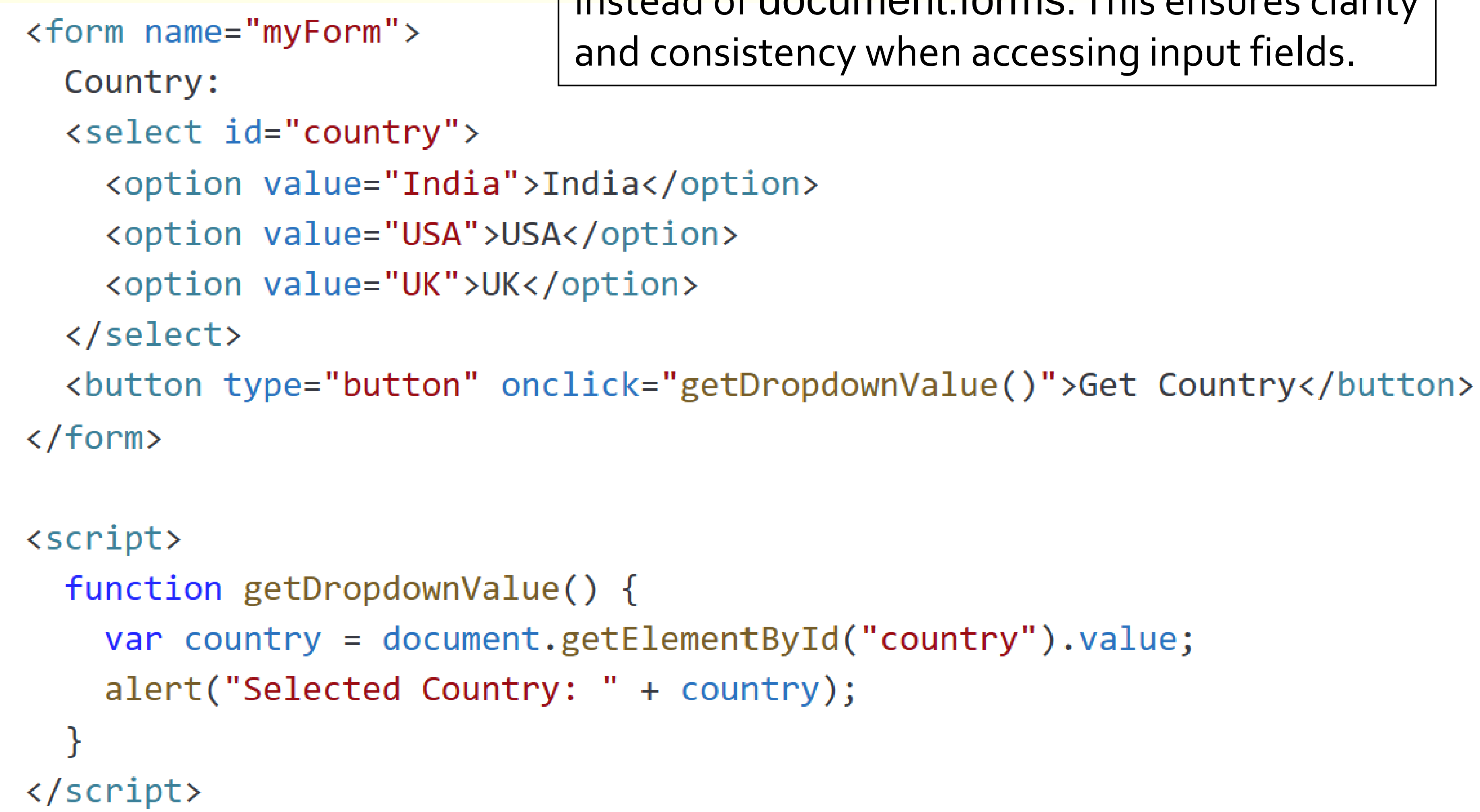


```
<body><form name="F1">
  Name: <input type="text" id="name" name="username"><br>
  PAN: <input type="text" id="pan" name="pan"><br>
  Age: <input type="number" id="age" name="age"><br>
  <button type="button" onclick="getValues()">Submit</button>
</form>

<script>
  function getValues() {
    var name = document.forms["F1"]["username"].value;
    var pan = document.F1.pan.value;
    var age = document.getElementById("age").value;
    alert("Name: " + name + "\nPAN: " + pan + "\nAge: " + age);
  }
</script></body>
```

