CSC-336
Web Technologies

Lecture 10

Topics:

- Incorporating JavaScript into Web Pages
- Document Object Model (DOM)

Muhammad Naveed Shaikh
Department of Computer Engineering
naveedshaikh@cuiatd.edu.pk

COMSATS University Islamabad | Abbottabad Campus

Why Integrate JavaScript into Web Pages?

- Console practice builds logic
  - but real power appears in the browser
- Brings websites to life with dynamic content
- Enables user interaction (clicks, forms, animations, etc.)
- Responds to changes instantly, no page reload required
- Essential for modern, responsive, and engaging web experiences

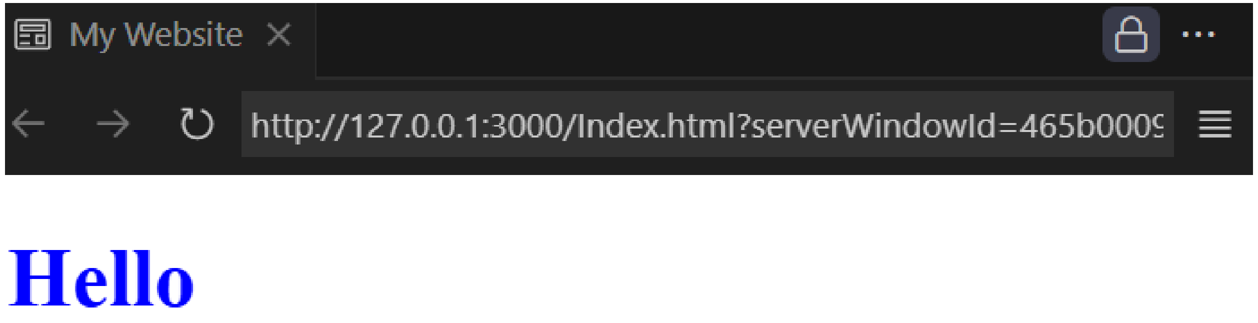
Setting Up Our Project

1. Create a new project folder named DOM
2. Inside it, add three files:
    1. index.html → Structure of the page
    2. styles.css → Styling and layout
    3. index.js → JavaScript functionality
3. Open the folder in your code editor (e.g., VS Code) to begin development

Basic HTML structure with CSS

```
<!DOCTYPE html>
<html>
    <head>
        <title>My Website</title>
        <link rel='stylesheet' href='styles.css'>
    </head>

    <body>
        <h1>Hello</h1>
    </body>
</html>
```

```
h1{ color: blue;}
```



http://127.0.0.1:3000/Index.html?serverWindowId=465b0009

Hello

Ways to Add JavaScript

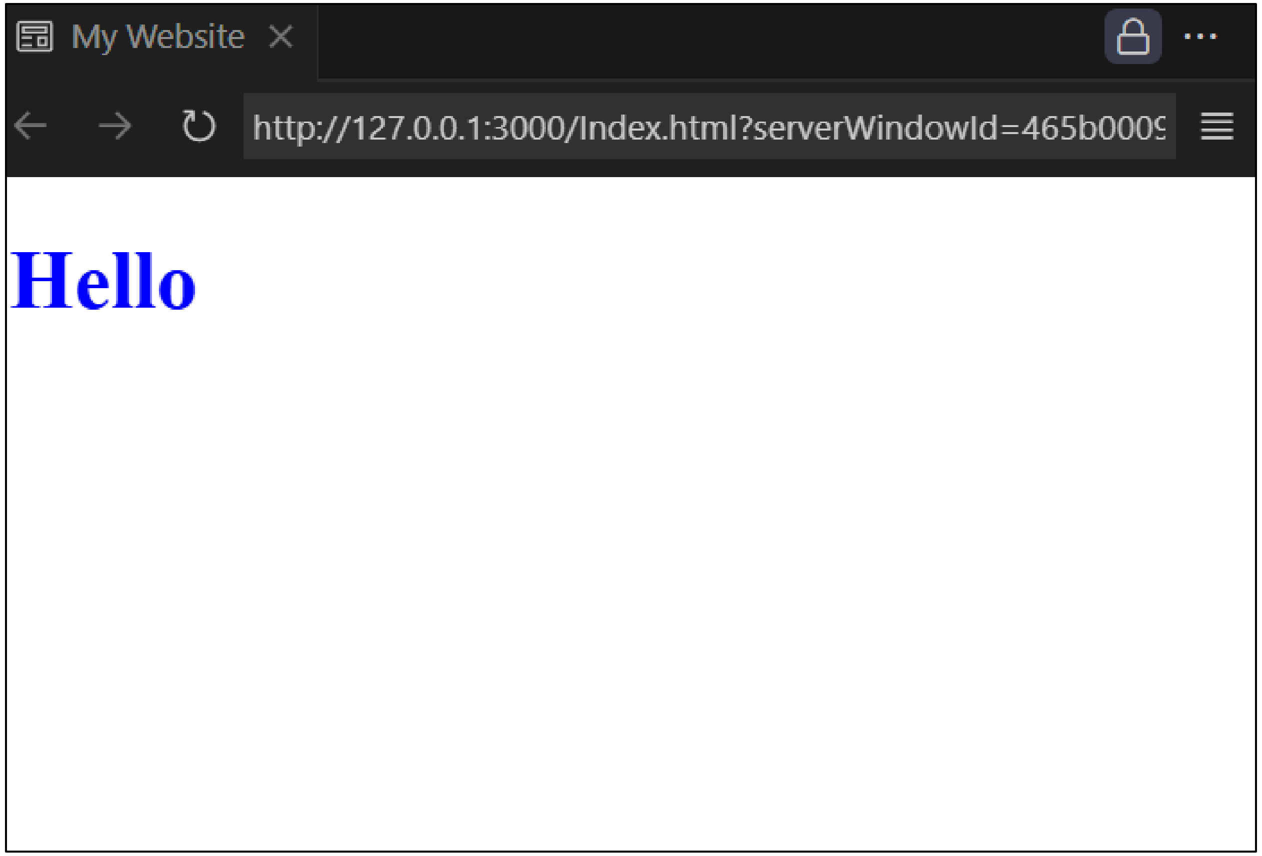
1. Inline JavaScript
2. Internal JavaScript
3. External JavaScript

These are like CSS styling methods.

Inline JavaScript

```
<!DOCTYPE html>
<html>
    <head>
        <title>My Website</title>
        <link rel='stylesheet' href='styles.css'>
    </head>

    <body onload="alert(“Hello”);">
     <h1>Hello</h1>
    </body>
</html>
```

```
h1{ color: blue;}
```

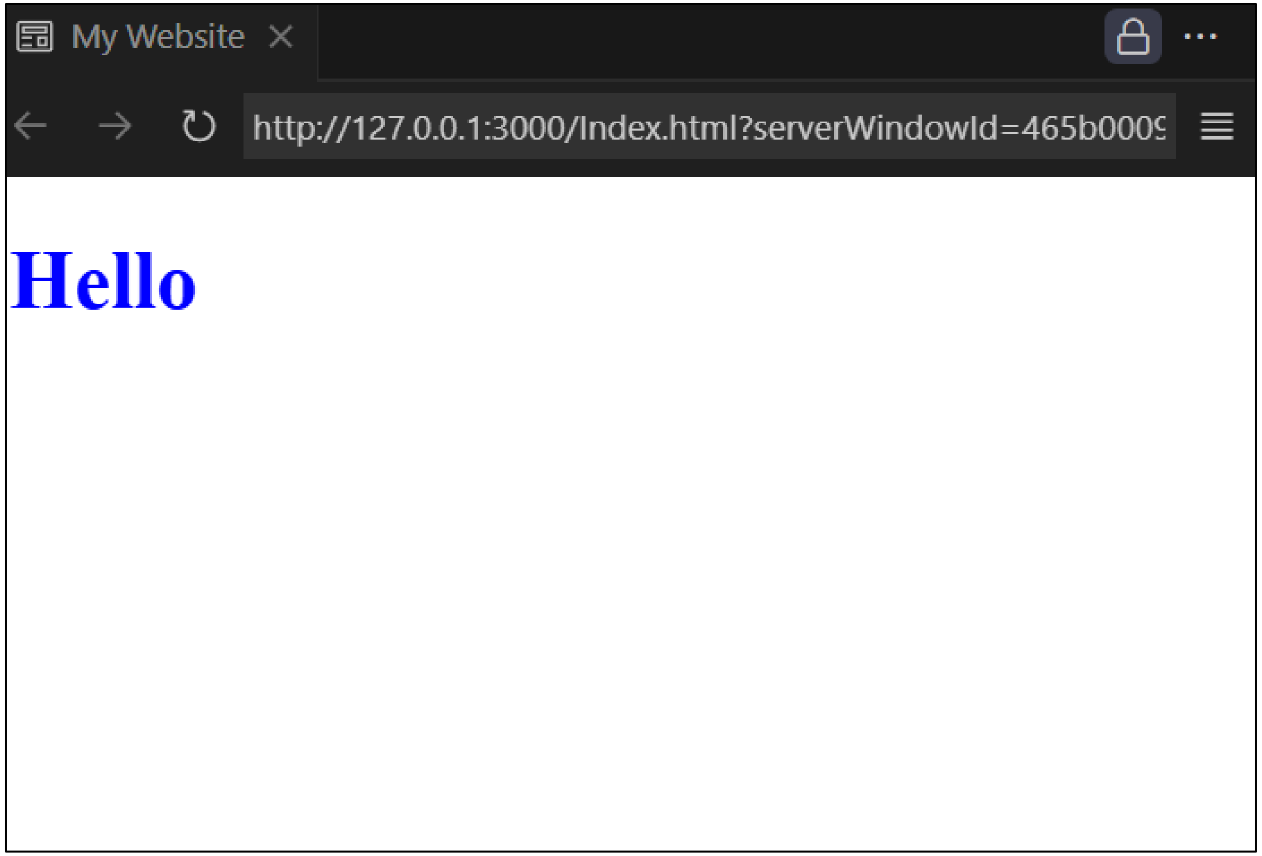


Hello

Inline JavaScript

```
<!DOCTYPE html>
<html>
    <head>
        <title>My Website</title>
        <link rel='stylesheet' href='styles.css'>
    </head>

    <body onload="alert(“Hello”);">
     <h1>Hello</h1>
    </body>
</html>
```

```
h1{ color: blue;}
```

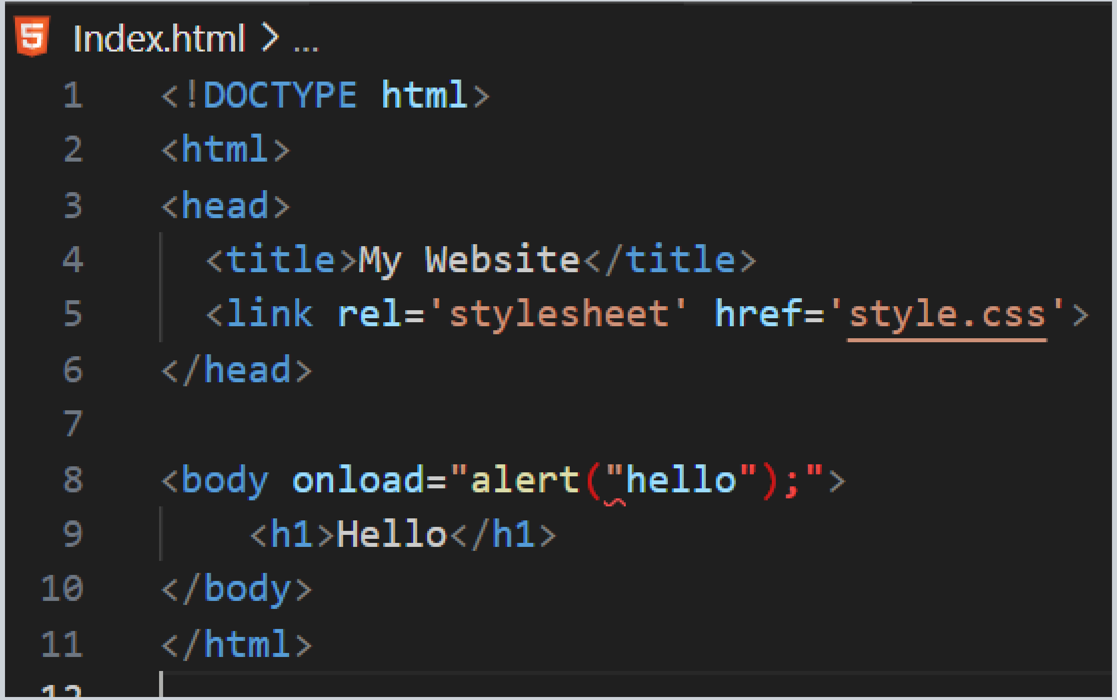


Hello

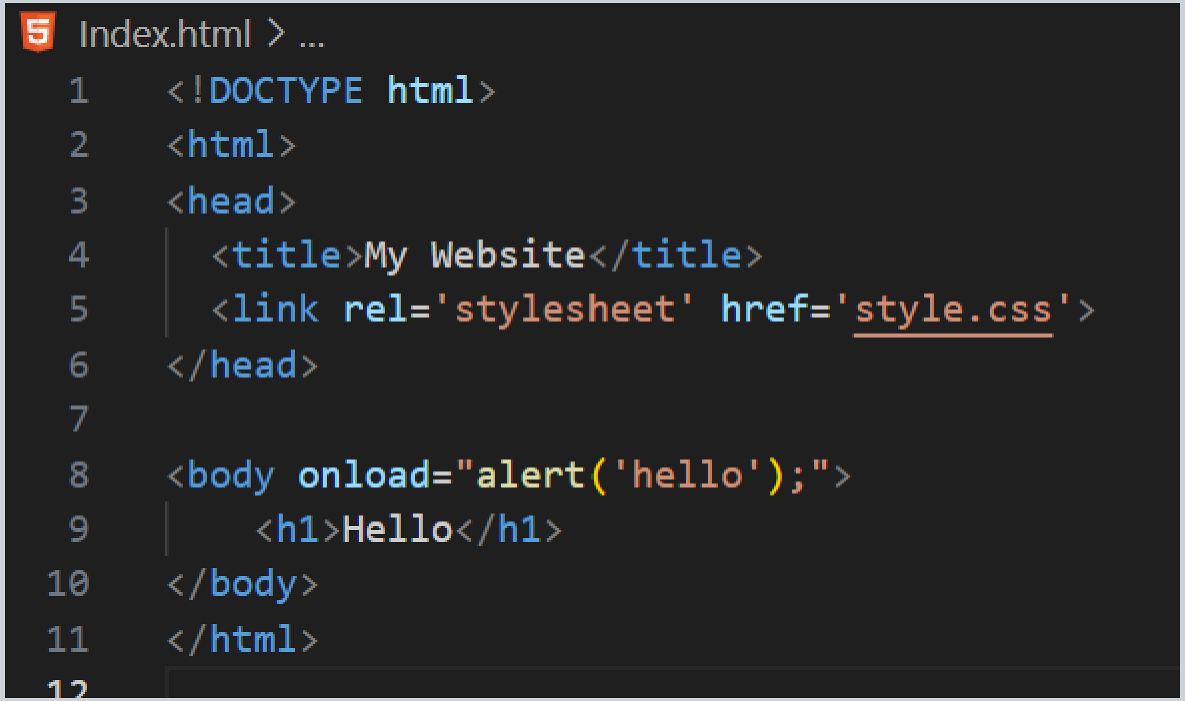
Something is wrong ? No alert?

Quotation Mark Issue

- Problem: Quotes inside quotes cause confusion.
- Use single quotes inside double quotes:
- <body onload="alert('Hello');">



```html
<!DOCTYPE html>
<html>
<head>
  <title>My Website</title>
  <link rel='stylesheet' href='style.css'>
</head>

<body onload="alert("hello");">
    <h1>Hello</h1>
</body>
</html>
```

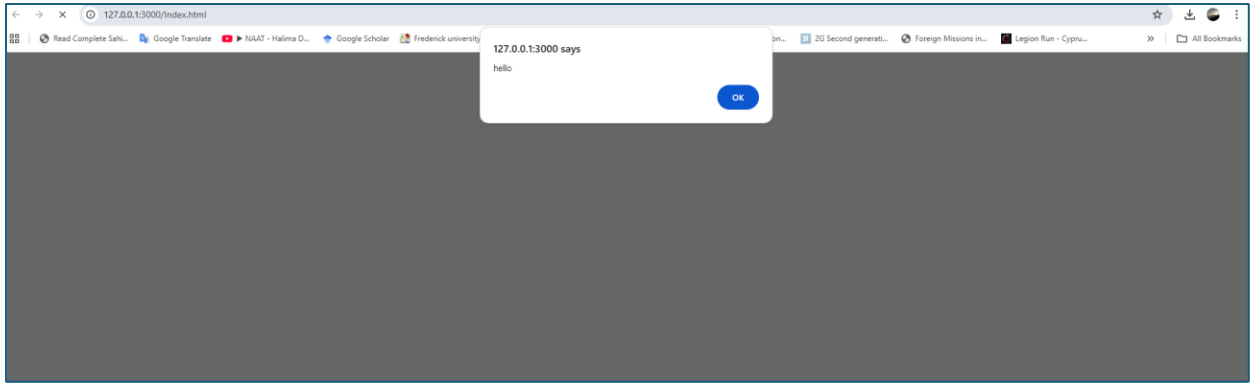


```html
<!DOCTYPE html>
<html>
<head>
  <title>My Website</title>
  <link rel='stylesheet' href='style.css'>
</head>

<body onload="alert('hello');">
    <h1>Hello</h1>
</body>
</html>
```

Inline JavaScript

```
<!DOCTYPE html>
<html>
    <head>
        <title>My Website</title>
        <link rel='stylesheet' href='styles.css'>
    </head>

    <body onload="alert(“Hello”);">
     <h1>Hello</h1>
    </body>
</html>
```

```
h1{ color: blue;}
```



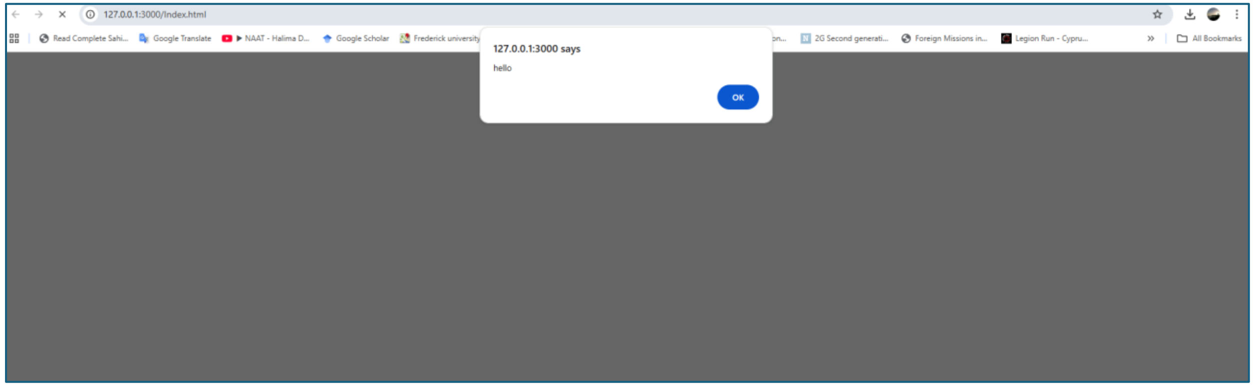
127.0.0.1:3000 says

hello

Inline JavaScript

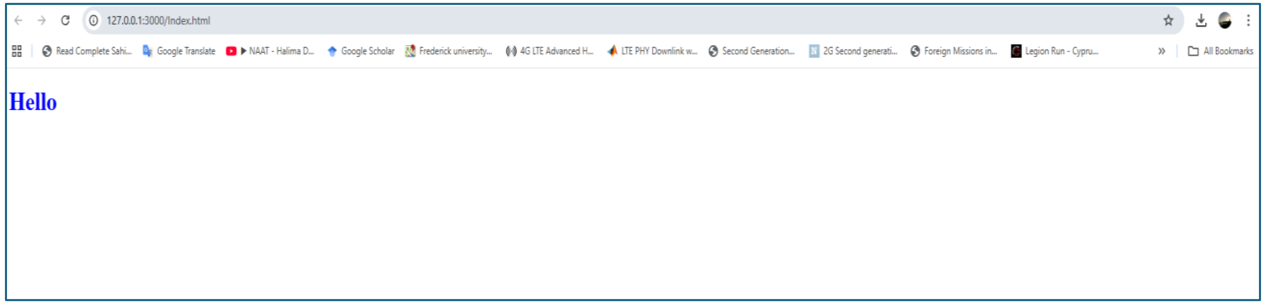
```
<!DOCTYPE html>
<html>
    <head>
        <title>My Website</title>
        <link rel='stylesheet' href='styles.css'>
    </head>

    <body onload="alert(“Hello”);">
     <h1>Hello</h1>
    </body>
</html>
```

```
h1{ color: blue;}
```



127.0.0.1:3000 says

hello



Hello

Inline JS: Drawbacks

Why It’s Not Recommended

- **Not Modular** – code is locked inside one element
- **Hard to Maintain** – must edit every HTML tag manually
- **Poor Reusability** – can’t reuse or scale code
- **Breaks Separation of Concerns** – mixes structure (HTML) and behavior (JS)
- **Difficult to Debug** – error tracking becomes messy

Internal JavaScript

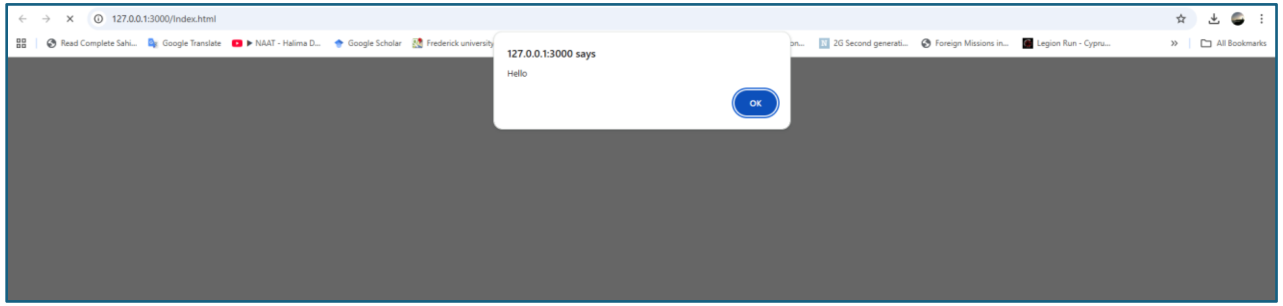
```
<!DOCTYPE html>
<html>
    <head>
        <title>My Website</title>
        <link rel='stylesheet' href='styles.css'>
    </head>

    <body >
        <h1>Hello</h1>
        <script >
        alert("Hello");
        </script>
    </body>
</html>
```

```
h1{ color: blue;}
```

Internal JavaScript

```
<!DOCTYPE html>
<html>
    <head>
        <title>My Website</title>
        <link rel='stylesheet' href='styles.css'>
    </head>

    <body >
        <h1>Hello</h1>
        <script >
        alert("Hello");
        </script>
    </body>
</html>
```

```
h1{ color: blue;}
```