# Accessing Dropdown (Select) Values

## getElementById

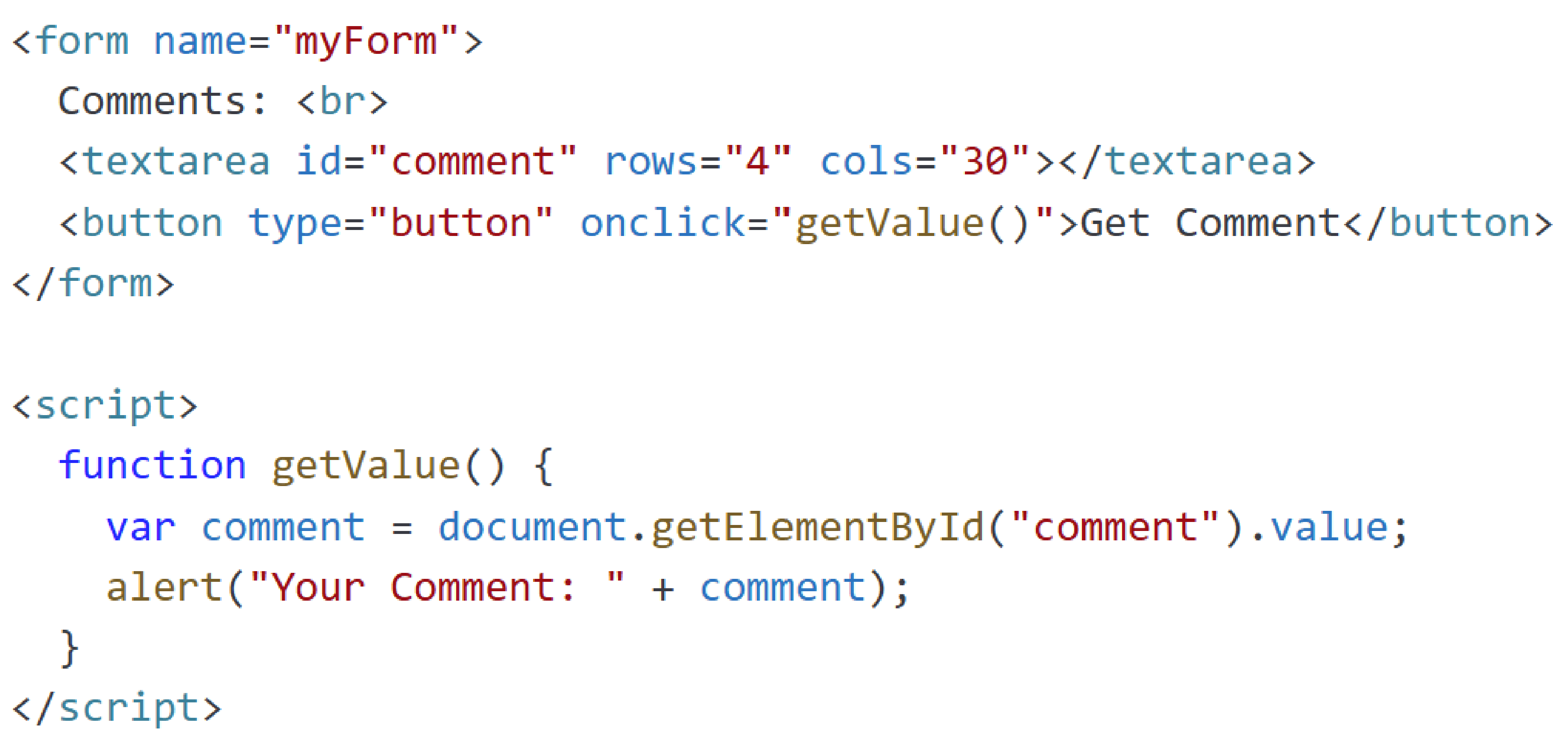
getElementById for fetching form values instead of document.forms. This ensures clarity and consistency when accessing input fields.



```
<form name="myForm">
  Country:
  <select id="country">
    <option value="India">India</option>
    <option value="USA">USA</option>
    <option value="UK">UK</option>
  </select>
  <button type="button" onclick="getDropdownValue()">Get Country</button>
</form>

<script>
  function getDropdownValue() {
    var country = document.getElementById("country").value;
    alert("Selected Country: " + country);
  }
</script>
```

# Accessing Textarea Value

## getElementById



```
<form name="myForm">
  Comments: <br>
  <textarea id="comment" rows="4" cols="30"></textarea>
  <button type="button" onclick="getValue()">Get Comment</button>
</form>

<script>
  function getValue() {
    var comment = document.getElementById("comment").value;
    alert("Your Comment: " + comment);
  }
</script>
```

# getElemnetsByName

- The getElementsByName() method returns a collection of all elements in the document with the specified name (the **value** of the name attribute), as an HTMLCollection object.
- The HTMLCollection object represents a collection of nodes. The nodes can be accessed by index numbers. **The index starts at 0**.
- **Tip:** You can use the **length property** to determine the number of elements, then **loop through all elements** and fetch the info.

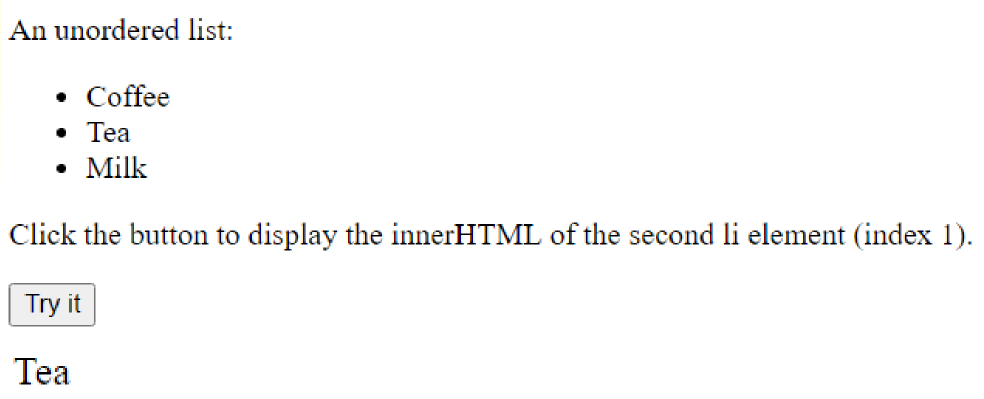
# Accessing radio button and combo box

```
<html>  <head>  <title> GetElementsByName </title> </head>
 <body> <p> Select a radio button and an option from Combo box. </p>
    Gender:
   <input type="radio" name="gender" value="Male">Male
   <input type="radio" name="gender" value="Female">Female
   <input type="radio" name="gender" value="Others">Others
   <br> Select Your Degree<Br>
   <select>
        <option name="degree" value="CSE">CSE
        <option name="degree" value="ECE">ECE
        <option name="degree" value="EEE">EEE
        <option name="degree" value="Mech">Mech
        <option name="degree" value="Civil">Civil
        </select><Br>
   <button type="button" onclick="displayValue()"> Submit </button> <br>
   <div id="result"></div>
   <div id="result2"></div>
```

```
<script>
      function displayValue() {
         var ele1 = document.getElementsByName('gender’); //creates an array
         var ele2= document.getElementsByName('degree');
         for(i = 0; i < ele1.length; i++) {
            if(ele1[i].checked)
            document.getElementById("result").innerHTML
                  = "Gender: "+ele1[i].value;
         }
            for(i = 0; i < ele2.length; i++) {
            if(ele2[i].selected)
            document.getElementById("result2").innerHTML
                  = "Degree: "+ele2[i].value;
         }
      }
   </script> </body>
</html>
```

# getElementsByTagName

```
<!DOCTYPE html>
<html>
<body>
<p>An unordered list:</p>
<ul>
  <LI>Coffee</LI>
  <LI>Tea</LI>
  <LI>Milk</LI>
</ul>
<p>Click the button to display the innerHTML of the second li element (index 1).</p>
<button onclick="myFunction()">Try it</button>
<p id="demo"></p>
<script>
function myFunction() {
  var x = document.getElementsByTagName("LI");
  document.getElementById("demo").innerHTML = x[1].innerHTML;
}
</script>
</body>
</html>
```



An unordered list:

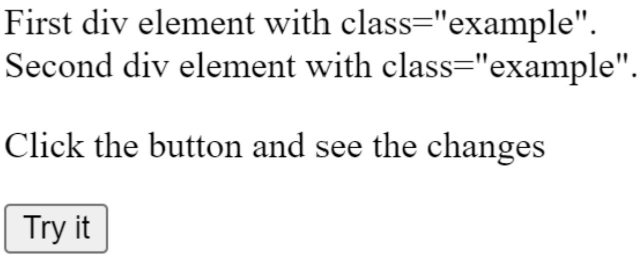
- Coffee
- Tea
- Milk

Click the button to display the innerHTML of the second li element (index 1).

Tea

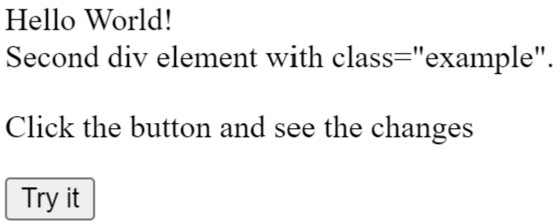
# document.getElementsByClassName

```
<!DOCTYPE html>
<html>
<body>
<div class="example">First div element with class="example".</div>
<div class="example">Second div element with class="example".</div>
<p>Click the button and see the changes</p>
<button onclick="myFunction()">Try it</button>
<script>
function myFunction() {
  var x = document.getElementsByClassName("example");
  x[0].innerHTML = "Hello World!";
}
</script>
</body>
</html>
```



First div element with class="example".
Second div element with class="example".

Click the button and see the changes

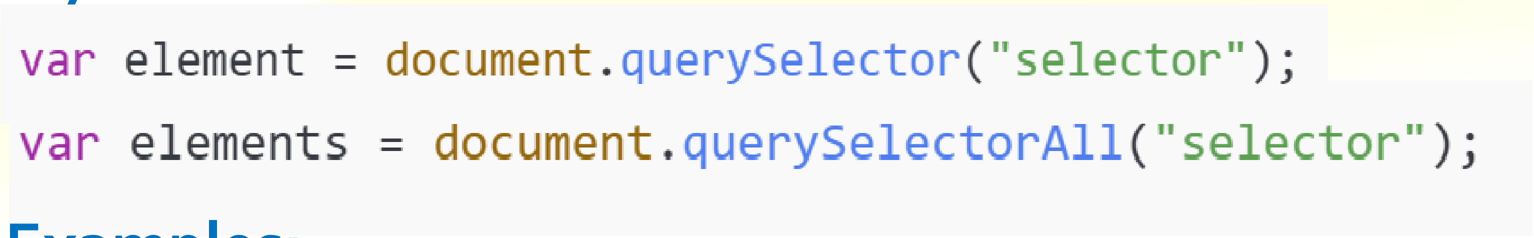


Hello World!
Second div element with class="example".