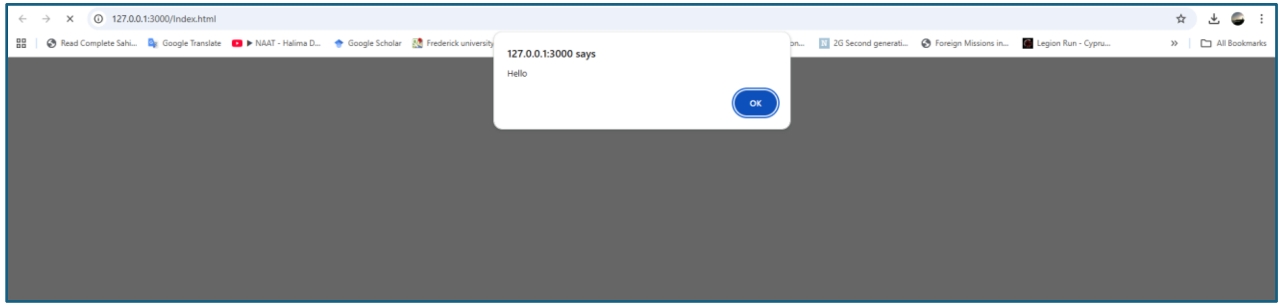
127.0.0.1:3000 says

Hello

Internal JavaScript

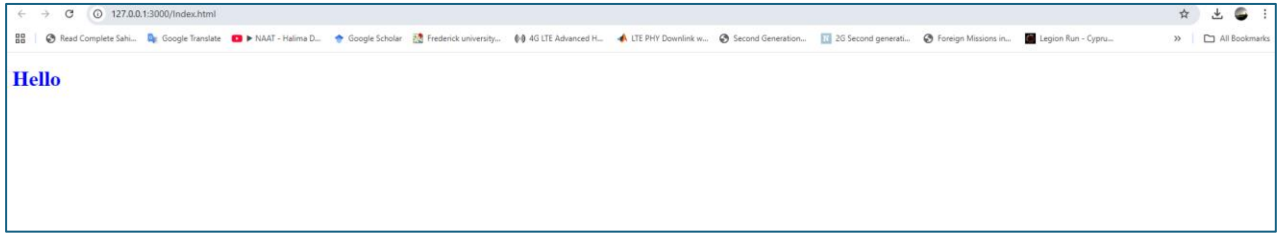
```
<!DOCTYPE html>
<html>
    <head>
        <title>My Website</title>
        <link rel='stylesheet' href='styles.css'>
    </head>

    <body >
        <h1>Hello</h1>
        <script >
        alert("Hello");
        </script>
    </body>
</html>
```

```
h1{ color: blue;}
```



127.0.0.1:3000 says

Hello

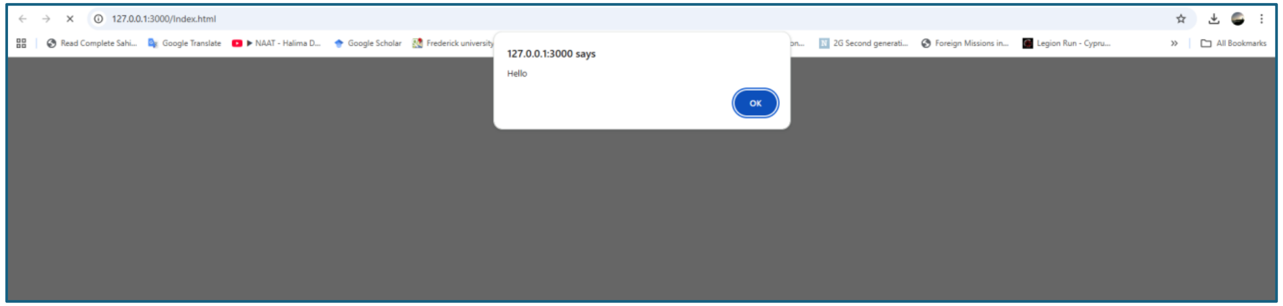


Hello

Internal JavaScript

```
<!DOCTYPE html>
<html>
    <head>
        <title>My Website</title>
        <link rel='stylesheet' href='styles.css'>
    </head>

    <body >
        <h1>Hello</h1>
        <script >
        alert("Hello");
        </script>
    </body>
</html>
```

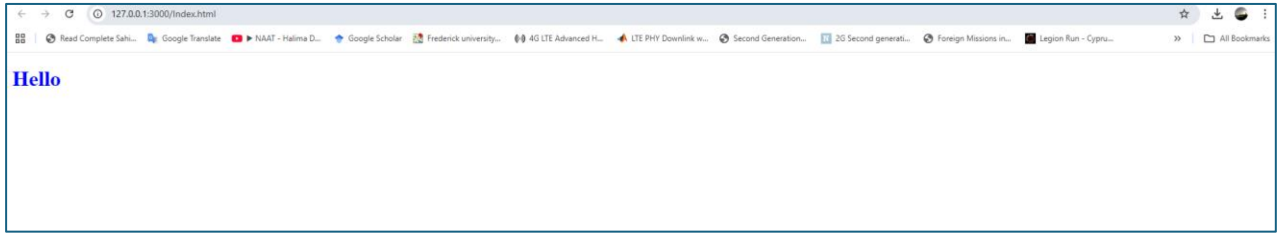
```
h1{ color: blue;}
```



127.0.0.1:3000 says

Hello

OK



Hello

- Browser reads top to bottom.
- Executes <script> when found.
- Then continues rest of page.

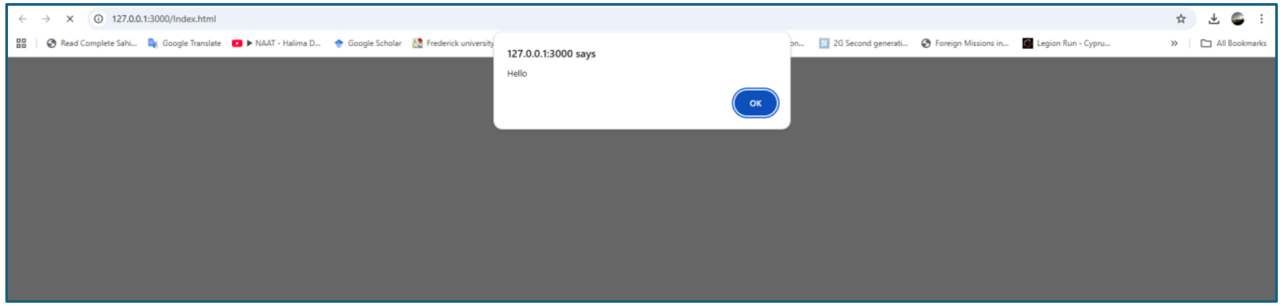
Something is wrong ? Can you guess?

External JavaScript

```
<!DOCTYPE html>
<html>
     <head>
           <title>My Website</title>
           <link rel='stylesheet' href='styles.css'>
     </head>

     <body onload="alert(“Hello”);">
          <h1>Hello</h1>

          <script src="index.js"></script></body>
</html>
```

```
h1{ color: blue;}
```

```
alert("Hello");
```



127.0.0.1:3000 says

Hello

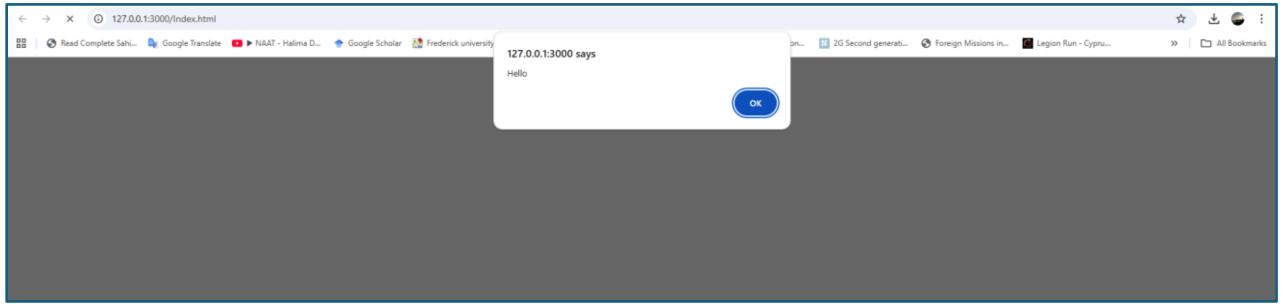
JavaScript is written in separate file index.js

External JavaScript

```
<!DOCTYPE html>
<html>
    <head>
        <title>My Website</title>
        <link rel='stylesheet' href='styles.css'>
    </head>

    <body onload="alert(“Hello”);">
        <h1>Hello</h1>

        <script src="index.js"></script></body>
</html>
```

```
h1{ color: blue;}
```

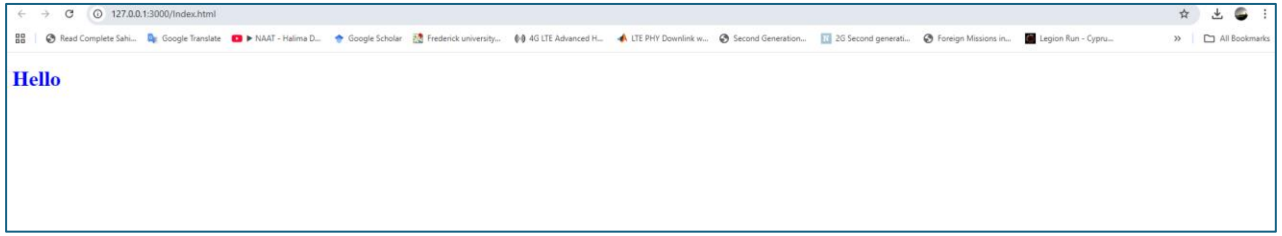
```
alert("Hello");
```



127.0.0.1:3000/index.html

127.0.0.1:3000 says

Hello



127.0.0.1:3000/index.html

Hello

JavaScript is written in separate file index.js

Something is wrong ? Can you guess?

Why Alert Appears Before HTML Rendering

Root Cause:

- JavaScript executes very quickly while the browser is still rendering the page.
- The alert() function is blocking, meaning it pauses rendering until the user closes it.
- The <h1> is already parsed in memory but hasn’t been painted to the screen yet.

Result:

The alert interrupts the browser before the visual paint, so you don’t see the heading.

Alert Appears Before HTML Rendering:- How to Fix It

Possible Solutions:

Solution	Pros	Cons
1. Use defer	Modern & clean and ensures DOM ready before JS runs	Only for external scripts
2. Move <script> to bottom	Simple & effective and works without extra attributes	On very small pages, alert may still block before paint
3. Use setTimeout()	Forces render before alert Works everywhere	Adds artificial delay logic
4. Use DOMContentLoaded event	Clean JS-side control Works	Looks bit complex for beginners
5. Use window.onload	Waits until full page (images, CSS) loads	Slowest i-e it runs after all resources

External JavaScript

```
<!DOCTYPE html>
<html>
        <head>
                 <title>My Website</title>
                 <link rel='stylesheet' href='styles.css'>
        </head>

        <body onload="alert(“Hello”);">
                <h1>Hello</h1>

                <script src="index.js"></script></body>
</html>
```

```
h1{ color: blue;}
```

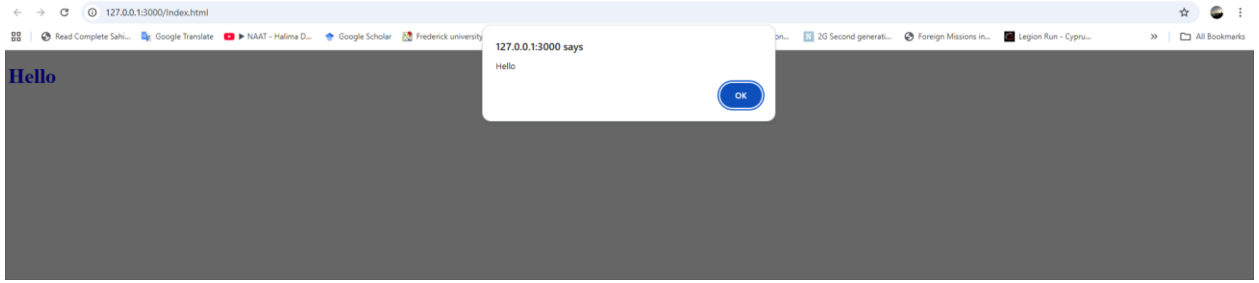
```
setTimeout(() => {
   alert("Hello");
}, 100);
```

newer syntax introduced in ES6 (modern JavaScript).

```
setTimeout(function() {
  alert("Hello!");
}, 10
```

External JavaScript

```
<!DOCTYPE html>
<html>
        <head>
                 <title>My Website</title>
                 <link rel='stylesheet' href='styles.css'>
        </head>

        <body onload="alert(“Hello”);">
                <h1>Hello</h1>

                <script src="index.js"></script></body>
</html>
```



Hello

127.0.0.1:3000 says

Hello

```
h1{ color: blue;}
```

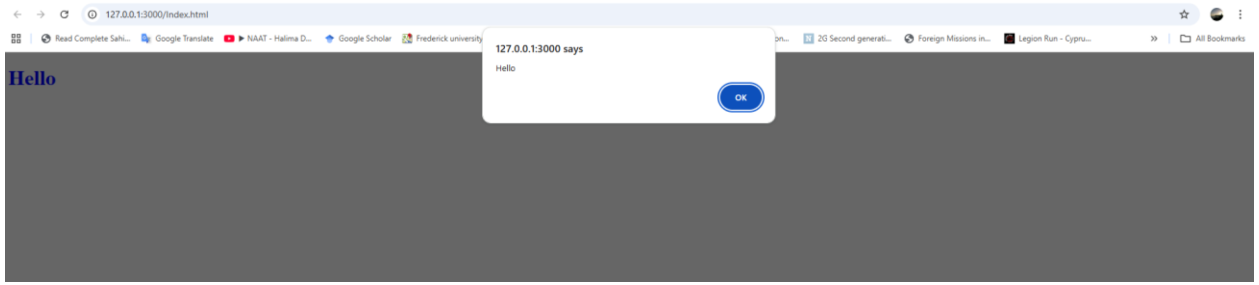
```
setTimeout(() => {
   alert("Hello");
}, 100);
```

```
setTimeout(function() {
 alert("Hello!");
}, 10
```

newer syntax introduced in ES6 (modern JavaScript).

External JavaScript

```
<!DOCTYPE html>
<html>
        <head>
                 <title>My Website</title>
                 <link rel='stylesheet' href='styles.css'>
        </head>

        <body onload="alert("Hello");">
                <h1>Hello</h1>

                <script src="index.js"></script></body>
</html>
```

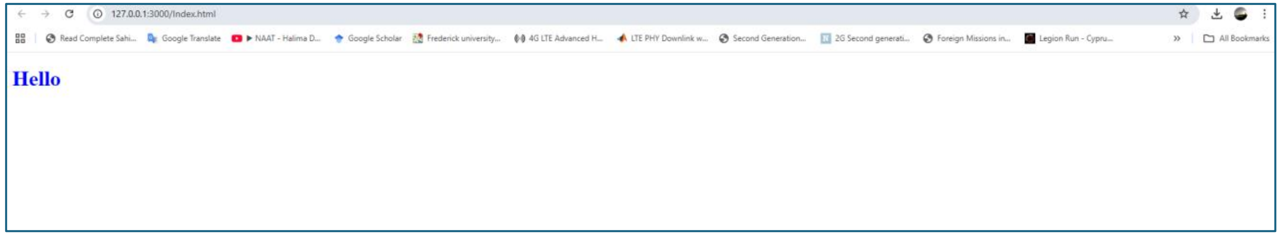


127.0.0.1:3000/Index.html

Hello

127.0.0.1:3000 says

Hello



127.0.0.1:3000/Index.html

Hello

```
h1{ color: blue;}
```

```
setTimeout(function() {
  alert("Hello!");
}, 10
```

```
setTimeout(() => {
    alert("Hello");
}, 100);
```

newer syntax introduced in ES6 (modern JavaScript).

How Arrow Functions Simplify JavaScript (ES6 Feature)

Traditional Function	Arrow Function	Meaning
function() { … }	() => { … }	Same behavior, just shorter
function(name) { … }	(name) => { … }	One parameter
function(a,b) { … }	(a,b) => { … }	Multiple parameters
function() { return 5; }	() => 5	Short return form

Browser Execution Order

Conceptual

- **HTML → Structure:** builds the DOM (page elements)
- **CSS → Style: decorates** and lays out elements
- **JavaScript → Behavior:** adds interactivity and changes DOM

Real parsing note:

Browsers parse HTML and may fetch/apply CSS while scripts run. If a script is encountered early, JS can run *before* the page is painted.

Wrong placement leads towards missing elements

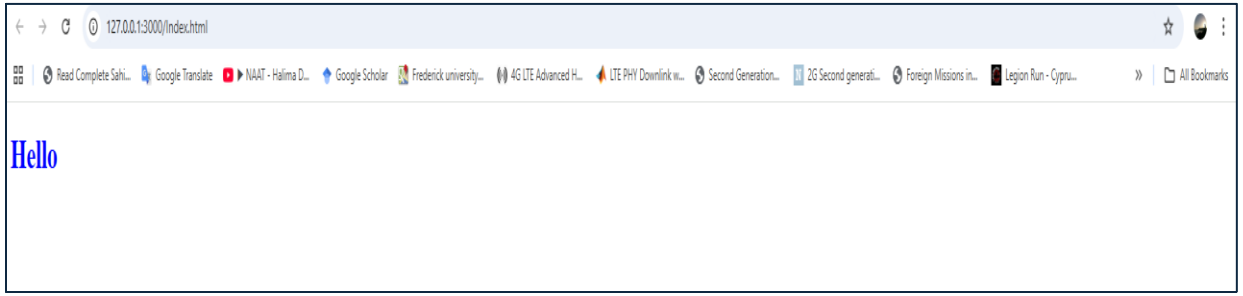
- If JS runs before the HTML element exists, it can’t find it
  - It results in null, errors, or UI not visible.

External JavaScript

```
<!DOCTYPE html>
<html>
        <head>
                <title>My Website</title>
                <link rel='stylesheet' href='style.css'>
        </head>
        <body >
                <h1>Hello</h1>
                <script src="index.js"></script>
        </body>
</html>
```

```
h1{ color: blue;}
```

```
setTimeout(() => {
 document.querySelector("h1").innerHTML="Good Bye";
}, 1000);
```



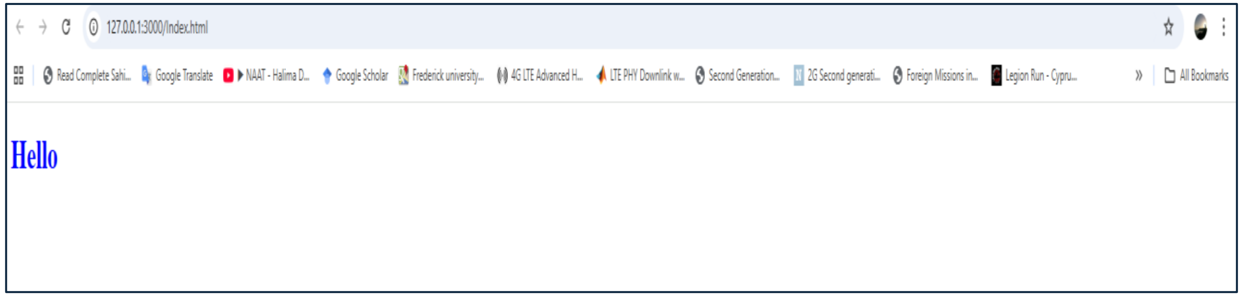
Hello

External JavaScript

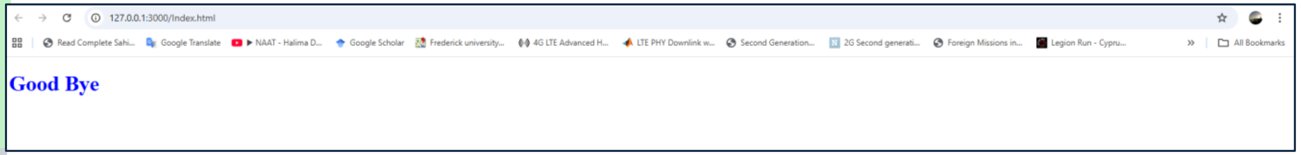
```
<!DOCTYPE html>
<html>
        <head>
                 <title>My Website</title>
                 <link rel='stylesheet' href='style.css'>
        </head>
        <body >
               <h1>Hello</h1>
               <script src="index.js"></script>
        </body>
</html>
```

```
h1{ color: blue;}
```

```
setTimeout(() => {
 document.querySelector("h1").innerHTML="Good Bye";
}, 1000);
```



Hello



Good Bye

Static Websites

- HTML + CSS only define structure and style
- After saving and refreshing, website is fixed

Issues:

- Can’t respond to user actions like clicks or input
- Need manual editing to change content

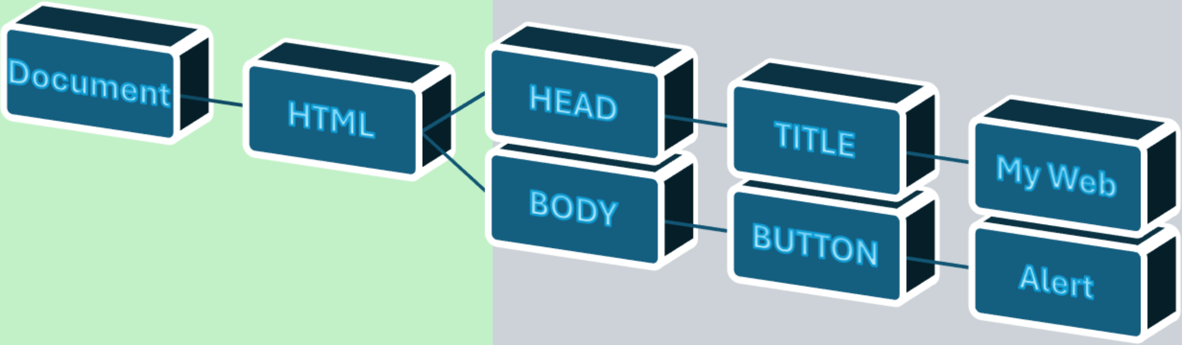
Goal:

We need a way to make pages **change on the fly, without reloading or editing HTML manually.**

DOM (Document Object Model)

- Converts webpage into a structured collection of objects
- Each HTML element becomes an object JS can select & manipulate
- It’s the bridge between HTML/CSS and JavaScript.
- Each element is stored in memory as object.

```
<html>
  <head>
    <title> My Website</title>
  </head>
  <body>
    <body>
      <button onclick="alert('Output');">Click here</button>
    </body>
  </body>
</html>
```