Click the button and see the changes

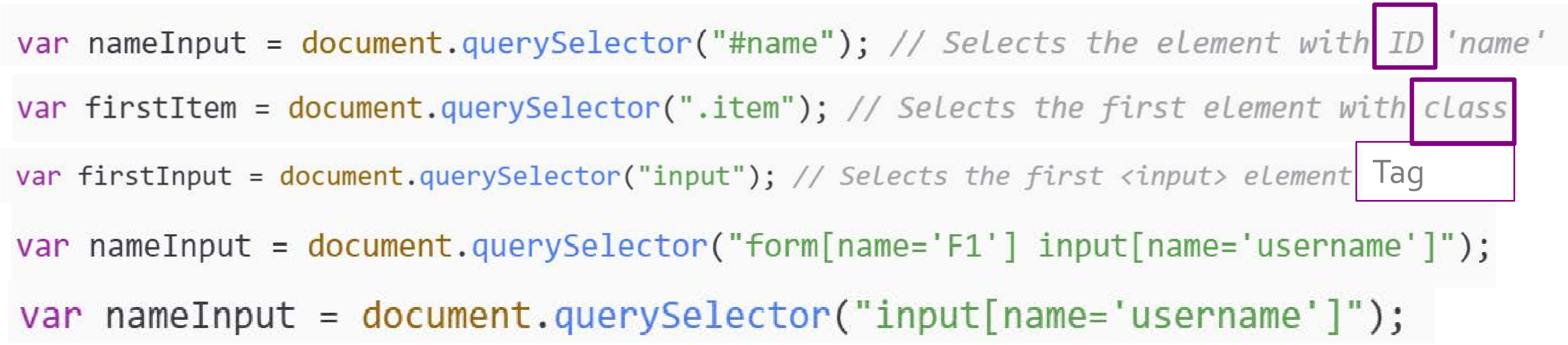
# querySelector and querySelectorAll

## Syntax:



```
var element = document.querySelector("selector");
var elements = document.querySelectorAll("selector");
```

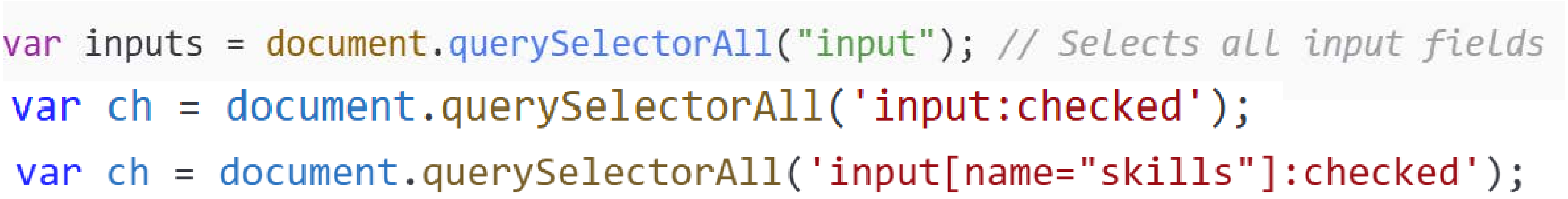
## Examples:



```
var nameInput = document.querySelector("#name"); // Selects the element with ID 'name'
var firstItem = document.querySelector(".item"); // Selects the first element with class
var firstInput = document.querySelector("input"); // Selects the first <input> element
var nameInput = document.querySelector("form[name='F1'] input[name='username']");
var nameInput = document.querySelector("input[name='username']");
```