Document

HTML

HEAD

BODY

TITLE

BUTTON

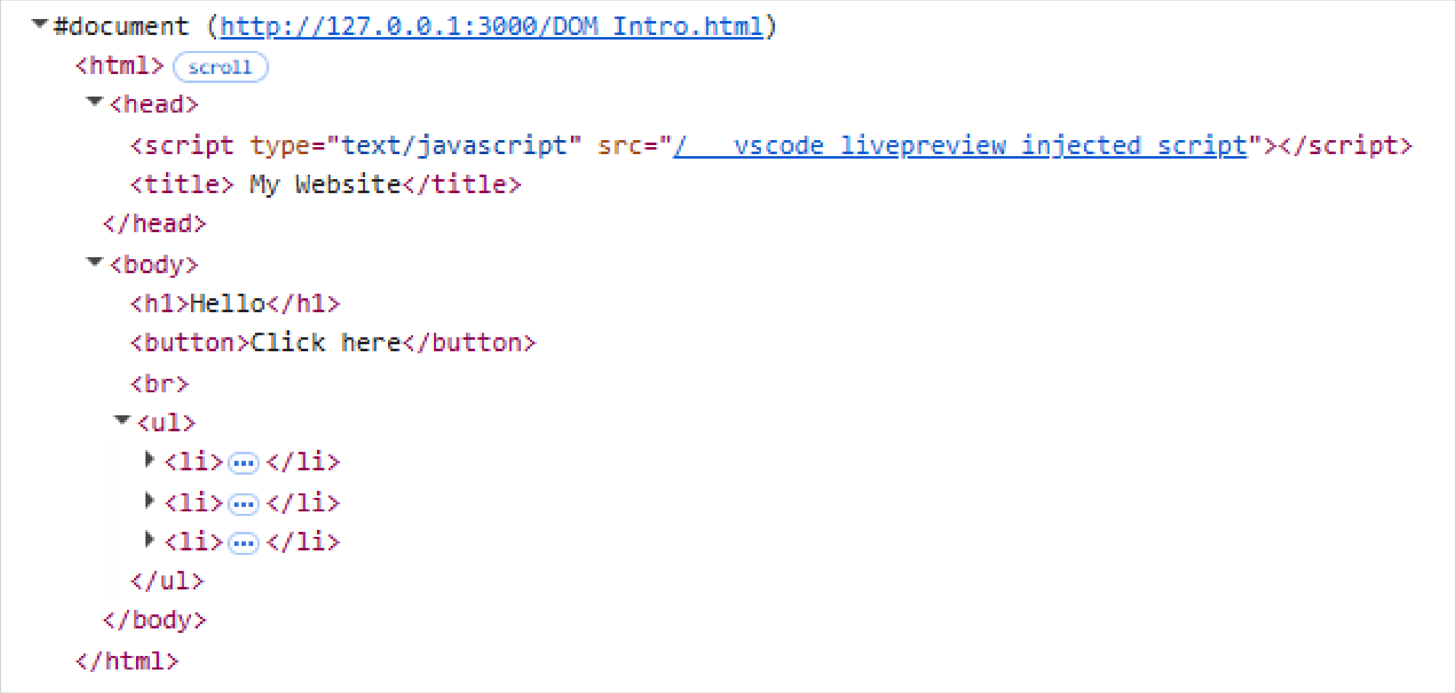
My Web

Alert

Viewing the DOM

```
<html>
  <head>
    <title> My Website</title>
  </head>
  <body>
    <h1>Hello</h1>
    <button >Click here</button><br>
    <ul>
      <li>Google</li>
      <li>COMSATS</li>
      <li>Comp: Eng:</li>
    </ul>
  </body>
</html>
```

Document;

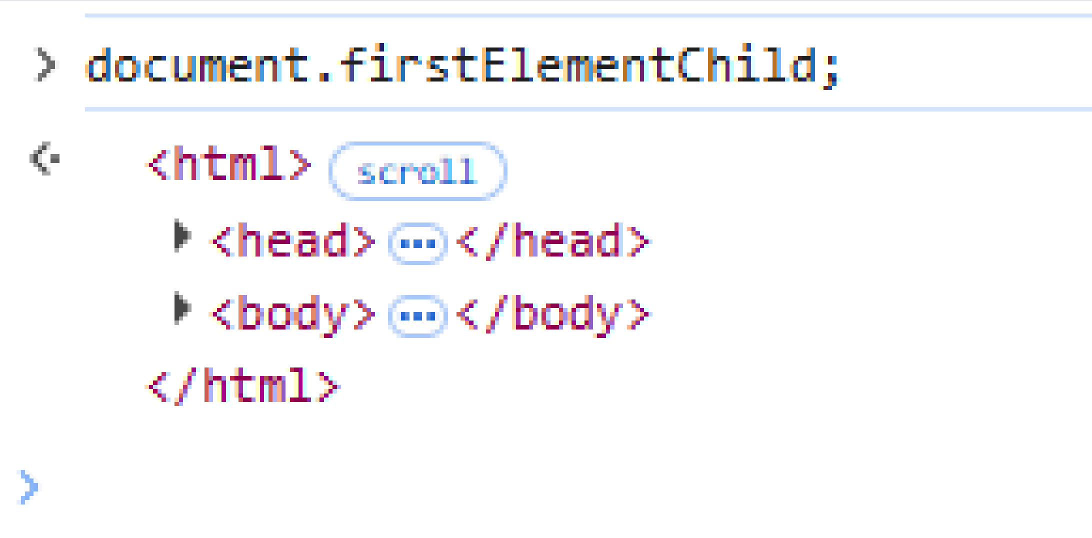


```
▾#document (http://127.0.0.1:3000/DOM_Intro.html)
   <html> scroll
    ▾<head>
       <script type="text/javascript" src="/__vscode_livepreview_injected_script"></script>
       <title> My Website</title>
     </head>
    ▾<body>
       <h1>Hello</h1>
       <button>Click here</button>
       <br>
      ▾<ul>
        ▸<li>…</li>
        ▸<li>…</li>
        ▸<li>…</li>
       </ul>
     </body>
   </html>
```

Viewing the DOM

```
<html>
  <head>
    <title> My Website</title>
  </head>
  <body>
    <h1>Hello</h1>
    <button >Click here</button><br>
    <ul>
      <li>Google</li>
      <li>COMSATS</li>
      <li>Comp: Eng:</li>
    </ul>
  </body>
</html>
```

```
document.firstElementChild;
```

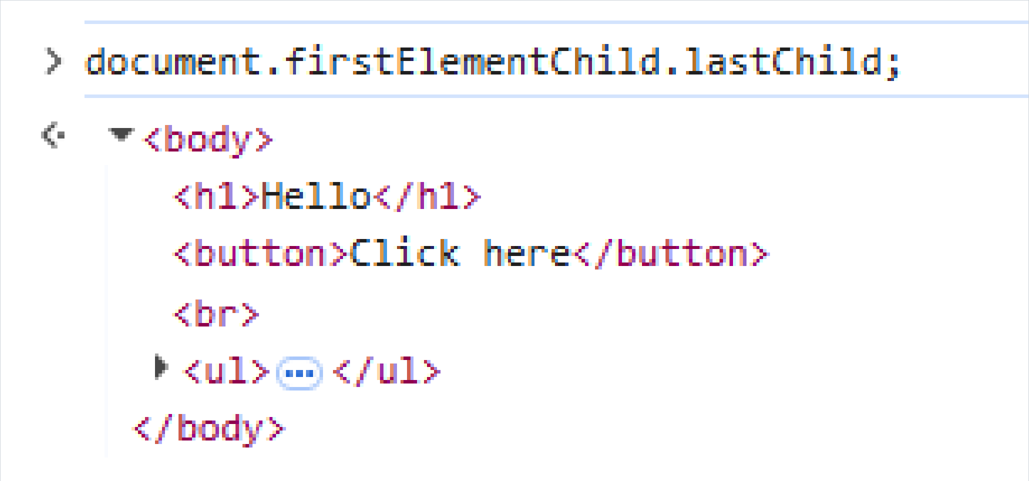


```
> document.firstElementChild;
<·   <html> scroll
       ▶ <head> ··· </head>
       ▶ <body> ··· </body>
     </html>
>
```

Viewing the DOM

```
<html>
  <head>
    <title> My Website</title>
  </head>
  <body>
    <h1>Hello</h1>
    <button >Click here</button><br>
    <ul>
      <li>Google</li>
      <li>COMSATS</li>
      <li>Comp: Eng:</li>
    </ul>
  </body>
</html>
```

```
document.firstElementChild.lastChild;
```

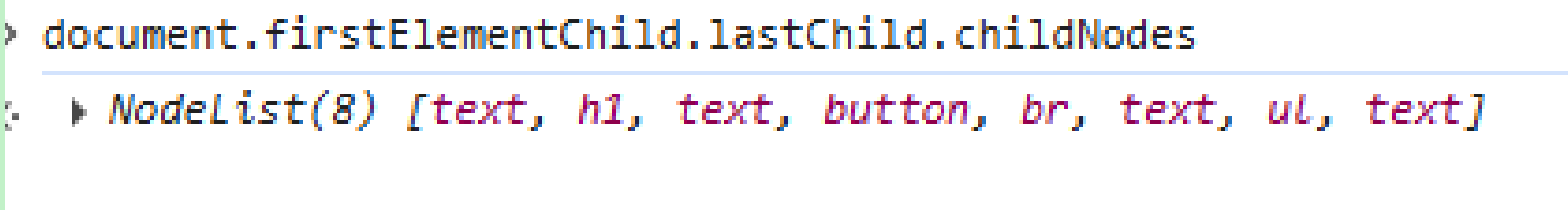


```
> document.firstElementChild.lastChild;
<· ▾<body>
      <h1>Hello</h1>
      <button>Click here</button>
      <br>
     ▸<ul>…</ul>
    </body>
```

Viewing the DOM

```
<html>
  <head>
    <title> My Website</title>
  </head>
  <body>
    <h1>Hello</h1>
    <button >Click here</button><br>
    <ul>
      <li>Google</li>
      <li>COMSATS</li>
      <li>Comp: Eng:</li>
    </ul>
  </body>
</html>
```

```
document.firstElementChild.lastChild.childNodes;
```

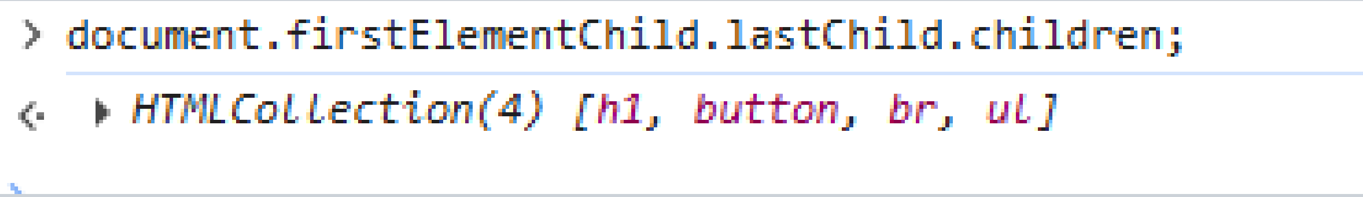


```
> document.firstElementChild.lastChild.childNodes
<· ▸ NodeList(8) [text, h1, text, button, br, text, ul, text]
```

Viewing the DOM

```
<html>
  <head>
    <title> My Website</title>
  </head>
  <body>
    <h1>Hello</h1>
    <button >Click here</button><br>
    <ul>
      <li>Google</li>
      <li>COMSATS</li>
      <li>Comp: Eng:</li>
    </ul>
  </body>
</html>
```

```
document.firstElementChild.lastChild.children;
```

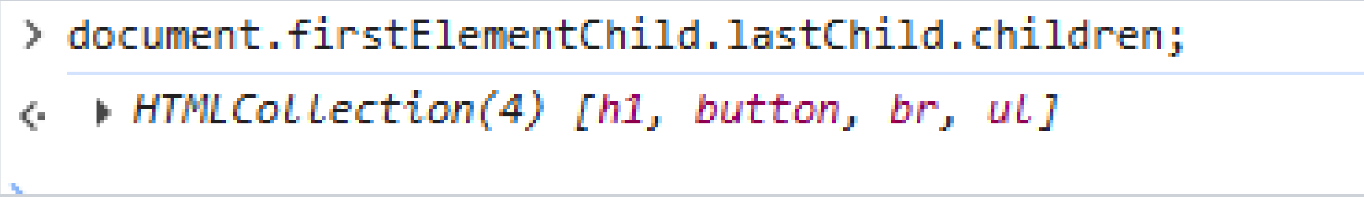


```
> document.firstElementChild.lastChild.children;
<·  ▸ HTMLCollection(4) [h1, button, br, ul]
```

Viewing the DOM

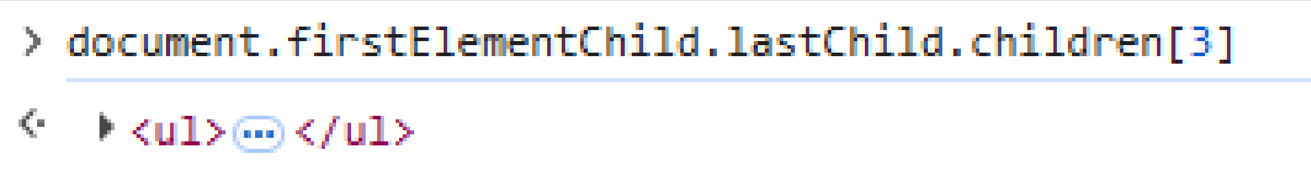
```
<html>
  <head>
    <title> My Website</title>
  </head>
  <body>
    <h1>Hello</h1>
    <button >Click here</button><br>
    <ul>
      <li>Google</li>
      <li>COMSATS</li>
      <li>Comp: Eng:</li>
    </ul>
  </body>
</html>
```

```
document.firstElementChild.lastChild.children;
```



```
> document.firstElementChild.lastChild.children;
<·  ▸ HTMLCollection(4) [h1, button, br, ul]
```

```
document.firstElementChild.lastChild.children[3];
```

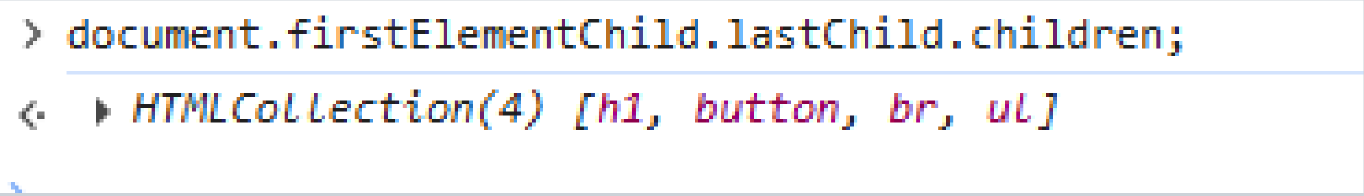


```
> document.firstElementChild.lastChild.children[3]
<·  ▸ <ul>…</ul>
```

Viewing the DOM

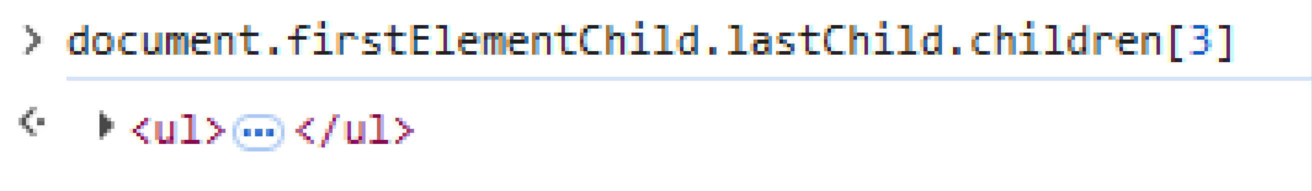
```
<html>
  <head>
    <title> My Website</title>
  </head>
  <body>
    <h1>Hello</h1>
    <button >Click here</button><br>
    <ul>
      <li>Google</li>
      <li>COMSATS</li>
      <li>Comp: Eng:</li>
    </ul>
  </body>
</html>
```

document.firstElementChild.lastChild.children;



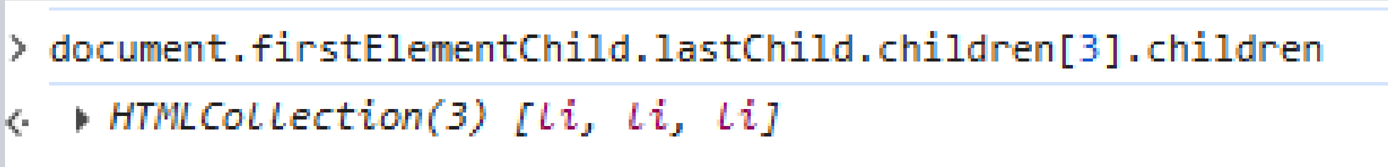
```
> document.firstElementChild.lastChild.children;
<· ▸ HTMLCollection(4) [h1, button, br, ul]
```

document.firstElementChild.lastChild.children[3];



```
> document.firstElementChild.lastChild.children[3]
<· ▸ <ul> … </ul>
```

document.firstElementChild.lastChild.children[3].children;

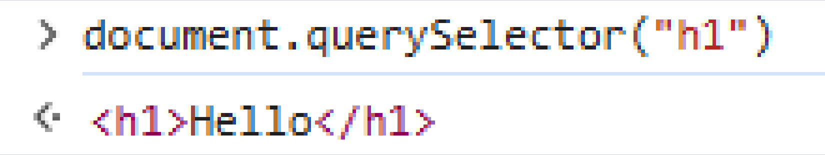


```
> document.firstElementChild.lastChild.children[3].children
<· ▸ HTMLCollection(3) [li, li, li]
```

Selecting Elements using querySelector

```
<html>
  <head>
    <title> My Website</title>
  </head>
  <body>
    <h1>Hello</h1>
    <button >Click here</button><br>
    <ul>
      <li>Google</li>
      <li>COMSATS</li>
      <li>Comp: Eng:</li>
    </ul>
  </body>
</html>
```

```
document.querySelector("h1")
```



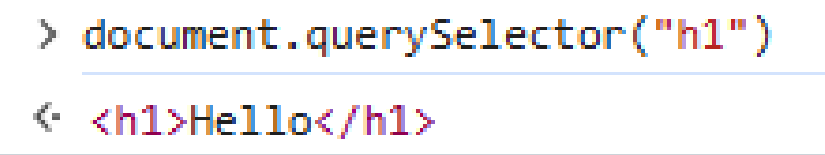
```
> document.querySelector("h1")
< <h1>Hello</h1>
```

- Value in parathesis is string
  - So “” has to be there.

Selecting Elements using querySelector

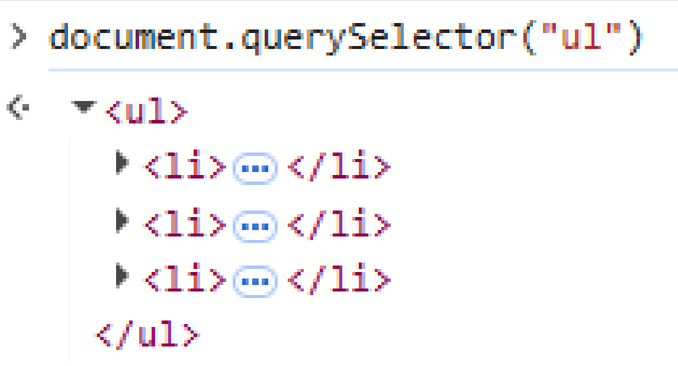
```
<html>
  <head>
    <title> My Website</title>
  </head>
  <body>
    <h1>Hello</h1>
    <button >Click here</button><br>
    <ul>
      <li>Google</li>
      <li>COMSATS</li>
      <li>Comp: Eng:</li>
    </ul>
  </body>
</html>
```

document.querySelector("h1")



```
> document.querySelector("h1")
<· <h1>Hello</h1>
```

document.querySelector(“ul")



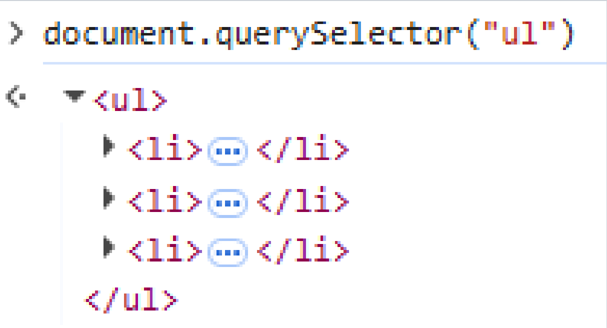
```
> document.querySelector("ul")
<·  ▼<ul>
      ▶<li>…</li>
      ▶<li>…</li>
      ▶<li>…</li>
    </ul>
```

- Value in parathesis is string
  - So “” has to be there.

Selecting Elements using querySelector

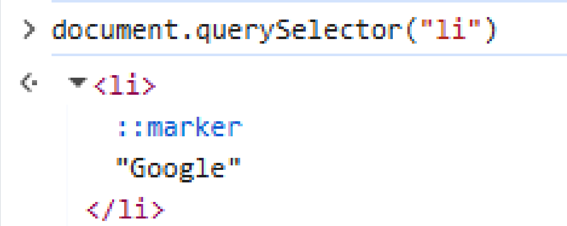
```
<html>
  <head>
    <title> My Website</title>
  </head>
  <body>
    <h1>Hello</h1>
    <button >Click here</button><br>
    <ul>
      <li>Google</li>
      <li>COMSATS</li>
      <li>Comp: Eng:</li>
    </ul>
  </body>
</html>
```

document.querySelector(“ul")



```
> document.querySelector("ul")
<· ▼<ul>
     ▶<li>…</li>
     ▶<li>…</li>
     ▶<li>…</li>
   </ul>
```

document.querySelector(“li")

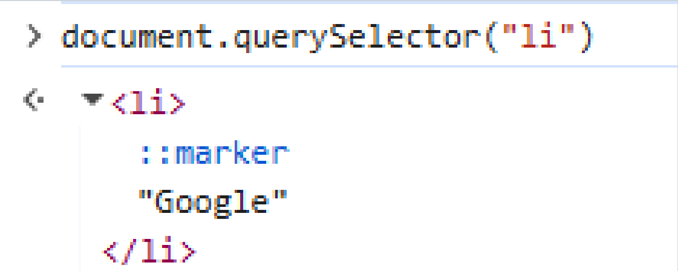


```
> document.querySelector("li")
<· ▼<li>
      ::marker
      "Google"
    </li>
```

Selecting Elements using querySelector

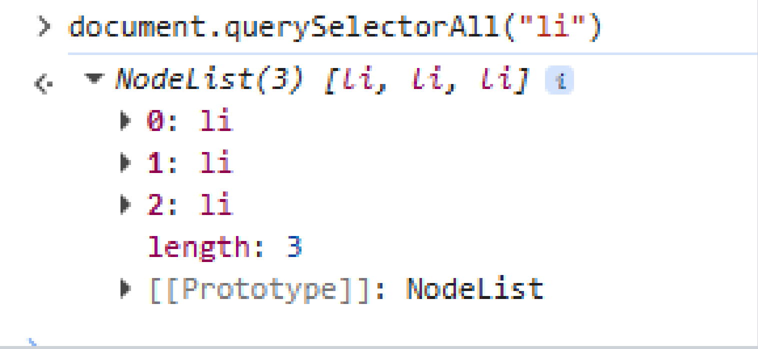
```
<html>
  <head>
    <title> My Website</title>
  </head>
  <body>
    <h1>Hello</h1>
    <button >Click here</button><br>
    <ul>
      <li>Google</li>
      <li>COMSATS</li>
      <li>Comp: Eng:</li>
    </ul>
  </body>
</html>
```

document.querySelector(“li")



```
> document.querySelector("li")
<· ▾<li>
      ::marker
      "Google"
    </li>
```

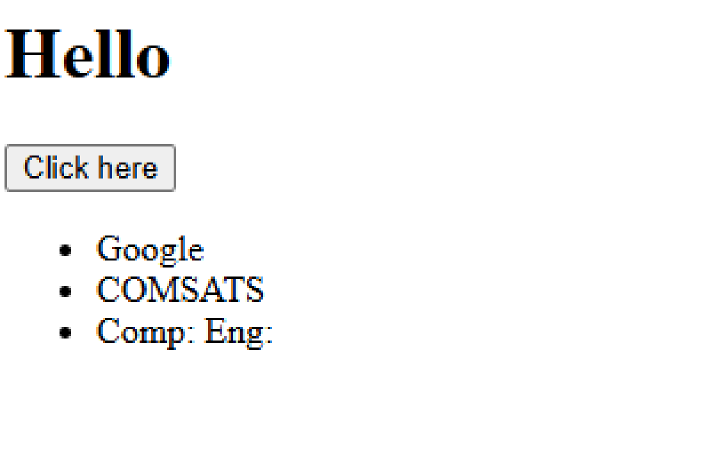
document.querySelectorAll(“li")



```
> document.querySelectorAll("li")
<· ▾NodeList(3) [li, li, li] i
     ▸0: li
     ▸1: li
     ▸2: li
      length: 3
     ▸[[Prototype]]: NodeList
```

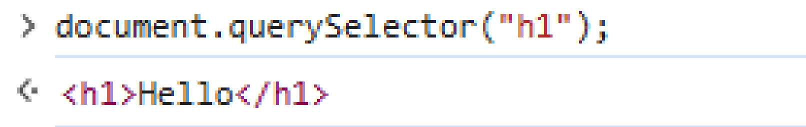
Manipulating Elements .innerHTML Property

```
<html>
  <head>
    <title> My Website</title>
  </head>
  <body>
    <h1>Hello</h1>
    <button >Click here</button><br>
    <ul>
      <li>Google</li>
      <li>COMSATS</li>
      <li>Comp: Eng:</li>
    </ul>
  </body>
</html>
```

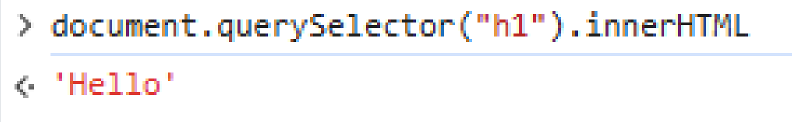


Hello

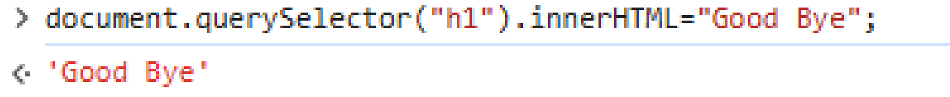
- Google
- COMSATS
- Comp: Eng:



```
> document.querySelector("h1");
< <h1>Hello</h1>
```



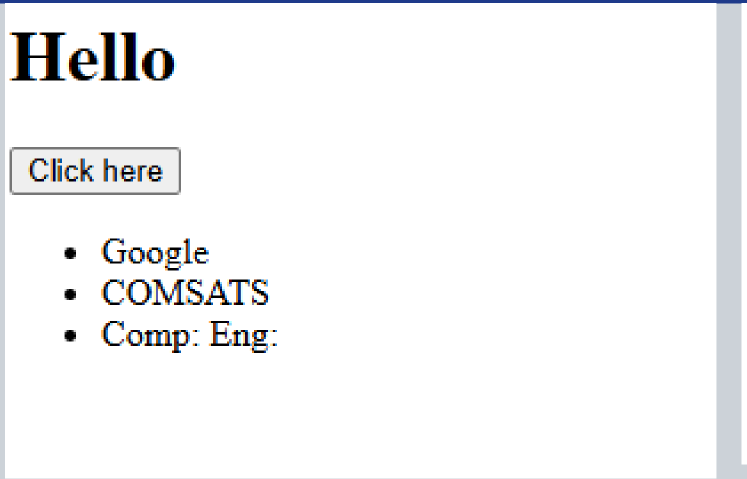
```
> document.querySelector("h1").innerHTML
< 'Hello'
```



```
> document.querySelector("h1").innerHTML="Good Bye";
< 'Good Bye'
```

Manipulating Elements .innerHTML Property

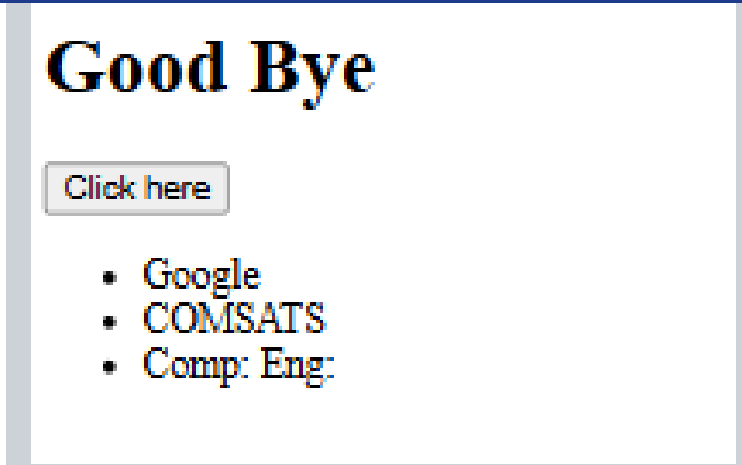
```
<html>
  <head>
    <title> My Website</title>
  </head>
  <body>
    <h1>Hello</h1>
    <button >Click here</button><br>
    <ul>
      <li>Google</li>
      <li>COMSATS</li>
      <li>Comp: Eng:</li>
    </ul>
  </body>
</html>
```



Hello

Click here

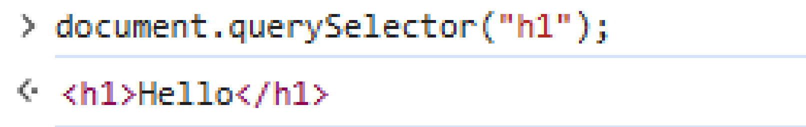
- Google
- COMSATS
- Comp: Eng:



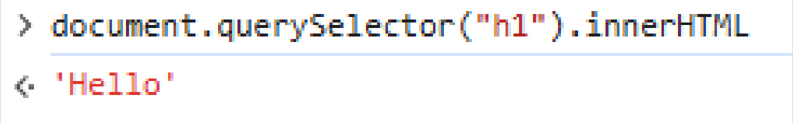
Good Bye

Click here

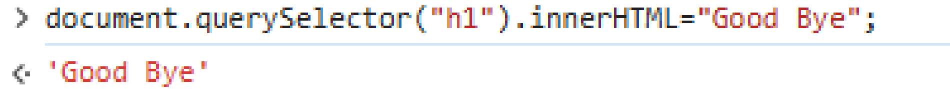
- Google
- COMSATS
- Comp: Eng:



```
> document.querySelector("h1");
<· <h1>Hello</h1>
```



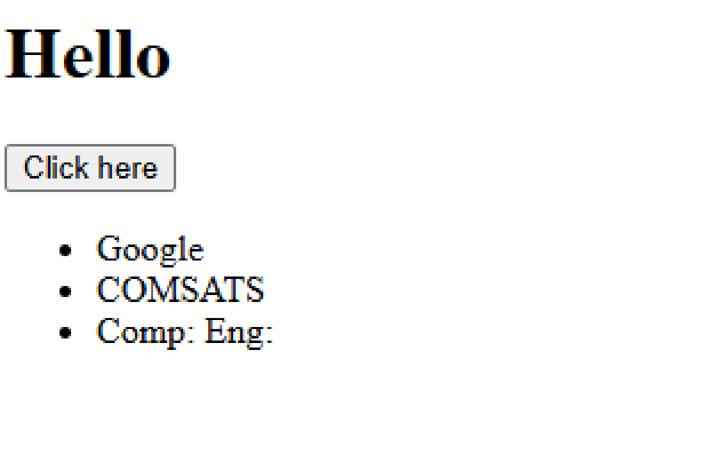
```
> document.querySelector("h1").innerHTML
<· 'Hello'
```



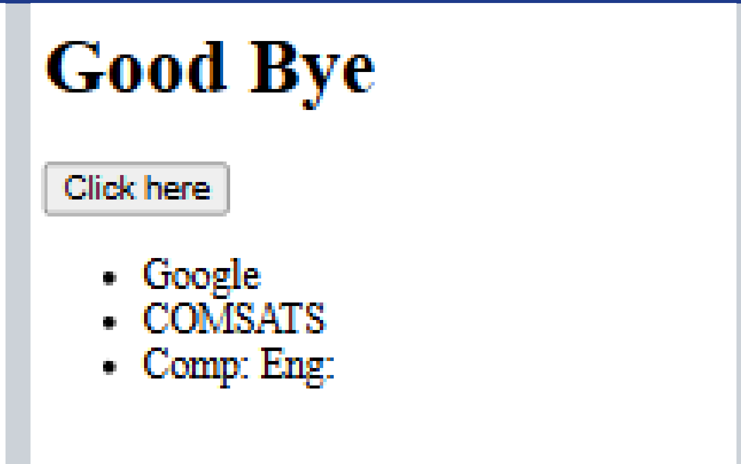
```
> document.querySelector("h1").innerHTML="Good Bye";
<· 'Good Bye'
```

Manipulating Elements .innerHTML Property

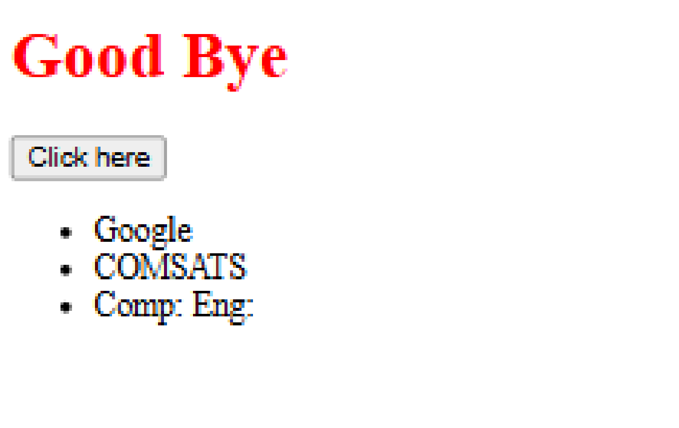
```
<html>
  <head>
    <title> My Website</title>
  </head>
  <body>
    <h1>Hello</h1>
    <button >Click here</button><br>
    <ul>
      <li>Google</li>
      <li>COMSATS</li>
      <li>Comp: Eng:</li>
    </ul>
  </body>
</html>
```


Hello

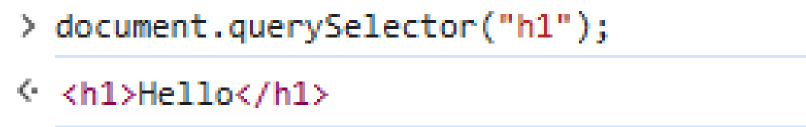
- Google
- COMSATS
- Comp: Eng:

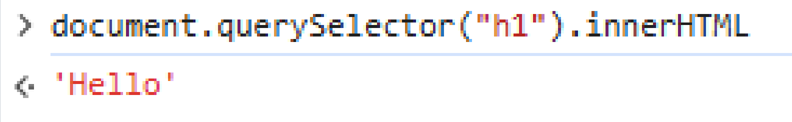

Good Bye

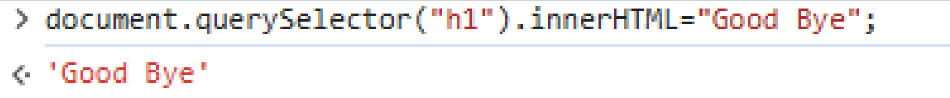
- Google
- COMSATS
- Comp: Eng:

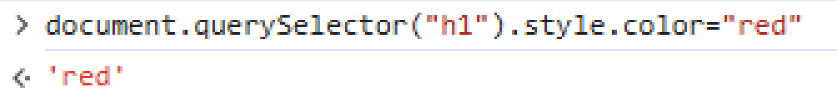

Good Bye

- Google
- COMSATS
- Comp: Eng:


```
> document.querySelector("h1");
<· <h1>Hello</h1>
```


```
> document.querySelector("h1").innerHTML
<· 'Hello'
```


```
> document.querySelector("h1").innerHTML="Good Bye";
<· 'Good Bye'
```


```
> document.querySelector("h1").style.color="red"
<· 'red'
```

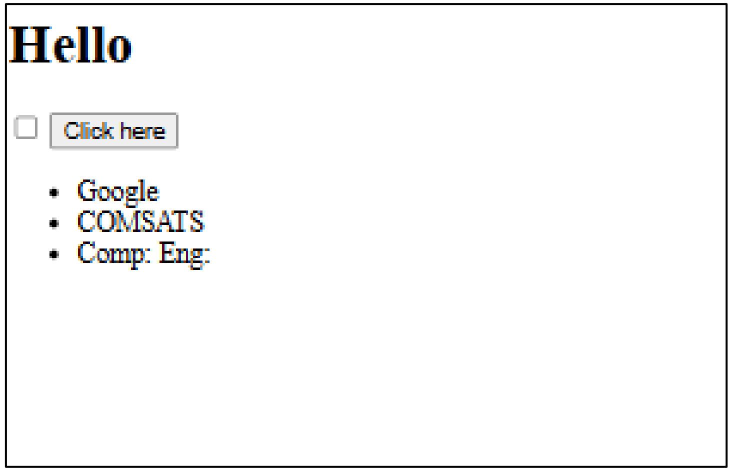
Objects Have Properties & Methods

Properties describe something about the **Objects**

Methods describe the actions that **Objects** can do

Objects Properties

```
<!DOCTYPE html>
<html>
  <head>
    <title> My Website</title>
  </head>
  <body>
            <h1>Hello</h1>
            <input type="checkbox">
            <button >Click here</button><br>
           <ul>
             <li>Google</li>
             <li>COMSATS</li>
             <li>Comp: Eng:</li>
           </ul>
  </body>
</html>
```



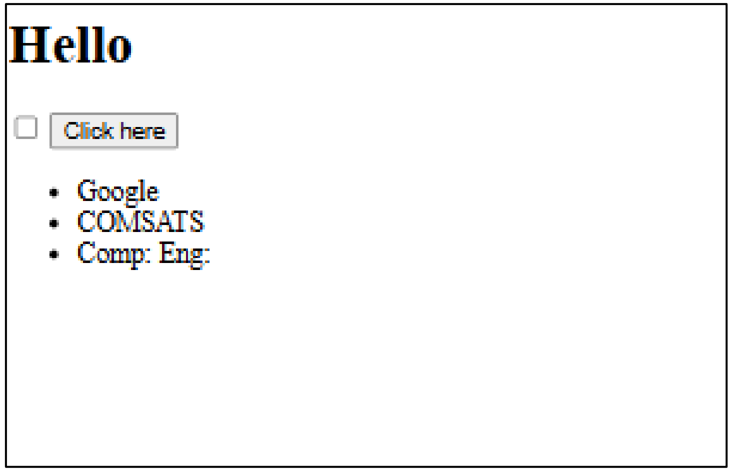
Hello

□

- Google
- COMSATS
- Comp: Eng:

Objects Properties

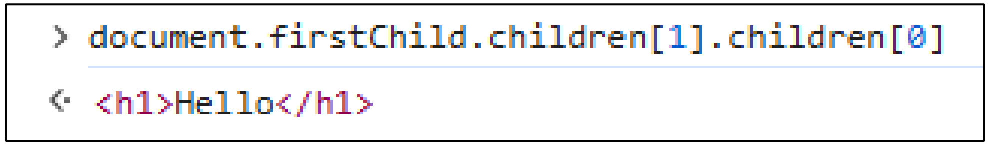
```
<!DOCTYPE html>
<html>
  <head>
    <title> My Website</title>
  </head>
  <body>
        <h1>Hello</h1>
        <input type="checkbox">
        <button >Click here</button><br>
       <ul>
         <li>Google</li>
         <li>COMSATS</li>
         <li>Comp: Eng:</li>
       </ul>
  </body>
</html>
```



Hello

□ Click here

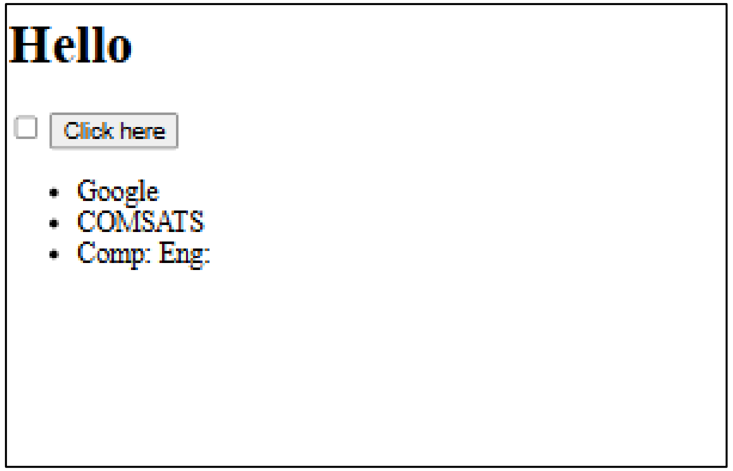
- Google
- COMSATS
- Comp: Eng:



```
> document.firstChild.children[1].children[0]
<· <h1>Hello</h1>
```

Objects Properties

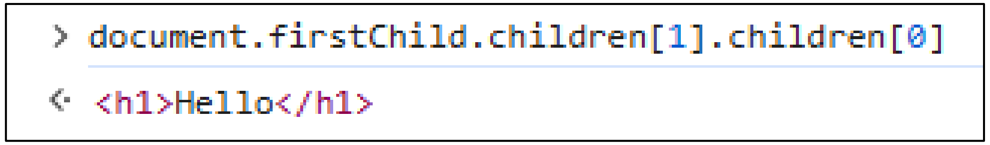
```
<!DOCTYPE html>
<html>
  <head>
    <title> My Website</title>
  </head>
  <body>
          <h1>Hello</h1>
          <input type="checkbox">
          <button >Click here</button><br>
         <ul>
           <li>Google</li>
           <li>COMSATS</li>
           <li>Comp: Eng:</li>
         </ul>
  </body>
</html>
```



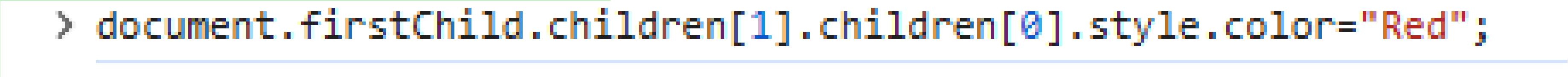
Hello

□ Click here

- Google
- COMSATS
- Comp: Eng:



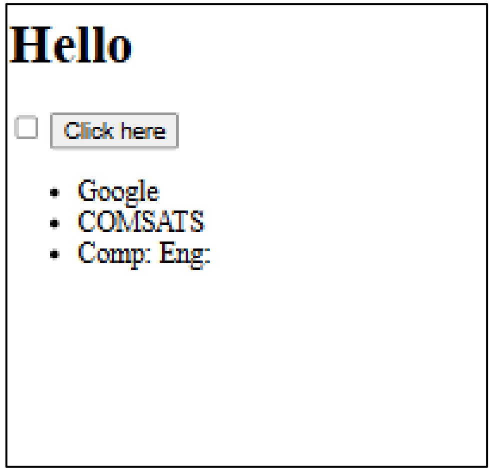
```
> document.firstChild.children[1].children[0]
< <h1>Hello</h1>
```



```
> document.firstChild.children[1].children[0].style.color="Red";
```

Objects Properties

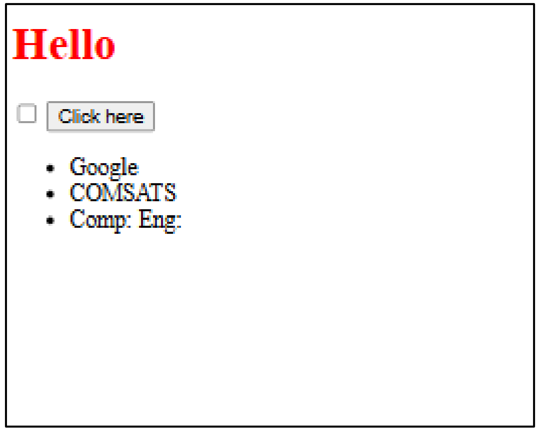
```
<!DOCTYPE html>
<html>
  <head>
    <title> My Website</title>
  </head>
  <body>
        <h1>Hello</h1>
        <input type="checkbox">
        <button >Click here</button><br>
       <ul>
         <li>Google</li>
         <li>COMSATS</li>
         <li>Comp: Eng:</li>
       </ul>
  </body>
</html>
```



Hello

☐ Click here

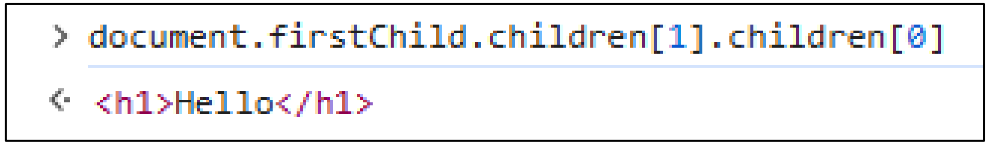
- Google
- COMSATS
- Comp: Eng:



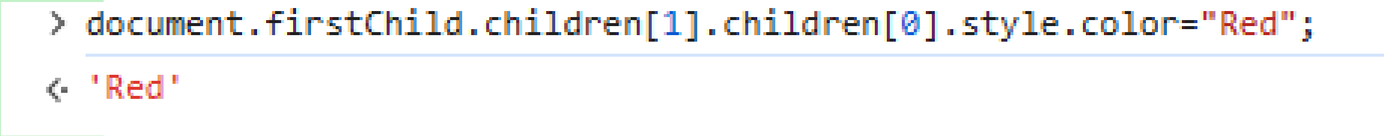
Hello

☐ Click here

- Google
- COMSATS
- Comp: Eng:



```
> document.firstChild.children[1].children[0]
<· <h1>Hello</h1>
```

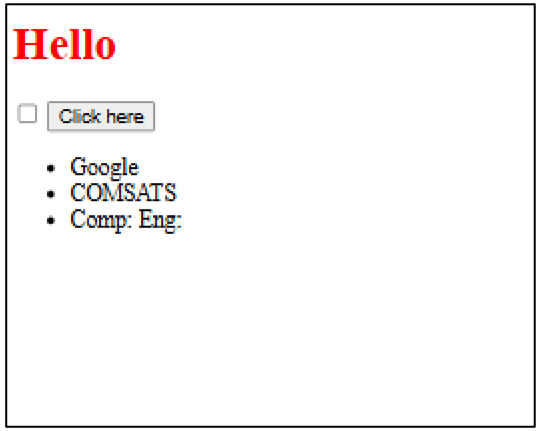


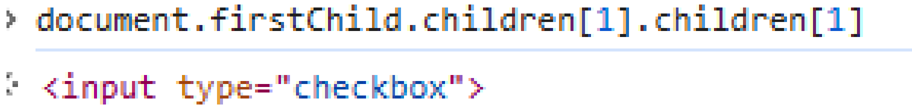
```
> document.firstChild.children[1].children[0].style.color="Red";
<· 'Red'
```