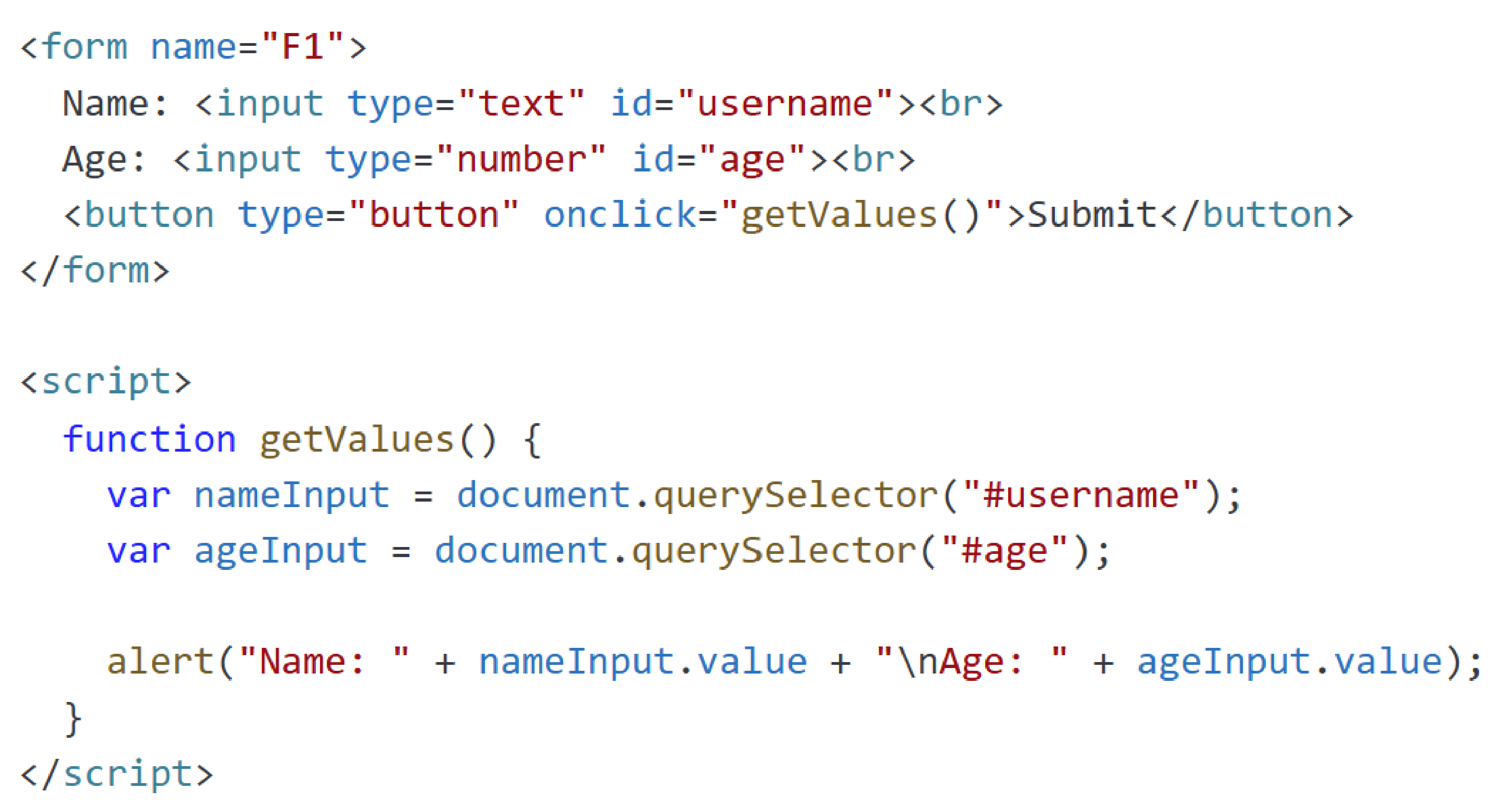
Tag

// it selects the **first** <input> element with the attribute name="username".



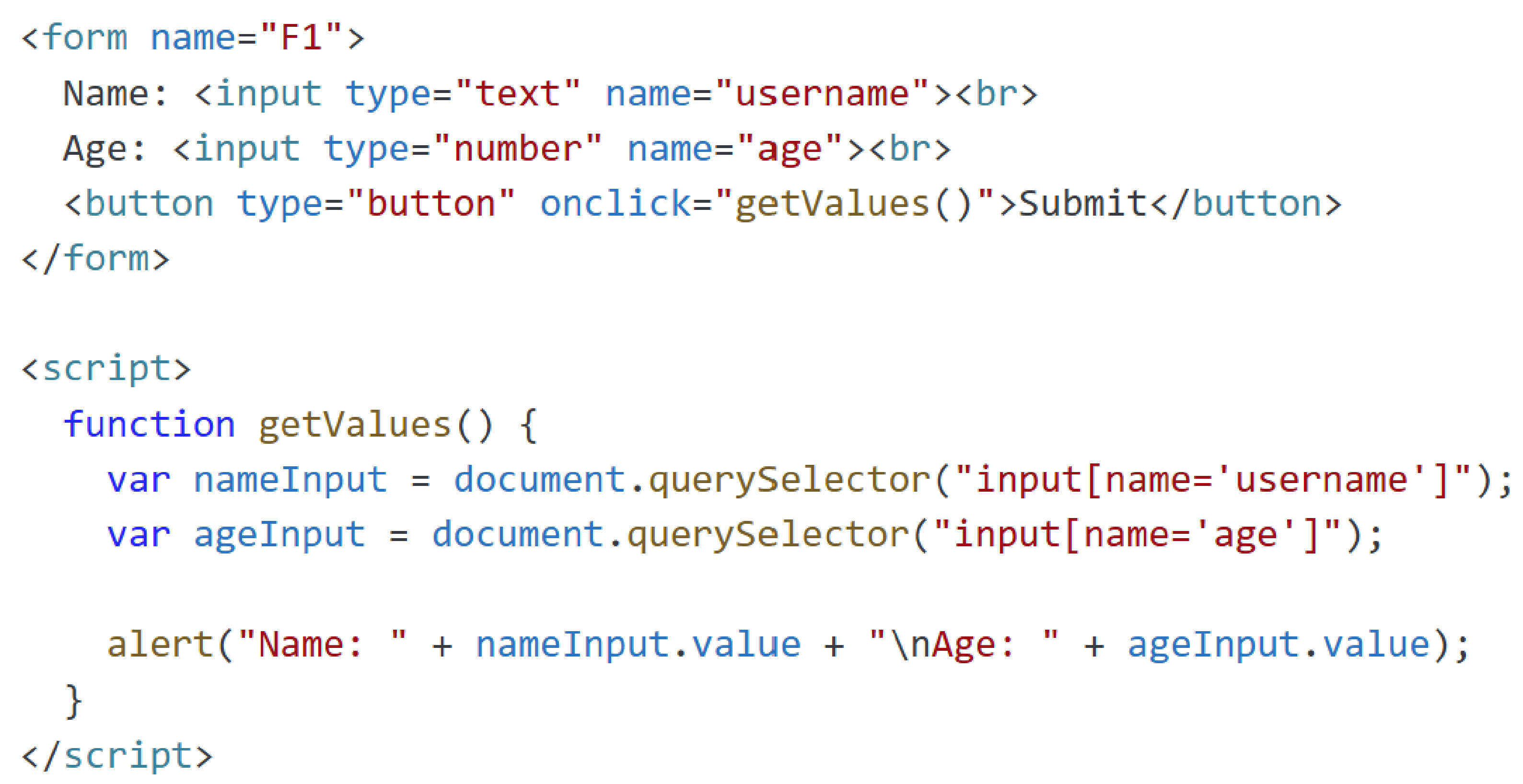
```
var inputs = document.querySelectorAll("input"); // Selects all input fields
var ch = document.querySelectorAll('input:checked');
var ch = document.querySelectorAll('input[name="skills"]:checked');
```