This is how we can change the properties of Objects

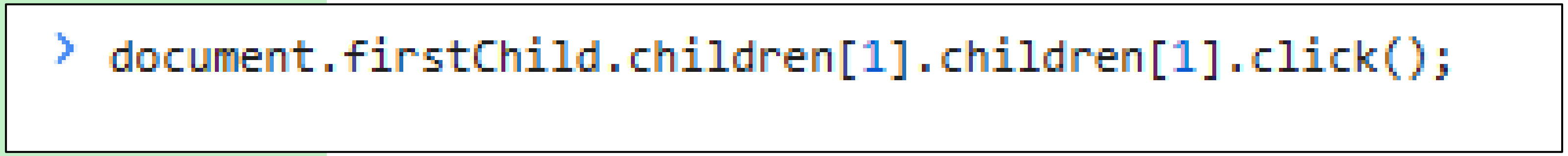
Objects Methods

```
<!DOCTYPE html>
<html>
  <head>
    <title> My Website</title>
  </head>
  <body>
        <h1>Hello</h1>
        <input type="checkbox">
        <button >Click here</button><br>
       <ul>
         <li>Google</li>
         <li>COMSATS</li>
         <li>Comp: Eng:</li>
       </ul>
  </body>
</html>
```





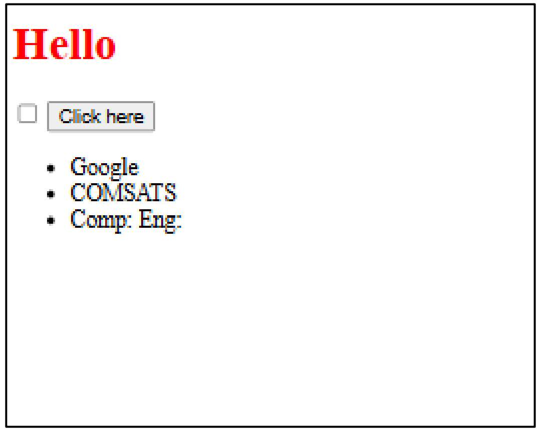
```
> document.firstChild.children[1].children[1]
< <input type="checkbox">
```



```
> document.firstChild.children[1].children[1].click();
```

Objects Methods

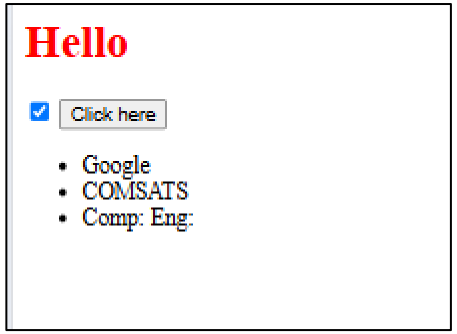
```
<!DOCTYPE html>
<html>
  <head>
    <title> My Website</title>
  </head>
  <body>
          <h1>Hello</h1>
          <input type="checkbox">
          <button >Click here</button><br>
         <ul>
           <li>Google</li>
           <li>COMSATS</li>
           <li>Comp: Eng:</li>
         </ul>
  </body>
</html>
```



Hello

☐ Click here

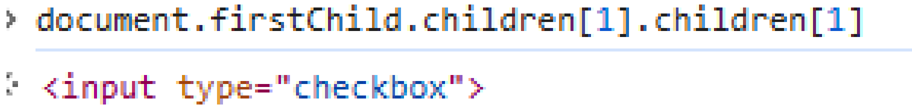
- Google
- COMSATS
- Comp: Eng:



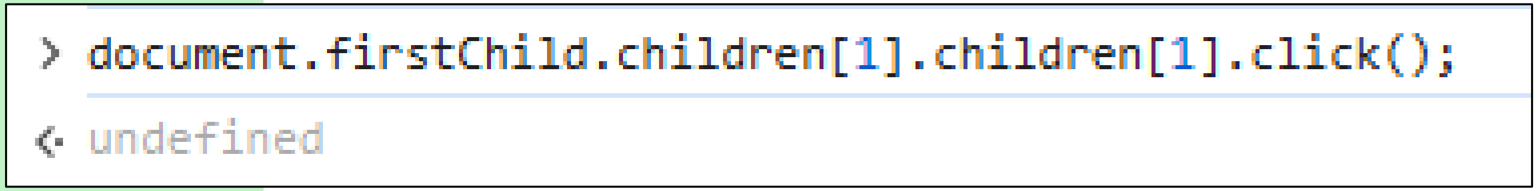
Hello

☑ Click here

- Google
- COMSATS
- Comp: Eng:



```
> document.firstChild.children[1].children[1]
< <input type="checkbox">
```



```
> document.firstChild.children[1].children[1].click();
< undefined
```

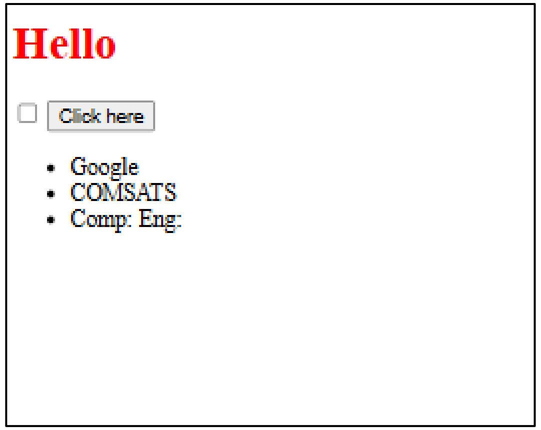
This is how Object Method Works!

Property vs Method

- **Property**
  - Gives information about an object
  - No parentheses
  - Example: element.style.color
- **Method**
  - An action belonging to an object
  - Has parentheses () because it runs something
  - Example: element.click()
- **Method vs Function (Important!)**
  - A function is just a reusable block of code
  - A method is a function attached to an object So:
    - All methods are functions → but not all functions are methods

Practice Task

```
<!DOCTYPE html>
<html>
  <head>
    <title> My Website</title>
  </head>
  <body>
          <h1>Hello</h1>
          <input type="checkbox">
          <button >Click here</button><br>
         <ul>
           <li>Google</li>
           <li>COMSATS</li>
           <li>Comp: Eng:</li>
         </ul>
  </body>
</html>
```



Hello

☐ Click here

- Google
- COMSATS
- Comp: Eng:

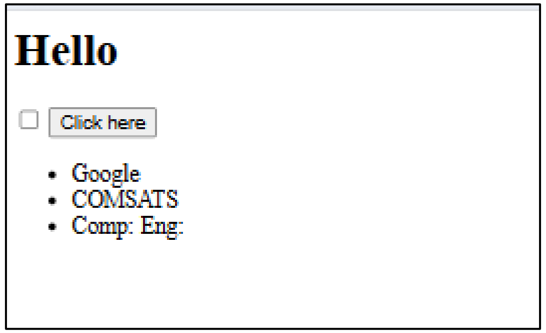
Task:-Change 3rd <li> text using console

Selecting HTML Elements with JavaScript

- getElementsByTagName()
- getElementById()
- getElementsByClassName()
- querySelector()
- querySelectorAll()

getElementsByTagName()

```
<!DOCTYPE html>
<html>
  <head>
    <title> My Website</title>
  </head>
  <body>
    <h1 class="heading">Hello</h1>
     <input class="check" type="checkbox">
    <button id="click" class="btn">Click here</button><br>
    <ul class="list">
      <li id="google" class="list item">Google</li>
      <li id="uni" class="list item">COMSATS</li>
      <li id="department" class="list item">Comp: Eng:</li>
    </ul>
  </body>
</html>
```



Hello

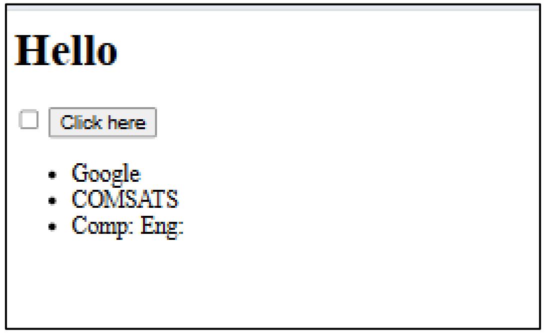
☐ Click here

- Google
- COMSATS
- Comp: Eng:

```
> document.getElementsByTagName("h1")
```

getElementsByTagName()

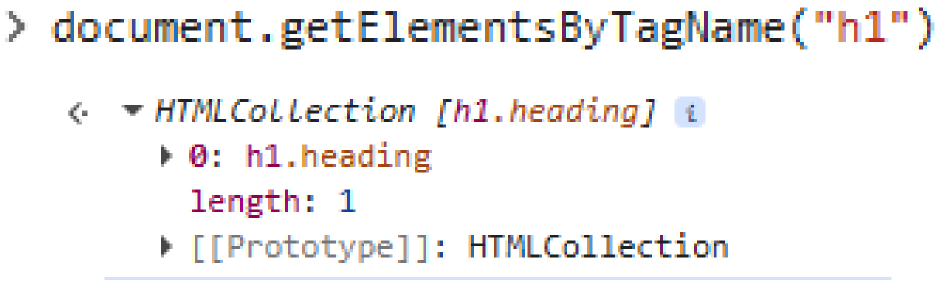
```
<!DOCTYPE html>
<html>
  <head>
    <title> My Website</title>
  </head>
  <body>
    <h1 class="heading">Hello</h1>
     <input class="check" type="checkbox">
    <button id="click" class="btn">Click here</button><br>
    <ul class="list">
      <li id="google" class="list item">Google</li>
      <li id="uni" class="list item">COMSATS</li>
      <li id="department" class="list item">Comp: Eng:</li>
    </ul>
  </body>
</html>
```



Hello

□ Click here

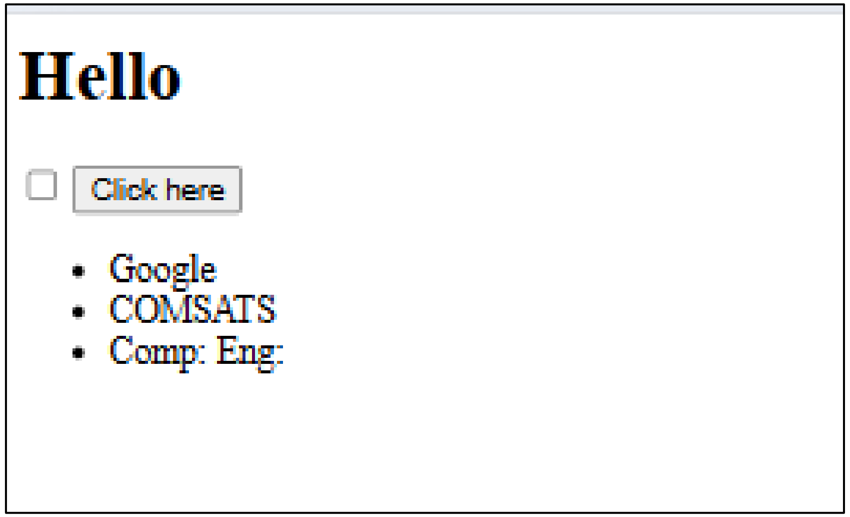
- Google
- COMSATS
- Comp: Eng:



```
> document.getElementsByTagName("h1")

  <· ▾ HTMLCollection [h1.heading] i
       ▸ 0: h1.heading
         length: 1
       ▸ [[Prototype]]: HTMLCollection
```

getElementsByTagName()

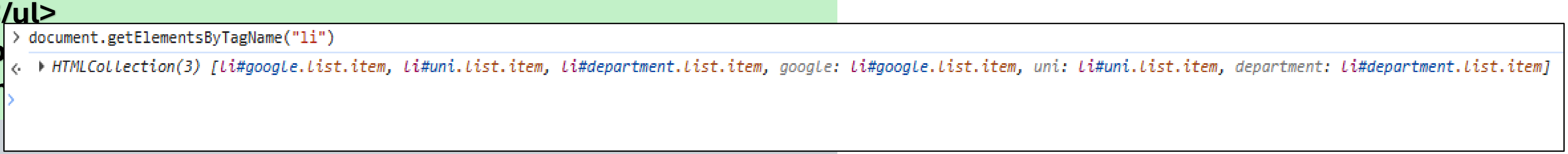
```
<!DOCTYPE html>
<html>
   <head>
      <title> My Website</title>
   </head>
   <body>
      <h1 class="heading">Hello</h1>
       <input class="check" type="checkbox">
      <button id="click" class="btn">Click here</button><br>
      <ul class="list">
         <li id="google" class="list item">Google</li>
         <li id="uni" class="list item">COMSATS</li>
         <li id="department" class="list item">Comp: Eng:</li>
      </ul>
   </b
</htr
```



Hello

□ Click here

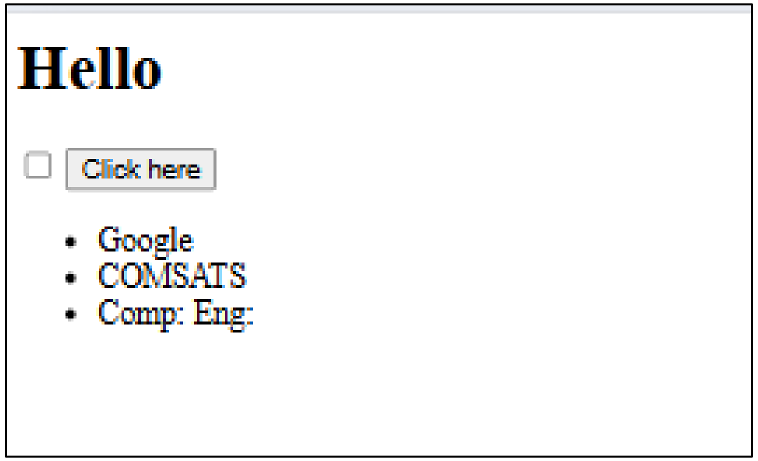
- Google
- COMSATS
- Comp: Eng:

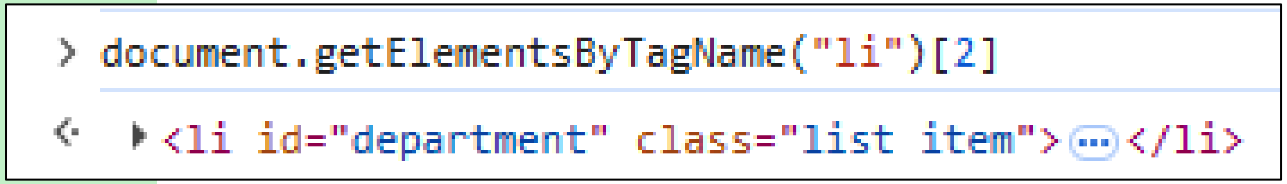


```
> document.getElementsByTagName("li")
‹ ▸ HTMLCollection(3) [li#google.list.item, li#uni.list.item, li#department.list.item, google: li#google.list.item, uni: li#uni.list.item, department: li#department.list.item]
>
```

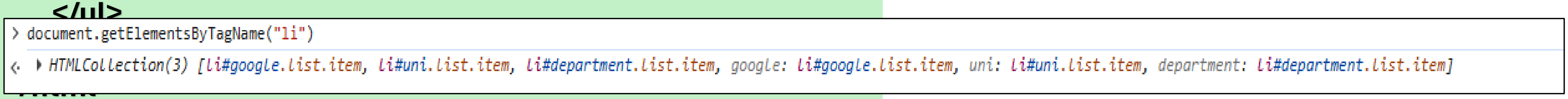
getElementsByTagName()

```
<!DOCTYPE html>
<html>
  <head>
    <title> My Website</title>
  </head>
  <body>
    <h1 class="heading">Hello</h1>
     <input class="check" type="checkbox">
    <button id="click" class="btn">Click here</button><br>
    <ul class="list">
      <li id="google" class="list item">Google</li>
      <li id="uni" class="list item">COMSATS</li>
      <li id="department" class="list item">Comp: Eng:</li>
    </ul>
</html>
```





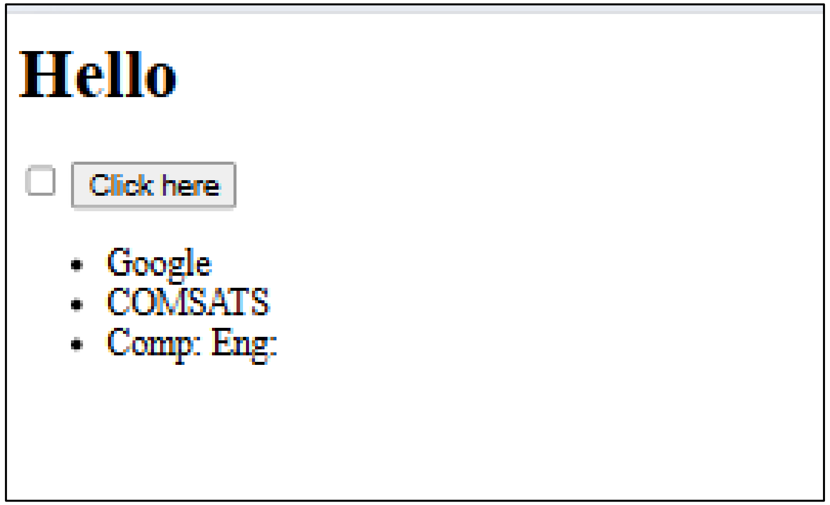
```
> document.getElementsByTagName("li")[2]
<· ▸<li id="department" class="list item">…</li>
```



```
> document.getElementsByTagName("li")
<· ▸HTMLCollection(3) [li#google.list.item, li#uni.list.item, li#department.list.item, google: li#google.list.item, uni: li#uni.list.item, department: li#department.list.item]
```

getElementsByTagName()

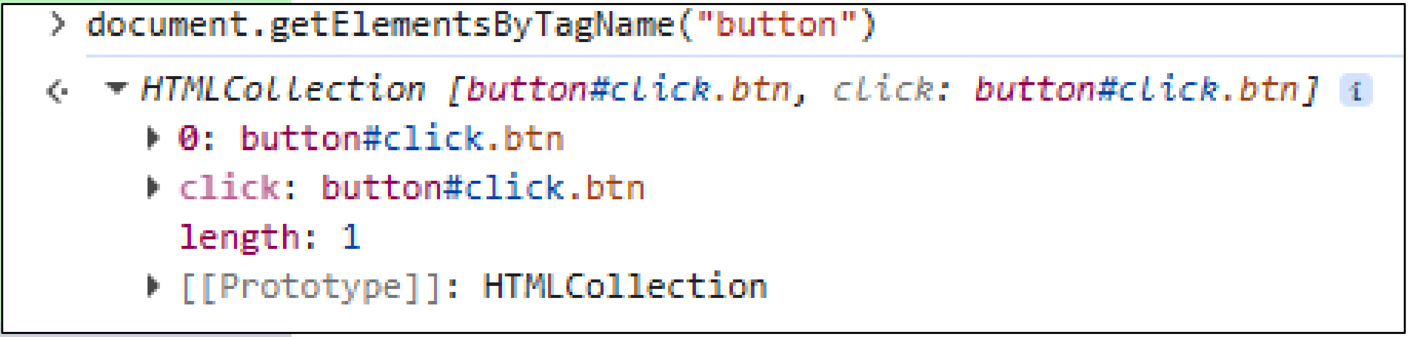
```
<!DOCTYPE html>
<html>
  <head>
    <title> My Website</title>
  </head>
  <body>
    <h1 class="heading">Hello</h1>
     <input class="check" type="checkbox">
    <button id="click" class="btn">Click here</button><br>
    <ul class="list">
      <li id="google" class="list item">Google</li>
      <li id="uni" class="list item">COMSATS</li>
      <li id="department" class="list item">Comp:
    </ul>
  </body>
</html>
```



Hello

☐ Click here

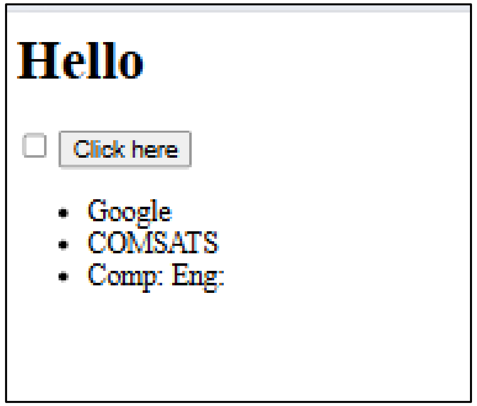
- Google
- COMSATS
- Comp: Eng:



```
> document.getElementsByTagName("button")
<· ▼HTMLCollection [button#click.btn, click: button#click.btn] ℹ
    ▶ 0: button#click.btn
    ▶ click: button#click.btn
      length: 1
    ▶ [[Prototype]]: HTMLCollection
```

Manipulating Items with getElementsByTagName()

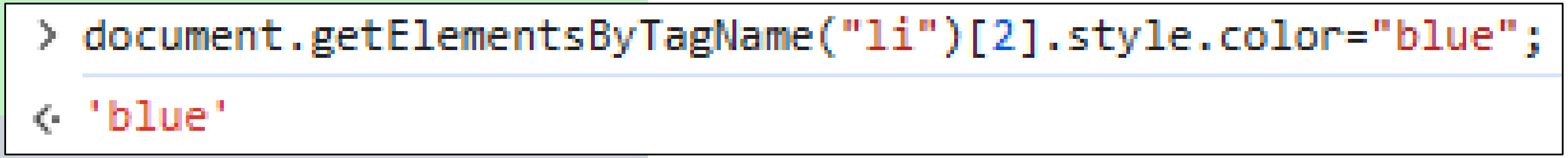
```
<!DOCTYPE html>
<html>
  <head>
    <title> My Website</title>
  </head>
  <body>
    <h1 class="heading">Hello</h1>
     <input class="check" type="checkbox">
    <button id="click" class="btn">Click here</button><br>
    <ul class="list">
      <li id="google" class="list item">Google</li>
      <li id="uni" class="list item">COMSATS</li>
      <li id="department" class="list item">Comp: Eng:</li>
    </ul>
  </body>
</html>
```



Hello

□

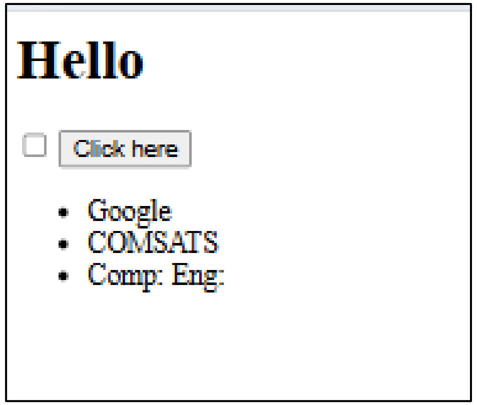
- Google
- COMSATS
- Comp: Eng:



```
> document.getElementsByTagName("li")[2].style.color="blue";
<· 'blue'
```

Manipulating Items with getElementsByTagName()

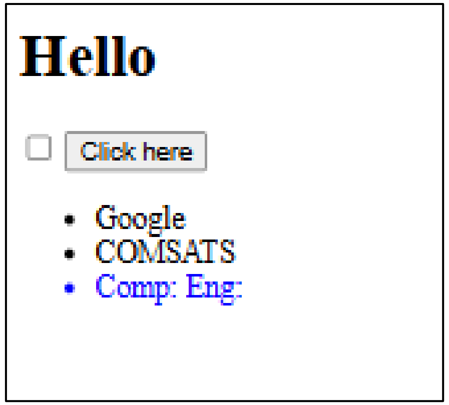
```
<!DOCTYPE html>
<html>
  <head>
    <title> My Website</title>
  </head>
  <body>
    <h1 class="heading">Hello</h1>
     <input class="check" type="checkbox">
    <button id="click" class="btn">Click here</button><br>
    <ul class="list">
      <li id="google" class="list item">Google</li>
      <li id="uni" class="list item">COMSATS</li>
      <li id="department" class="list item">Comp: Eng:</li>
    </ul>
  </body>
</html>
```



Hello

☐ Click here

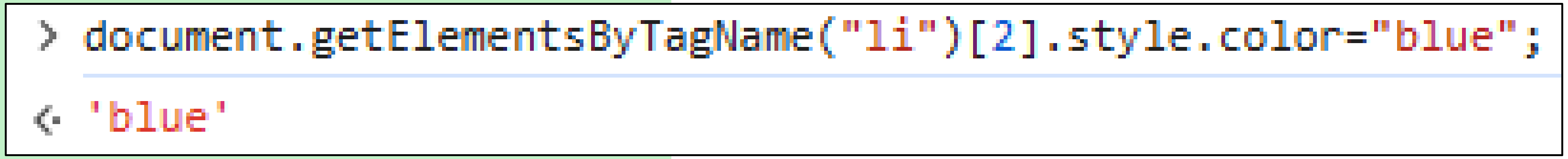
- Google
- COMSATS
- Comp: Eng:



Hello

☐ Click here

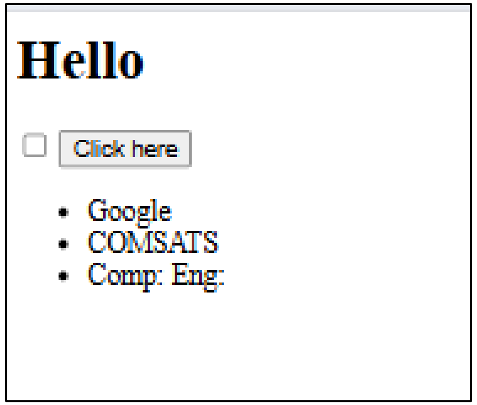
- Google
- COMSATS
- Comp: Eng:



```
> document.getElementsByTagName("li")[2].style.color="blue";
<· 'blue'
```

Working with Arrays

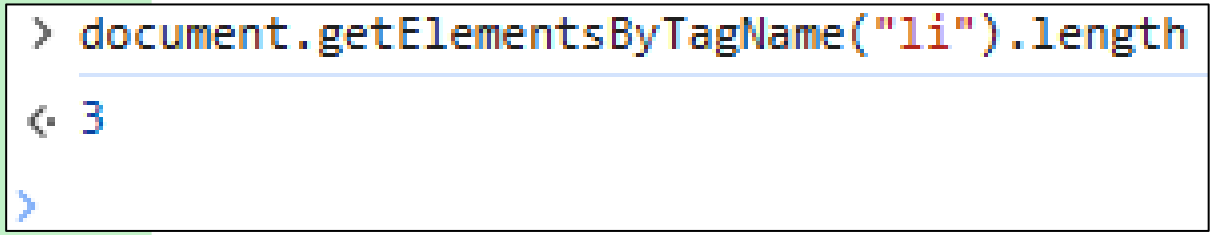
```
<!DOCTYPE html>
<html>
  <head>
    <title> My Website</title>
  </head>
  <body>
    <h1 class="heading">Hello</h1>
     <input class="check" type="checkbox">
    <button id="click" class="btn">Click here</button><br>
    <ul class="list">
      <li id="google" class="list item">Google</li>
      <li id="uni" class="list item">COMSATS</li>
      <li id="department" class="list item">Comp: Eng:</li>
    </ul>
  </body>
</html>
```



Hello

☐ Click here

- Google
- COMSATS
- Comp: Eng:

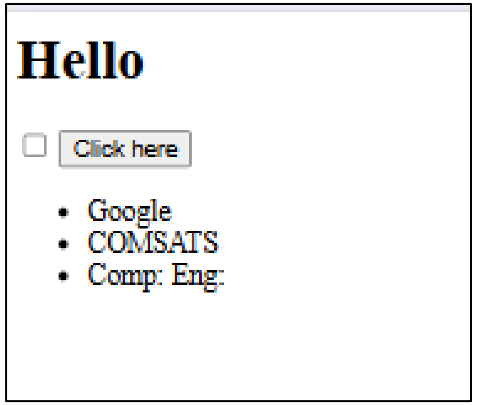


```
> document.getElementsByTagName("li").length
<· 3
>
```

- Returned items behave like arrays
- Use .length to count elements

getElementsByClassName()

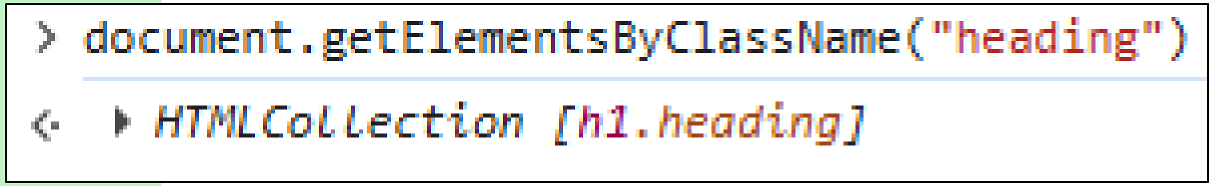
```
<!DOCTYPE html>
<html>
  <head>
    <title> My Website</title>
  </head>
  <body>
    <h1 class="heading">Hello</h1>
     <input class="check" type="checkbox">
    <button id="click" class="btn">Click here</button><br>
    <ul class="list">
      <li id="google" class="list item">Google</li>
      <li id="uni" class="list item">COMSATS</li>
      <li id="department" class="list item">Comp: Eng:</li>
    </ul>
  </body>
</html>
```



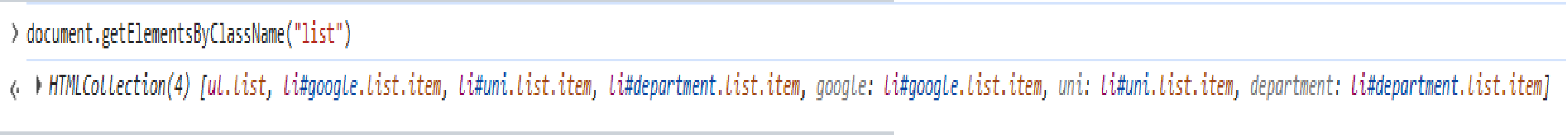
Hello

□ Click here

- Google
- COMSATS
- Comp: Eng:



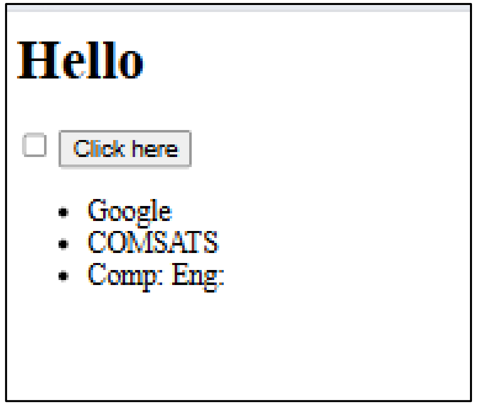
```
> document.getElementsByClassName("heading")
<· ▸ HTMLCollection [h1.heading]
```



```
> document.getElementsByClassName("list")
<· ▸ HTMLCollection(4) [ul.list, li#google.list.item, li#uni.list.item, li#department.list.item, google: li#google.list.item, uni: li#uni.list.item, department: li#department.list.item]
```

getElementsByClassName()

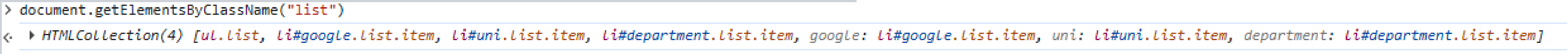
```
<!DOCTYPE html>
<html>
  <head>
    <title> My Website</title>
  </head>
  <body>
    <h1 class="heading">Hello</h1>
     <input class="check" type="checkbox">
    <button id="click" class="btn">Click here</button><br>
    <ul class="list">
      <li id="google" class="list item">Google</li>
      <li id="uni" class="list item">COMSATS</li>
      <li id="department" class="list item">Comp: Eng:</li>
    </ul>
  </body>
</html>
```



Hello

□ Click here

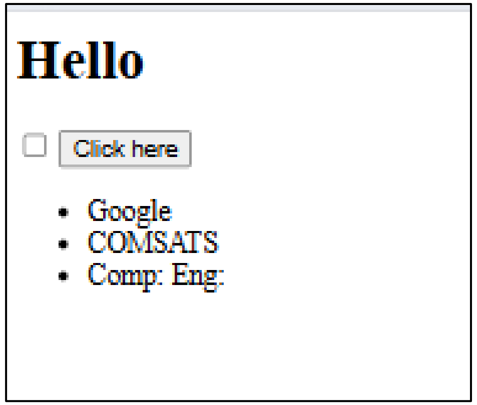
- Google
- COMSATS
- Comp: Eng:



```
> document.getElementsByClassName("list")
<· ▸ HTMLCollection(4) [ul.list, li#google.list.item, li#uni.list.item, li#department.list.item, google: li#google.list.item, uni: li#uni.list.item, department: li#department.list.item]
```

getElementsByClassName()

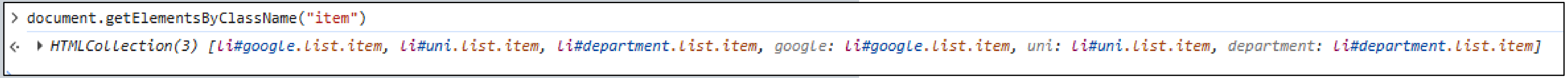
```
<!DOCTYPE html>
<html>
  <head>
    <title> My Website</title>
  </head>
  <body>
    <h1 class="heading">Hello</h1>
     <input class="check" type="checkbox">
    <button id="click" class="btn">Click here</button><br>
    <ul class="list">
      <li id="google" class="list item">Google</li>
      <li id="uni" class="list item">COMSATS</li>
      <li id="department" class="list item">Comp: Eng:</li>
    </ul>
  </body>
</html>
```



Hello

□ Click here

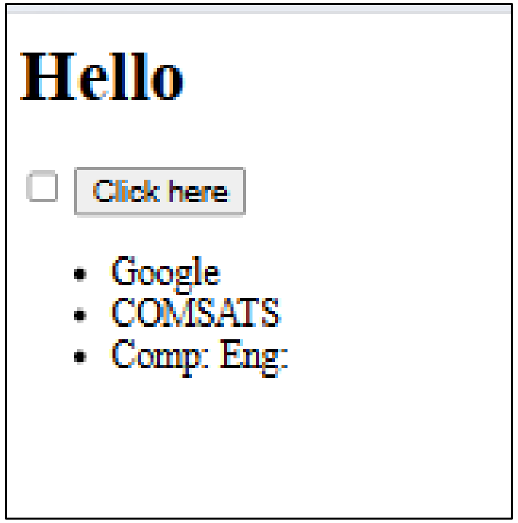
- Google
- COMSATS
- Comp: Eng:



```
> document.getElementsByClassName("item")
< ▸ HTMLCollection(3) [li#google.list.item, li#uni.list.item, li#department.list.item, google: li#google.list.item, uni: li#uni.list.item, department: li#department.list.item]
```

getElementById()

```
<body>
     <h1 class="heading">Hello</h1>
      <input class="check" type="checkbox">
     <button id="click" class="btn">Click here</button><br>
     <ul class="list">
        <li id="google" class="list item">Google</li>
        <li id="uni" class="list item">COMSATS</li>
        <li id="department" class="list item">Comp: Eng:</li>
     </ul>
 </body>
```

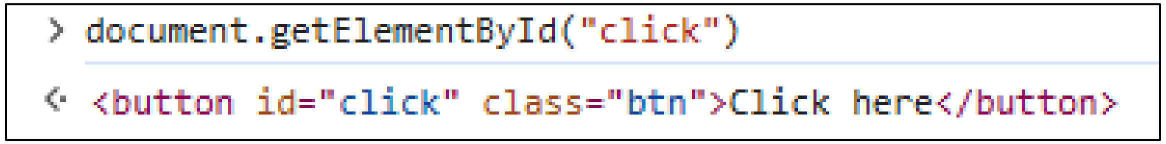


Hello

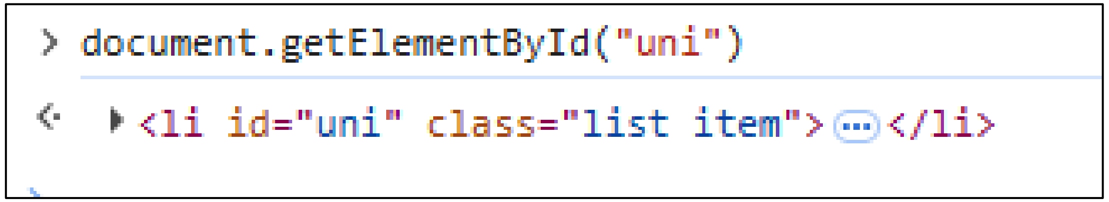
☐ Click here

- Google
- COMSATS
- Comp: Eng:

- Returns a single element (ID must be unique)



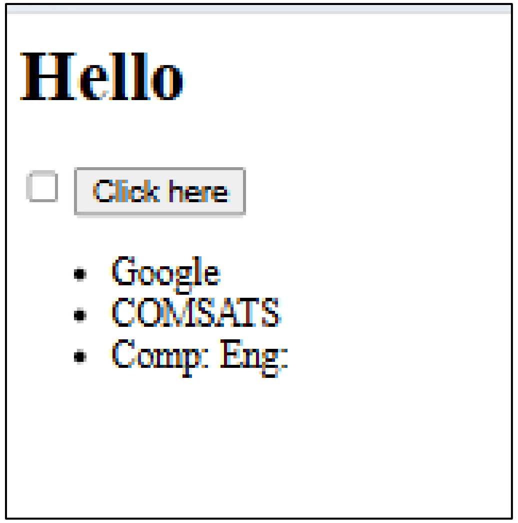
```
> document.getElementById("click")
< <button id="click" class="btn">Click here</button>
```



```
> document.getElementById("uni")
< ▸<li id="uni" class="list item">…</li>
```

getElementById()

```
<body>
     <h1 class="heading">Hello</h1>
      <input class="check" type="checkbox">
     <button id="click" class="btn">Click here</button><br>
     <ul class="list">
        <li id="google" class="list item">Google</li>
        <li id="uni" class="list item">COMSATS</li>
        <li id="department" class="list item">Comp: Eng:</li>
     </ul>
 </body>
```

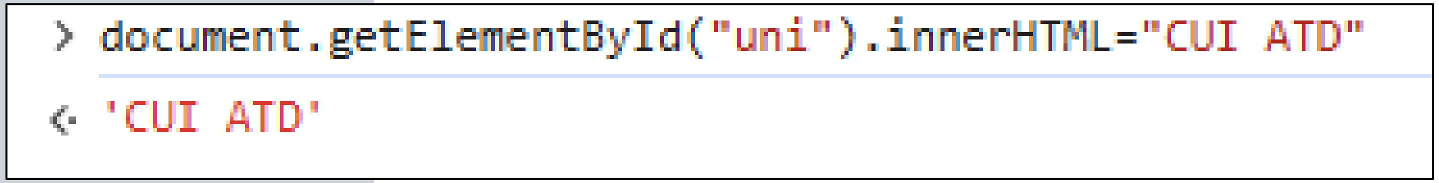


Hello

☐ Click here

- Google
- COMSATS
- Comp: Eng:

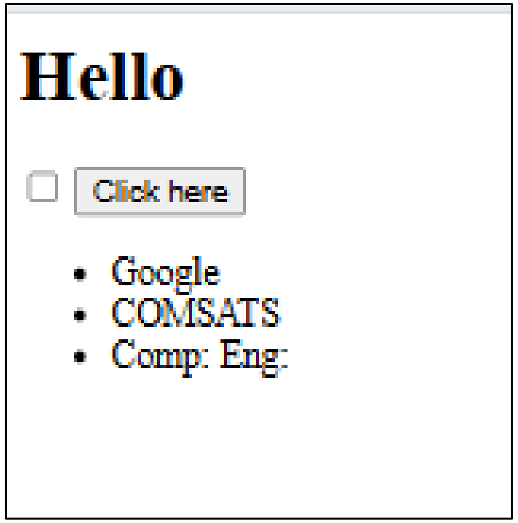
- Returns a single element (ID must be unique)
- No array returned → can modify directly

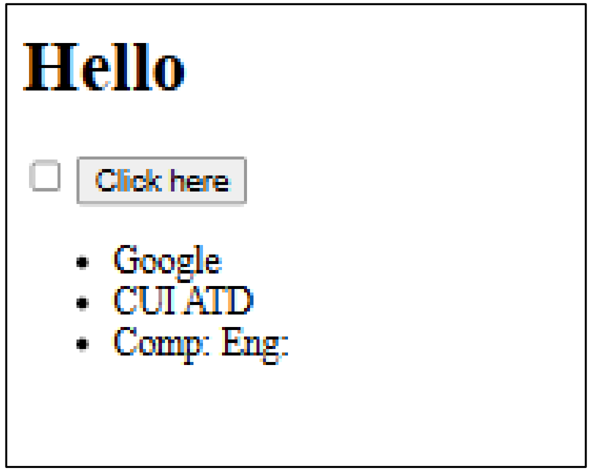


```
> document.getElementById("uni").innerHTML="CUI ATD"
<· 'CUI ATD'
```

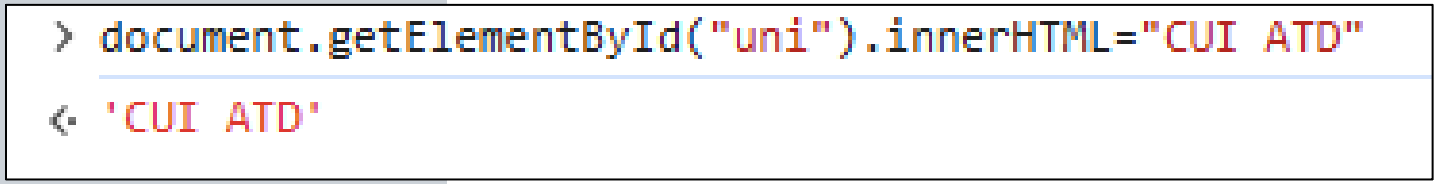
getElementById()

```
<body>
    <h1 class="heading">Hello</h1>
     <input class="check" type="checkbox">
    <button id="click" class="btn">Click here</button><br>
    <ul class="list">
      <li id="google" class="list item">Google</li>
      <li id="uni" class="list item">COMSATS</li>
      <li id="department" class="list item">Comp: Eng:</li>
    </ul>
 </body>
```





- Returns a single element (ID must be unique)
- No array returned → can modify directly

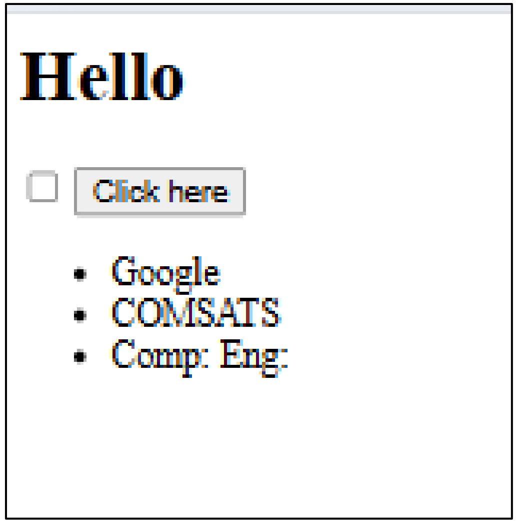


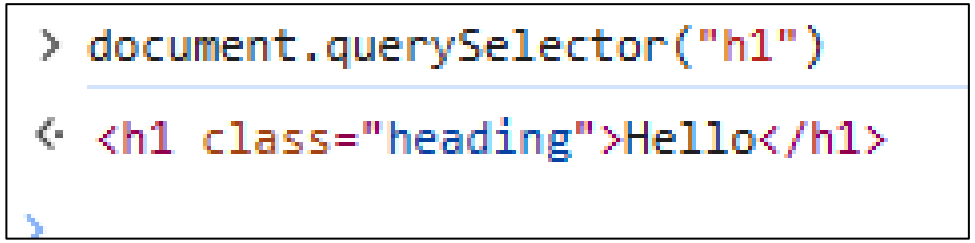
```
> document.getElementById("uni").innerHTML="CUI ATD"
<· 'CUI ATD'
```

querySelector()

```
<body>
      <h1 class="heading">Hello</h1>
       <input class="check" type="checkbox">
      <button id="click" class="btn">Click here</button><br>
      <ul class="list">
         <li id="google" class="list item">Google</li>
         <li id="uni" class="list item">COMSATS</li>
         <li id="department" class="list item">Comp: Eng:</li>
      </ul>
  </body>
```

- Returns first element matching a CSS selector



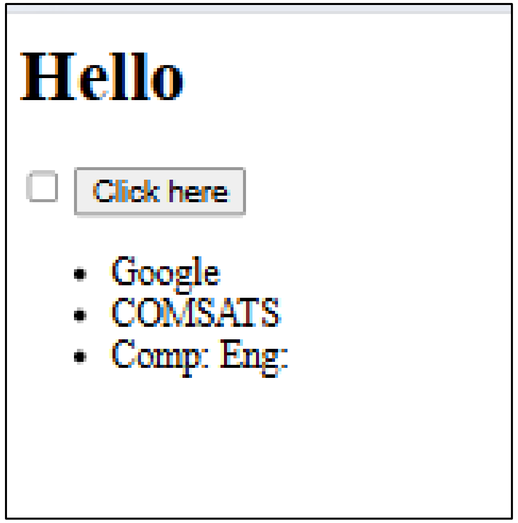


```
> document.querySelector("h1")
< <h1 class="heading">Hello</h1>
>
```

querySelector()

```
<body>
     <h1 class="heading">Hello</h1>
      <input class="check" type="checkbox">
     <button id="click" class="btn">Click here</button><br>
     <ul class="list">
        <li id="google" class="list item">Google</li>
        <li id="uni" class="list item">COMSATS</li>
        <li id="department" class="list item">Comp: Eng:</li>
     </ul>
 </body>
```

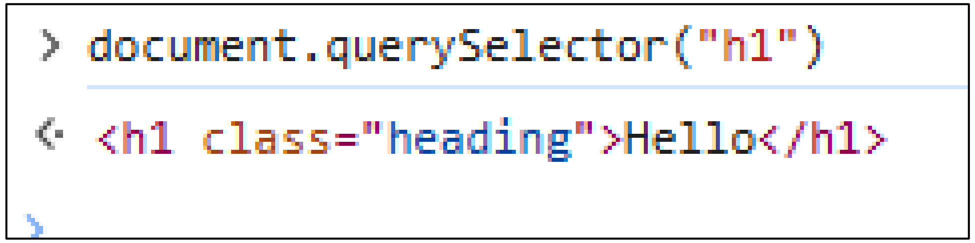
- Returns first element matching a CSS selector