# querySelector



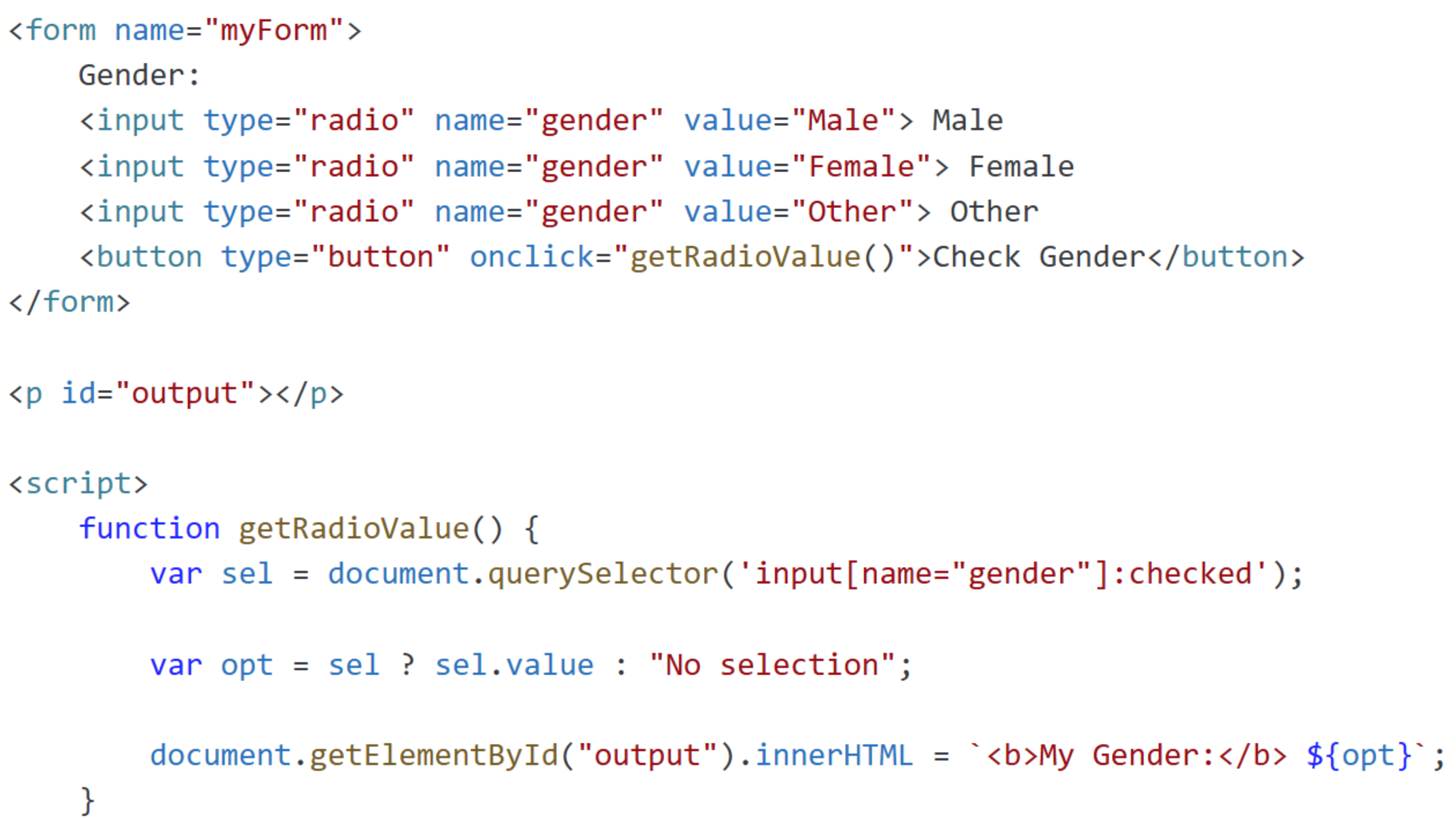
```
<form name="F1">
  Name: <input type="text" id="username"><br>
  Age: <input type="number" id="age"><br>
  <button type="button" onclick="getValues()">Submit</button>
</form>

<script>
  function getValues() {
    var nameInput = document.querySelector("#username");
    var ageInput = document.querySelector("#age");

    alert("Name: " + nameInput.value + "\nAge: " + ageInput.value);
  }
</script>
```

# querySelector

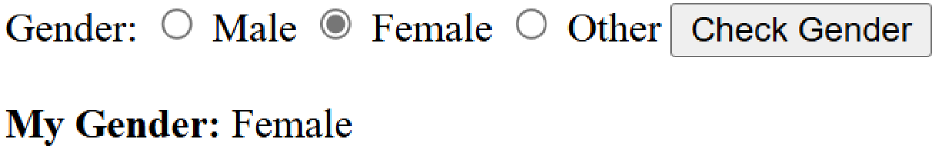


```
<form name="F1">
  Name: <input type="text" name="username"><br>
  Age: <input type="number" name="age"><br>
  <button type="button" onclick="getValues()">Submit</button>
</form>

<script>
  function getValues() {
    var nameInput = document.querySelector("input[name='username']");
    var ageInput = document.querySelector("input[name='age']");

    alert("Name: " + nameInput.value + "\nAge: " + ageInput.value);
  }
</script>
```