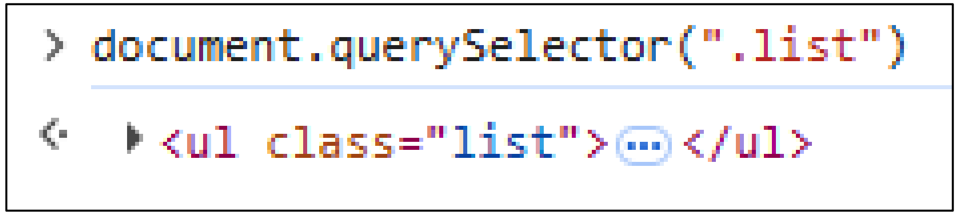
Hello

☐ Click here

- Google
- COMSATS
- Comp: Eng:



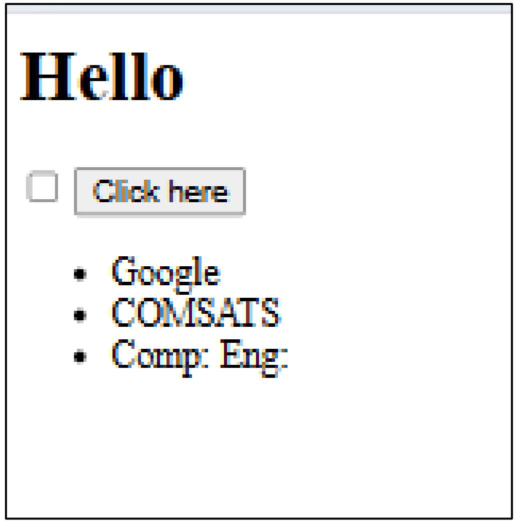
```
> document.querySelector("h1")
<· <h1 class="heading">Hello</h1>
>
```



```
> document.querySelector(".list")
<·  ▶<ul class="list">…</ul>
```

querySelector()

```
<body>
     <h1 class="heading">Hello</h1>
      <input class="check" type="checkbox">
     <button id="click" class="btn">Click here</button><br>
     <ul class="list">
        <li id="google" class="list item">Google</li>
        <li id="uni" class="list item">COMSATS</li>
        <li id="department" class="list item">Comp: Eng:</li>
     </ul>
 </body>
```

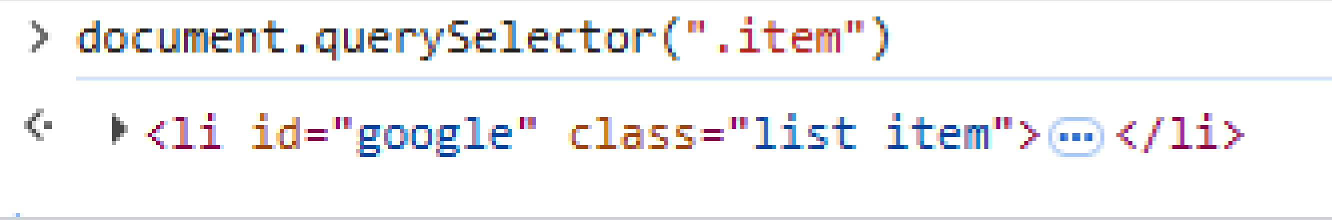


Hello

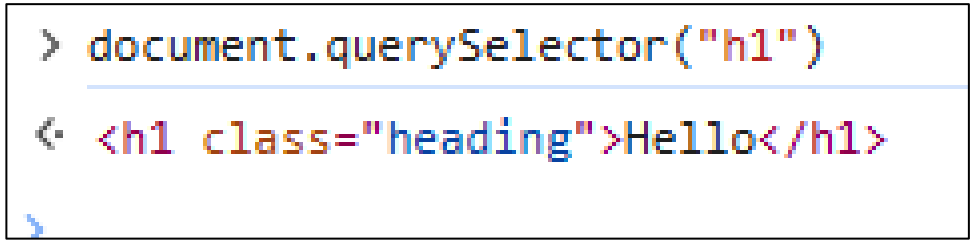
☐ Click here

- Google
- COMSATS
- Comp: Eng:

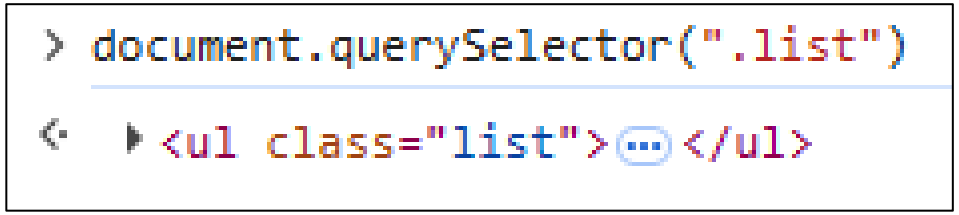
- Returns first element matching a CSS selector
- Selector like CSS: element, .class, #id



```
> document.querySelector(".item")
<  ▸<li id="google" class="list item">…</li>
```



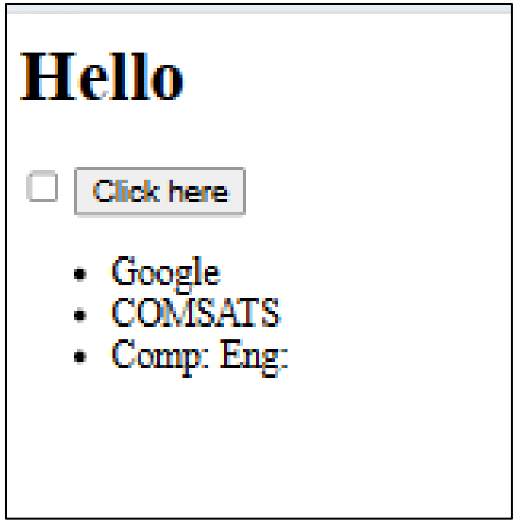
```
> document.querySelector("h1")
<  <h1 class="heading">Hello</h1>
>
```



```
> document.querySelector(".list")
<  ▸<ul class="list">…</ul>
```

querySelector()

```
<body>
    <h1 class="heading">Hello</h1>
     <input class="check" type="checkbox">
    <button id="click" class="btn">Click here</button><br>
    <ul class="list">
      <li id="google" class="list item">Google</li>
      <li id="uni" class="list item">COMSATS</li>
      <li id="department" class="list item">Comp: Eng:</li>
    </ul>
 </body>
```

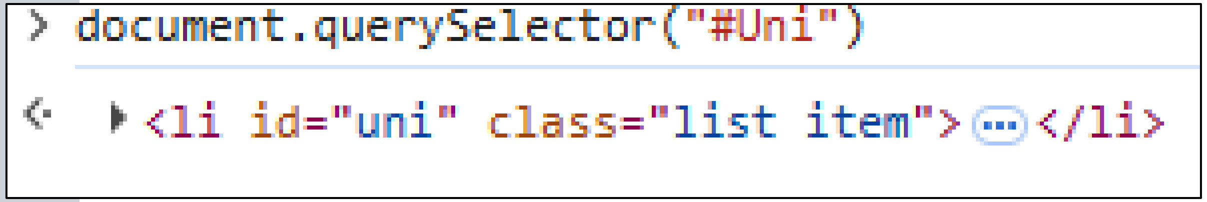


Hello

□ Click here

- Google
- COMSATS
- Comp: Eng:

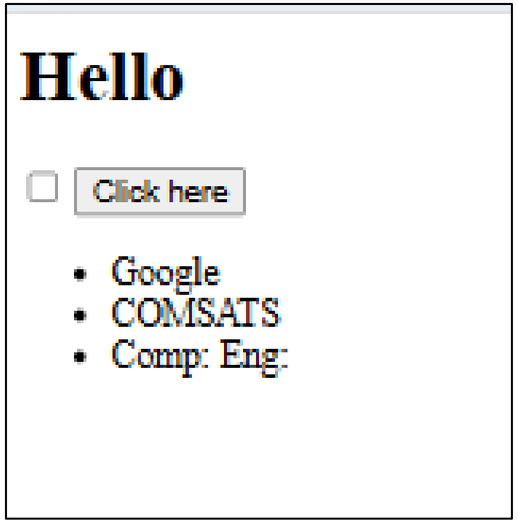
- Returns first element matching a CSS selector
- Selector like CSS: element, .class, #id



```
> document.querySelector("#Uni")
<·  ▶ <li id="uni" class="list item">…</li>
```

querySelector()

```
<body>
    <h1 class="heading">Hello</h1>
     <input class="check" type="checkbox">
    <button id="click" class="btn">Click here</button><br>
    <ul class="list">
      <li id="google" class="list item">Google</li>
      <li id="uni" class="list item">COMSATS</li>
      <li id="department" class="list item">Comp: Eng:</li>
    </ul>
 </body>
```

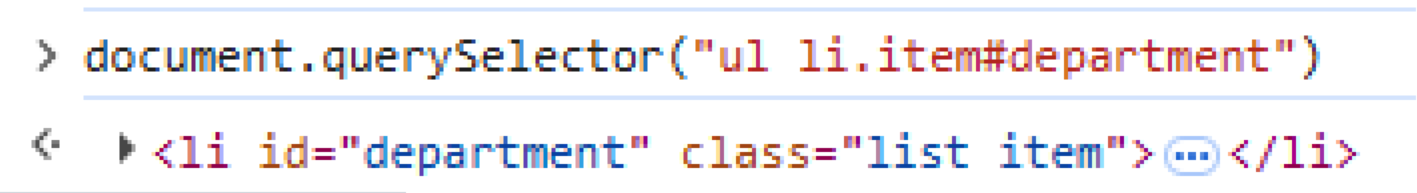


Hello

☐ Click here

- Google
- COMSATS
- Comp: Eng:

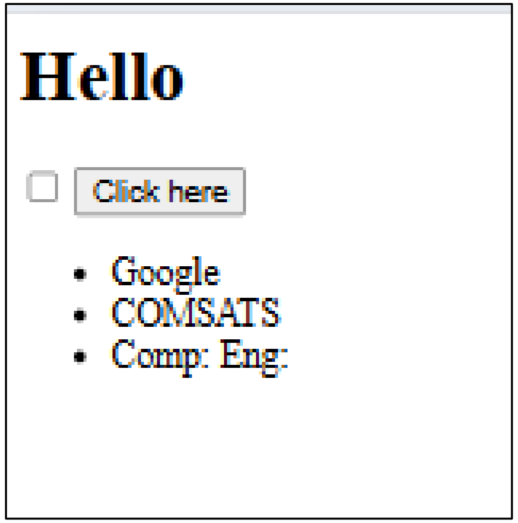
- Returns first element matching a CSS selector
- Selector like CSS: element, .class, #id
- Supports complex selectors



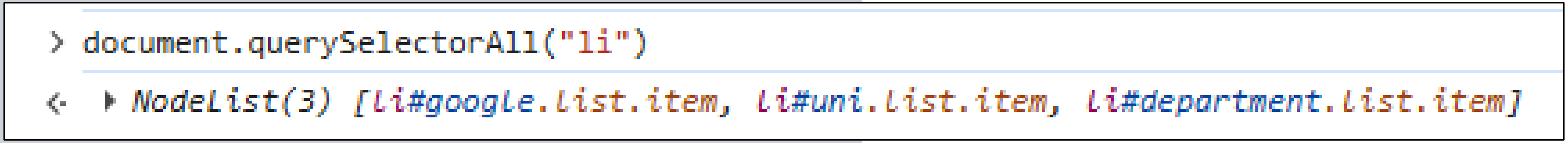
```
> document.querySelector("ul li.item#department")
<· ▸<li id="department" class="list item">…</li>
```

querySelectorAll()

```
<body>
      <h1 class="heading">Hello</h1>
       <input class="check" type="checkbox">
      <button id="click" class="btn">Click here</button><br>
      <ul class="list">
         <li id="google" class="list item">Google</li>
         <li id="uni" class="list item">COMSATS</li>
         <li id="department" class="list item">Comp: Eng:</li>
      </ul>
  </body>
```



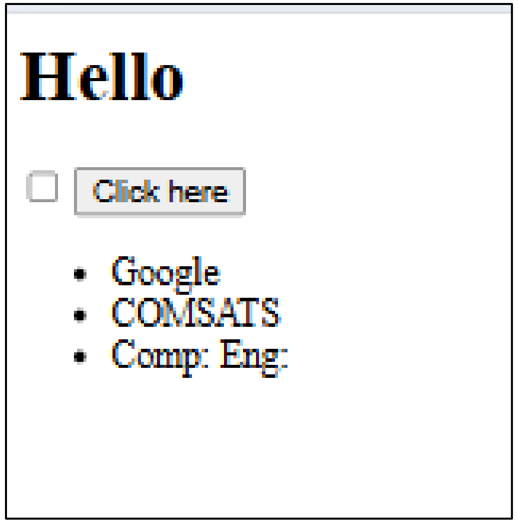
- Returns all elements matching selector



```
> document.querySelectorAll("li")
<· ▸ NodeList(3) [li#google.list.item, li#uni.list.item, li#department.list.item]
```

querySelectorAll()

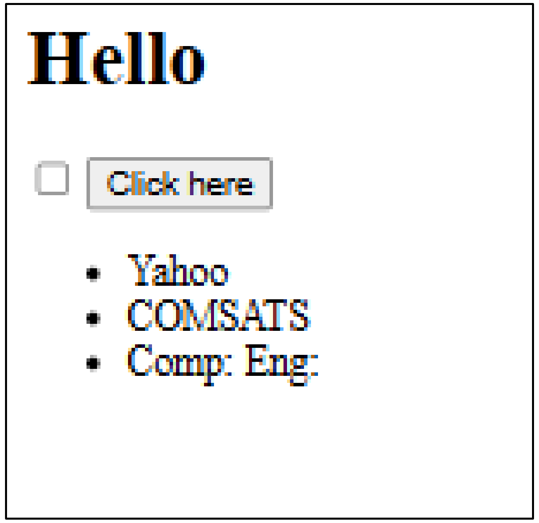
```
<body>
      <h1 class="heading">Hello</h1>
       <input class="check" type="checkbox">
      <button id="click" class="btn">Click here</button><br>
      <ul class="list">
         <li id="google" class="list item">Google</li>
         <li id="uni" class="list item">COMSATS</li>
         <li id="department" class="list item">Comp: Eng:</li>
      </ul>
 </body>
```



Hello

□ Click here

- Google
- COMSATS
- Comp: Eng:

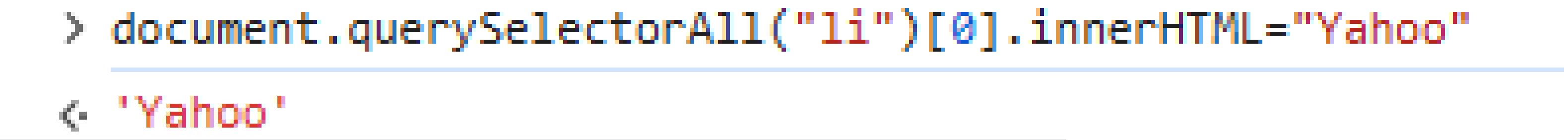


Hello

□ Click here

- Yahoo
- COMSATS
- Comp: Eng:

- Returns all elements matching selector
- Use index to manipulate items



```
> document.querySelectorAll("li")[0].innerHTML="Yahoo"
<· 'Yahoo'
```

When to Use Which?

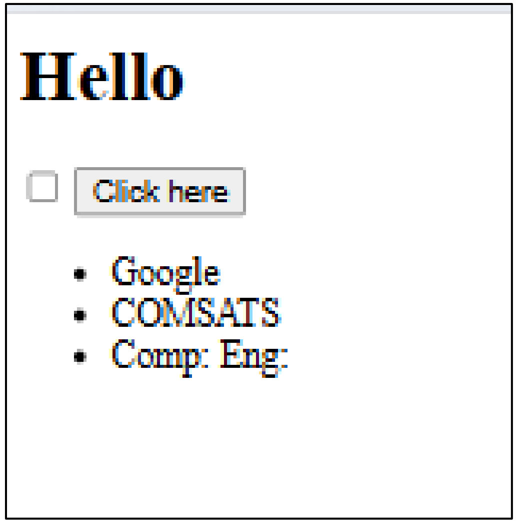
- querySelector / querySelectorAll
  - Most flexible
  - Can select by tag, class, ID, attributes, combinations
  - Preferred in modern, real-world development
- getElement... methods
  - Faster but more specific
  - Only target one type (ID, class, tag, name)
  - Still useful for simple selections or performance-focused cases
- Simple rule:
  - If you can select it with CSS, you can select it with querySelector

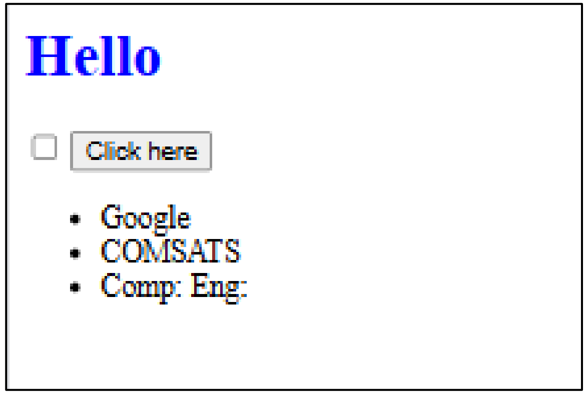
DOM Manipulation: Changing CSS with JavaScript

- Select elements → change their styles dynamically
- Uses .style property to modify CSS in JS

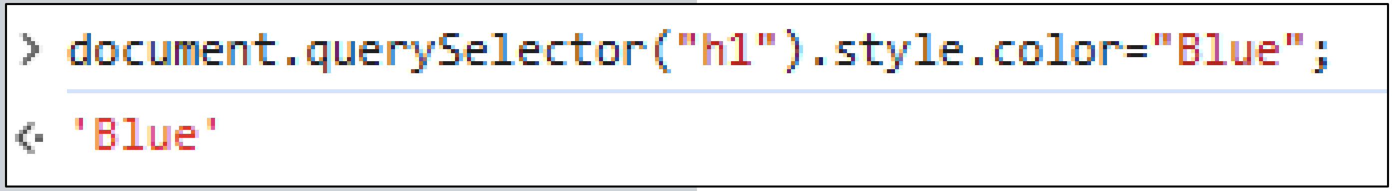
Changing CSS with JavaScript :Changing Text Color

```
<body>
    <h1 class="heading">Hello</h1>
     <input class="check" type="checkbox">
    <button id="click" class="btn">Click here</button><br>
    <ul class="list">
      <li id="google" class="list item">Google</li>
      <li id="uni" class="list item">COMSATS</li>
      <li id="department" class="list item">Comp: Eng:</li>
    </ul>
 </body>
```





- Direct manipulation of selected element



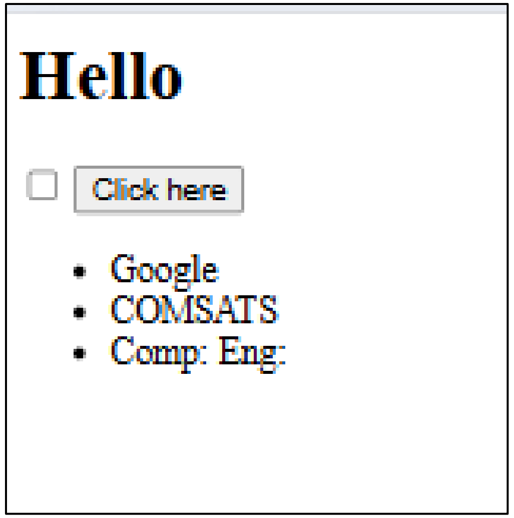
```
> document.querySelector("h1").style.color="Blue";
<· 'Blue'
```

Camel Case in JavaScript

- CSS uses: font-size, background-color
- JS uses: fontSize, backgroundColor
- Rule: remove dashes + capitalize next word

Changing CSS with JavaScript :CSS Values in JavaScript

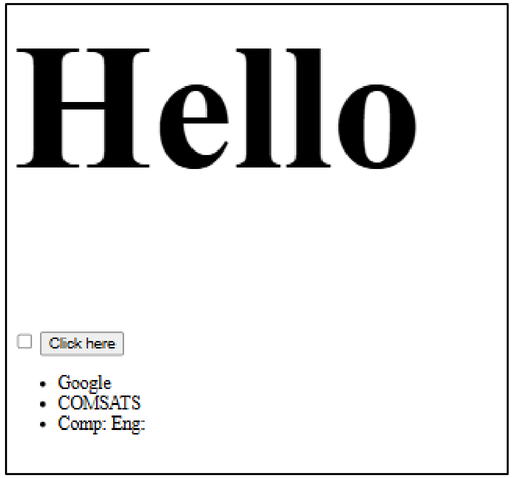
```
<body>
    <h1 class="heading">Hello</h1>
     <input class="check" type="checkbox">
    <button id="click" class="btn">Click here</button><br>
    <ul class="list">
      <li id="google" class="list item">Google</li>
      <li id="uni" class="list item">COMSATS</li>
      <li id="department" class="list item">Comp: Eng:</li>
    </ul>
 </body>
```



Hello

☐ Click here

- Google
- COMSATS
- Comp: Eng:

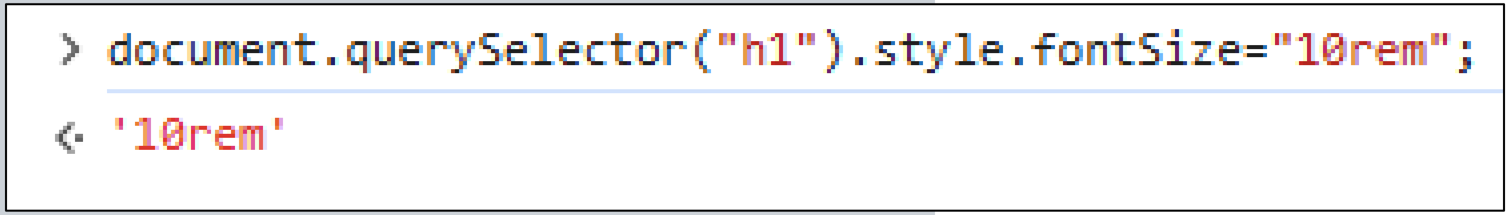


Hello

☐ Click here

- Google
- COMSATS
- Comp: Eng:

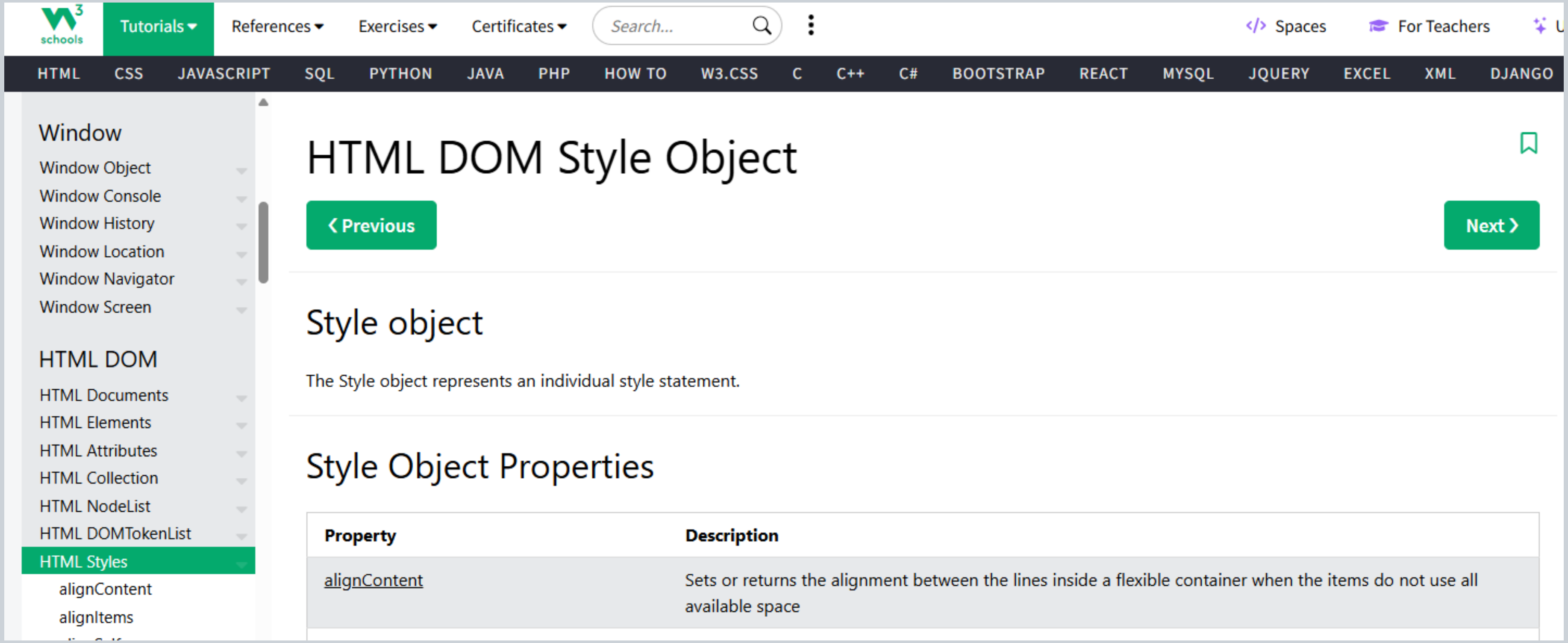
- Values must be strings: '10rem', 'hidden', '7%'



```
> document.querySelector("h1").style.fontSize="10rem";
<· '10rem'
```

Style Properties Documentation

- DOM Style object lists all supported properties
- https://www.w3schools.com/jsref/dom_obj_style.asp
- Names follow camelCase format



HTML DOM Style Object

Style object

The Style object represents an individual style statement.