# Accessing Radio Button Values
# querySelector

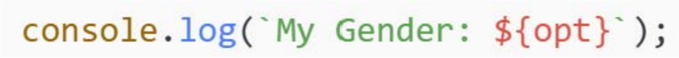


```
<form name="myForm">
    Gender:
    <input type="radio" name="gender" value="Male"> Male
    <input type="radio" name="gender" value="Female"> Female
    <input type="radio" name="gender" value="Other"> Other
    <button type="button" onclick="getRadioValue()">Check Gender</button>
</form>

<p id="output"></p>

<script>
    function getRadioValue() {
        var sel = document.querySelector('input[name="gender"]:checked');

        var opt = sel ? sel.value : "No selection";

        document.getElementById("output").innerHTML = `<b>My Gender:</b> ${opt}`;
    }
</script>
```



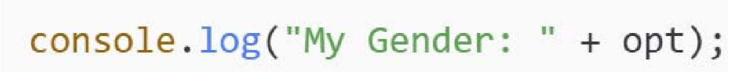
Gender: ○ Male ● Female ○ Other [Check Gender]

**My Gender:** Female



```
console.log(`My Gender: ${opt}`);
```

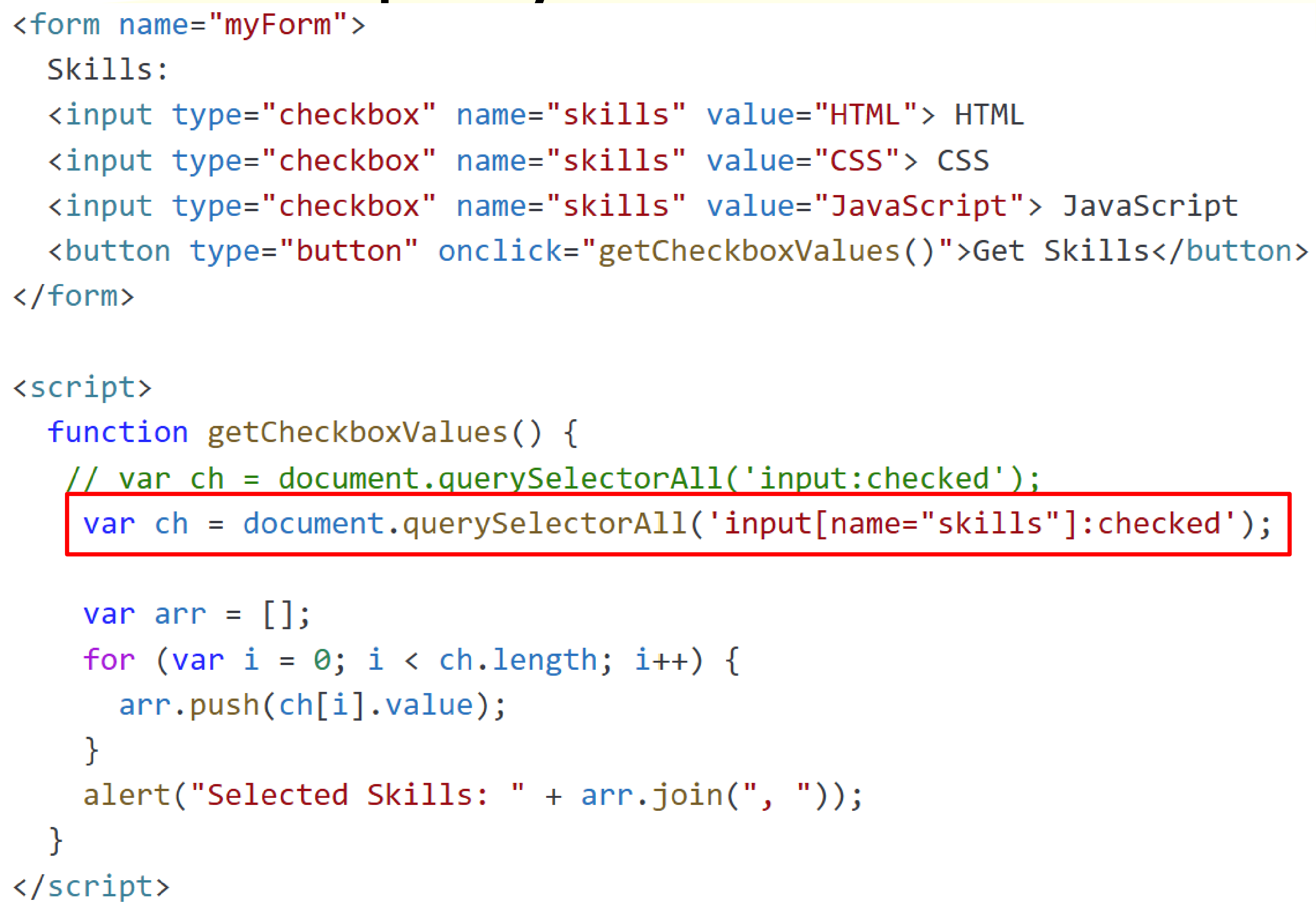
or



```
console.log("My Gender: " + opt);
```

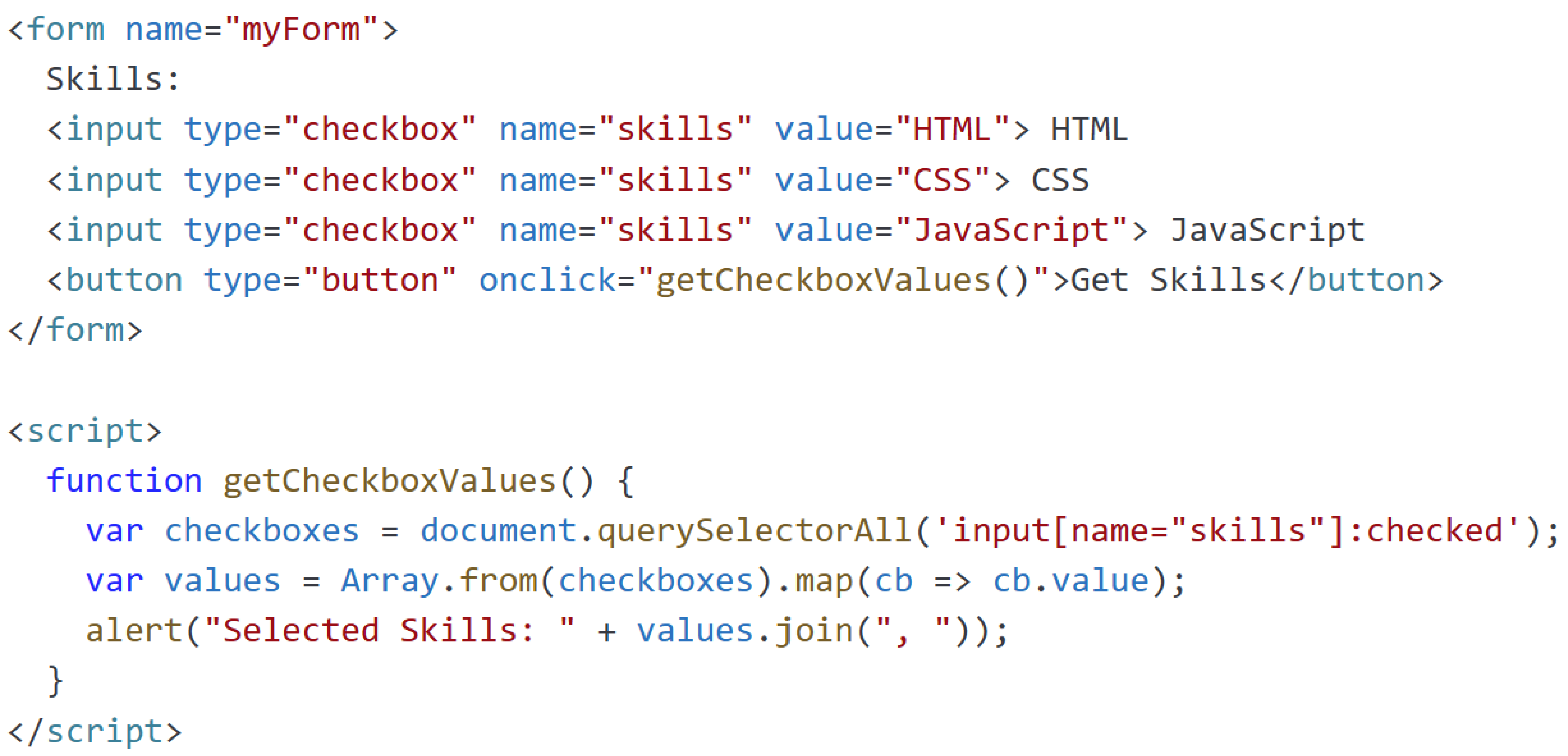
# Accessing Checkbox Values

# querySelectorAll



```
<form name="myForm">
  Skills:
  <input type="checkbox" name="skills" value="HTML"> HTML
  <input type="checkbox" name="skills" value="CSS"> CSS
  <input type="checkbox" name="skills" value="JavaScript"> JavaScript
  <button type="button" onclick="getCheckboxValues()">Get Skills</button>
</form>

<script>
  function getCheckboxValues() {
   // var ch = document.querySelectorAll('input:checked');
    var ch = document.querySelectorAll('input[name="skills"]:checked');

    var arr = [];
    for (var i = 0; i < ch.length; i++) {
      arr.push(ch[i].value);
    }
    alert("Selected Skills: " + arr.join(", "));
  }
</script>
```

# Accessing Checkbox Values
# querySelectorAll



```
<form name="myForm">
  Skills:
  <input type="checkbox" name="skills" value="HTML"> HTML
  <input type="checkbox" name="skills" value="CSS"> CSS
  <input type="checkbox" name="skills" value="JavaScript"> JavaScript
  <button type="button" onclick="getCheckboxValues()">Get Skills</button>
</form>

<script>
  function getCheckboxValues() {
    var checkboxes = document.querySelectorAll('input[name="skills"]:checked');
    var values = Array.from(checkboxes).map(cb => cb.value);
    alert("Selected Skills: " + values.join(", "));
  }
</script>
```

Thank You