Style Object Properties

Property	Description
alignContent	Sets or returns the alignment between the lines inside a flexible container when the items do not use all available space

Practice Task: Change Button Background

- Use querySelector to grab button
- Use style.backgroundColor property
- Try modifying more CSS properties with JS
- Explore Developer Tools for immediate feedback

Separation of Concerns

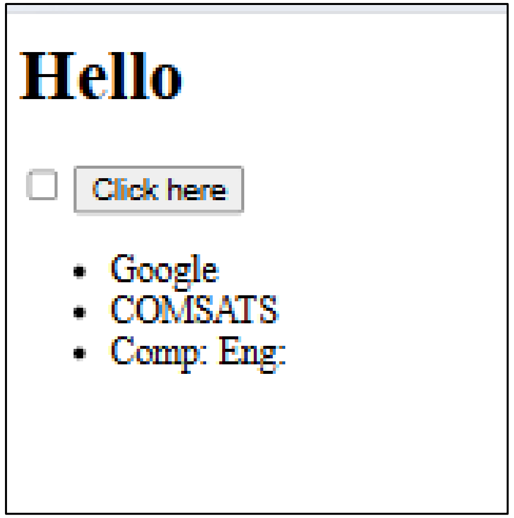
Structure vs Style vs Behavior

Separation of Concerns

- HTML → Content
- CSS → Style
- JavaScript → Behavior
- Keep responsibilities separated for clean + maintainable code
- Bad Practice:
  - document.querySelector("h1").style.color = "red";
  - JS should not directly style elements
  - Styles belong in CSS
- Better Approach: classList
  - Every DOM element has a classList property
  - We can add / remove / toggle CSS classes dynamically

classList Methods

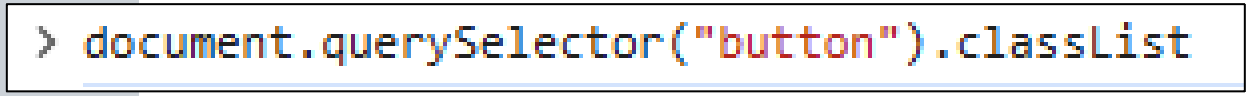
```
<body>
     <h1 class="heading">Hello</h1>
      <input class="check" type="checkbox">
     <button id="click" class="btn">Click here</button><br>
     <ul class="list">
        <li id="google" class="list item">Google</li>
        <li id="uni" class="list item">COMSATS</li>
        <li id="department" class="list item">Comp: Eng:</li>
     </ul>
 </body>
```



Hello

□ Click here

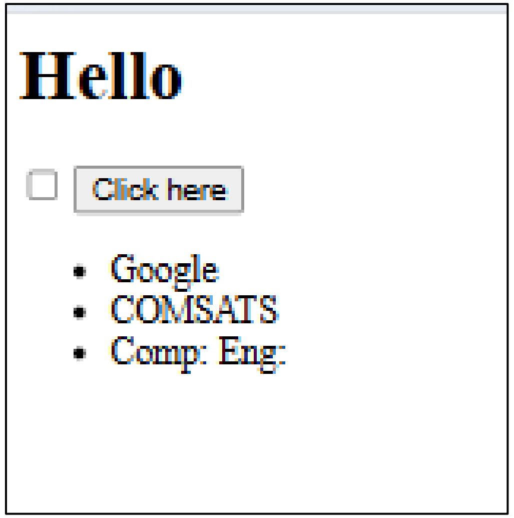
- Google
- COMSATS
- Comp: Eng:



```
> document.querySelector("button").classList
```

classList Methods

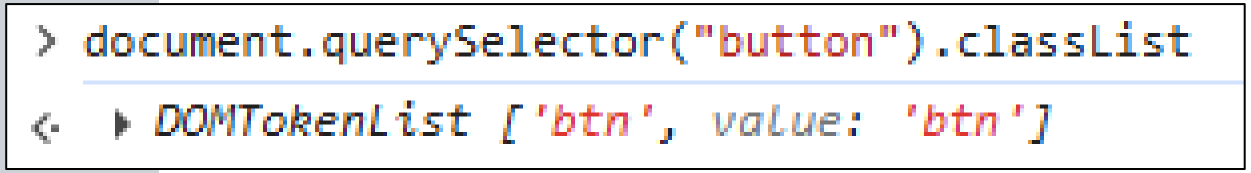
```
<body>
    <h1 class="heading">Hello</h1>
     <input class="check" type="checkbox">
    <button id="click" class="btn">Click here</button><br>
    <ul class="list">
      <li id="google" class="list item">Google</li>
      <li id="uni" class="list item">COMSATS</li>
      <li id="department" class="list item">Comp: Eng:</li>
    </ul>
 </body>
```



Hello

□ Click here

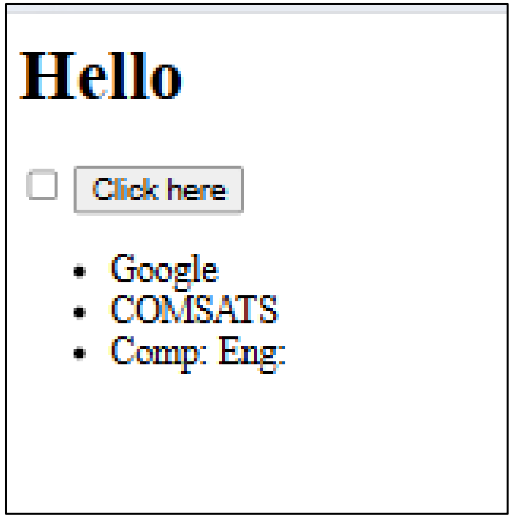
- Google
- COMSATS
- Comp: Eng:



```
> document.querySelector("button").classList
<·  ▸ DOMTokenList ['btn', value: 'btn']
```

classList Methods

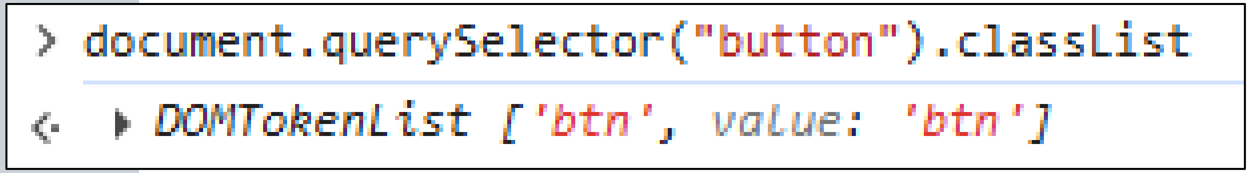
```
<body>
      <h1 class="heading">Hello</h1>
       <input class="check" type="checkbox">
      <button id="click" class="btn">Click here</button><br>
      <ul class="list">
         <li id="google" class="list item">Google</li>
         <li id="uni" class="list item">COMSATS</li>
         <li id="department" class="list item">Comp: Eng:</li>
      </ul>
 </body>
```



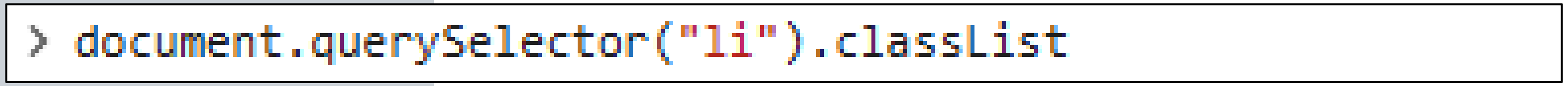
Hello

□ Click here

- Google
- COMSATS
- Comp: Eng:



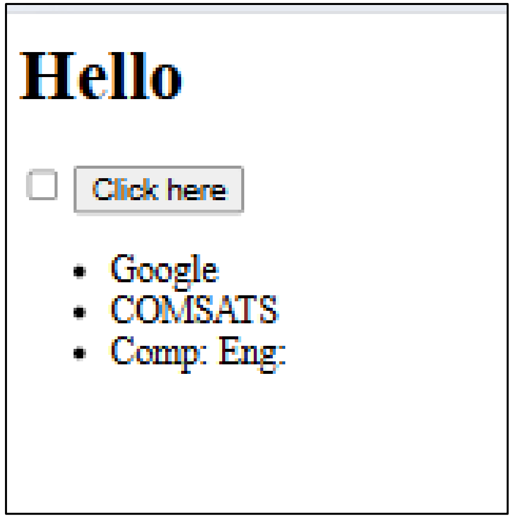
```
> document.querySelector("button").classList
<· ▸ DOMTokenList ['btn', value: 'btn']
```



```
> document.querySelector("li").classList
```

classList Methods

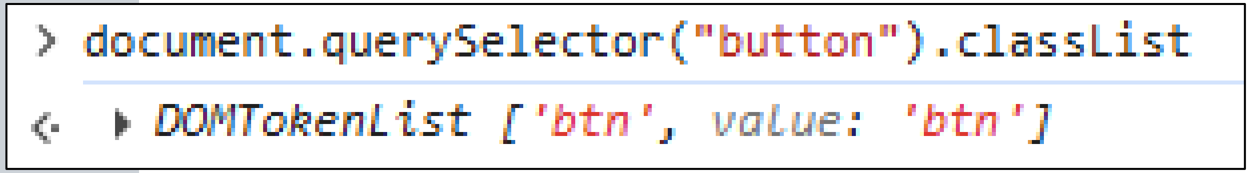
```
<body>
      <h1 class="heading">Hello</h1>
       <input class="check" type="checkbox">
      <button id="click" class="btn">Click here</button><br>
      <ul class="list">
         <li id="google" class="list item">Google</li>
         <li id="uni" class="list item">COMSATS</li>
         <li id="department" class="list item">Comp: Eng:</li>
      </ul>
 </body>
```



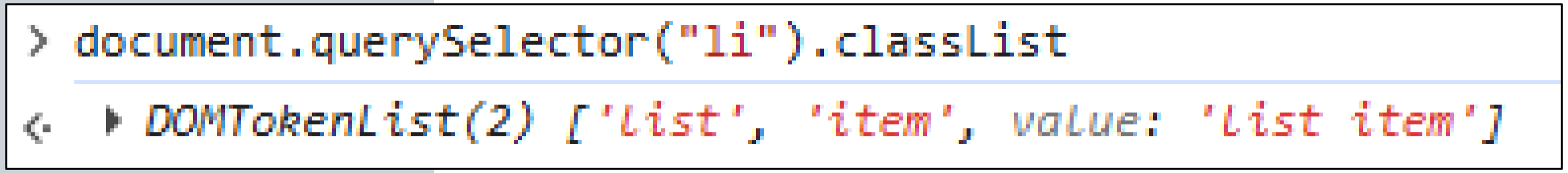
Hello

☐ Click here

- Google
- COMSATS
- Comp: Eng:



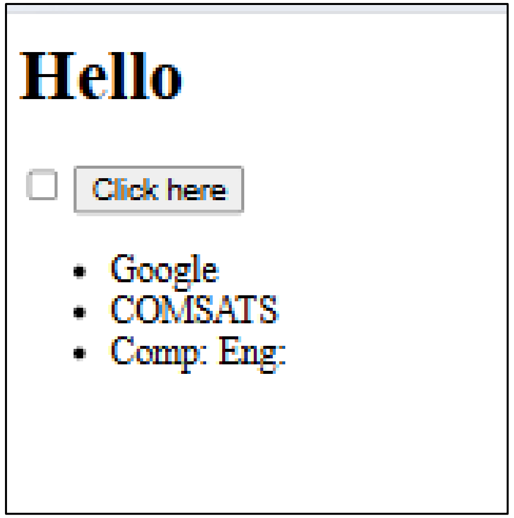
```
> document.querySelector("button").classList
<· ▸ DOMTokenList ['btn', value: 'btn']
```



```
> document.querySelector("li").classList
<· ▸ DOMTokenList(2) ['list', 'item', value: 'list item']
```

classList Methods : .add()

```
<body>
    <h1 class="heading">Hello</h1>
     <input class="check" type="checkbox">
    <button id="click" class="btn">Click here</button><br>
    <ul class="list">
      <li id="google" class="list item">Google</li>
      <li id="uni" class="list item">COMSATS</li>
      <li id="department" class="list item">Comp: Eng:</li>
    </ul>
 </body>
```

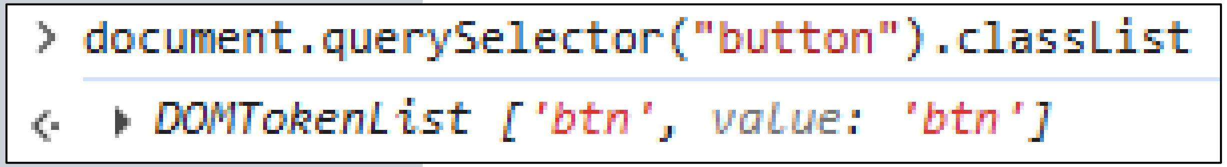


Hello

☐ Click here

- Google
- COMSATS
- Comp: Eng:

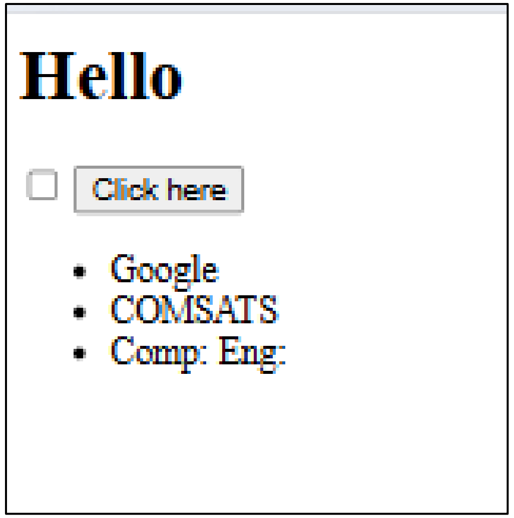
```
.invisible{
  visibility: hidden;
}
```



```
> document.querySelector("button").classList
<· ▸ DOMTokenList ['btn', value: 'btn']
```

classList Methods : .add()

```
<body>
    <h1 class="heading">Hello</h1>
     <input class="check" type="checkbox">
    <button id="click" class="btn">Click here</button><br>
    <ul class="list">
      <li id="google" class="list item">Google</li>
      <li id="uni" class="list item">COMSATS</li>
      <li id="department" class="list item">Comp: Eng:</li>
    </ul>
 </body>
```

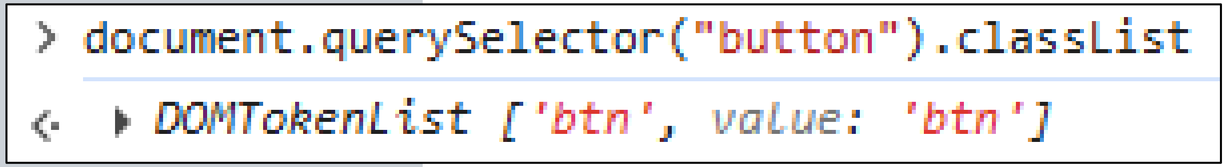


Hello

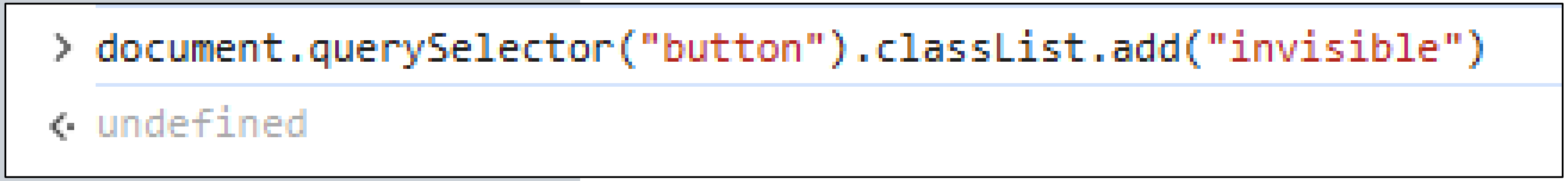
□

- Google
- COMSATS
- Comp: Eng:

```
.invisible{
   visibility: hidden;
}
```



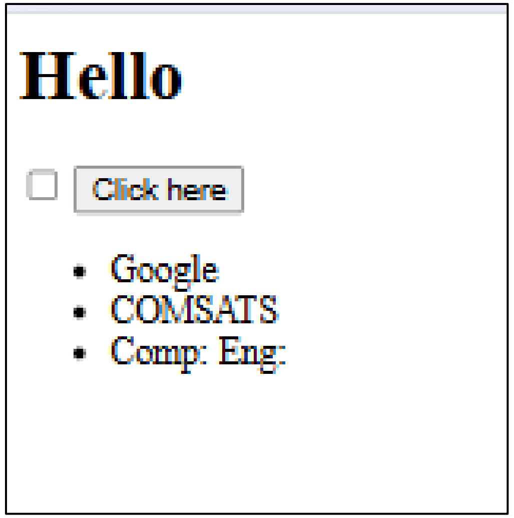
```
> document.querySelector("button").classList
<· ▸ DOMTokenList ['btn', value: 'btn']
```



```
> document.querySelector("button").classList.add("invisible")
<· undefined
```

classList Methods : .add()

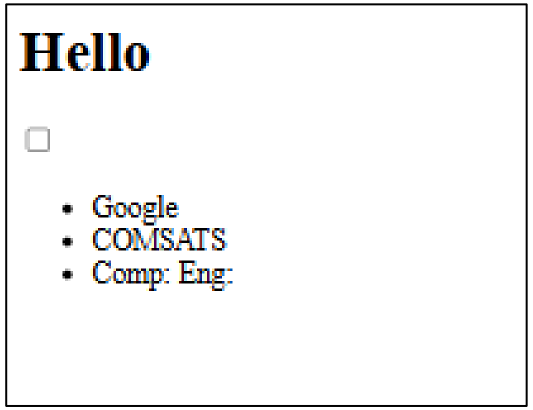
```
<body>
    <h1 class="heading">Hello</h1>
     <input class="check" type="checkbox">
    <button id="click" class="btn">Click here</button><br>
    <ul class="list">
      <li id="google" class="list item">Google</li>
      <li id="uni" class="list item">COMSATS</li>
      <li id="department" class="list item">Comp: Eng:</li>
    </ul>
 </body>
```



Hello

□ Click here

- Google
- COMSATS
- Comp: Eng:

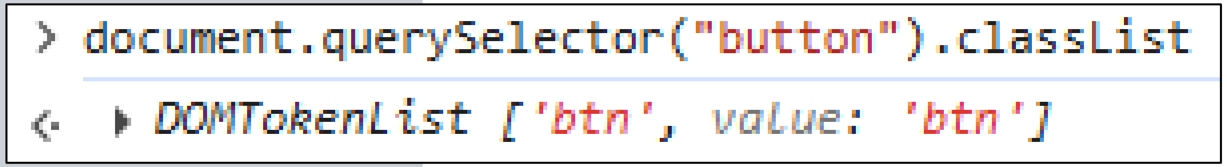


Hello

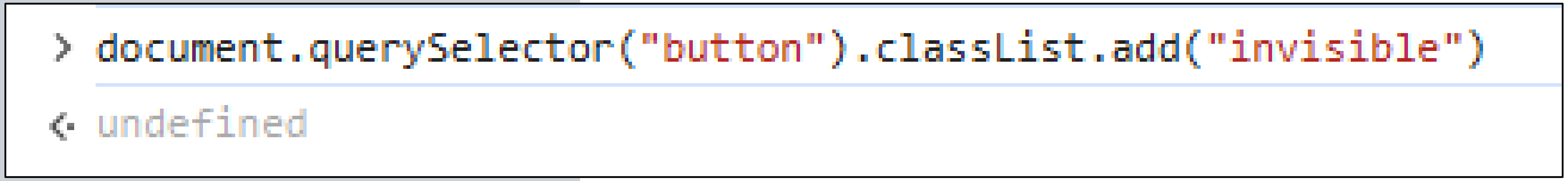
□

- Google
- COMSATS
- Comp: Eng:

```
.invisible{
  visibility: hidden;
}
```



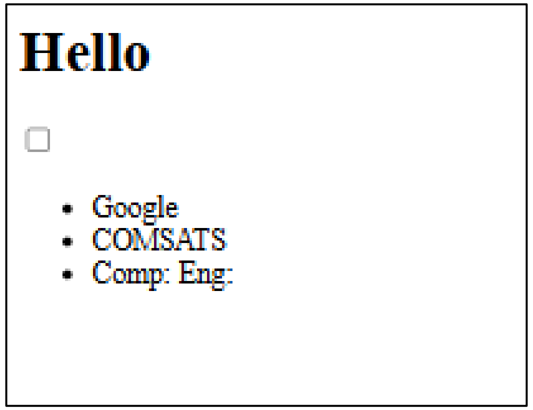
```
> document.querySelector("button").classList
<· ▸ DOMTokenList ['btn', value: 'btn']
```



```
> document.querySelector("button").classList.add("invisible")
<· undefined
```

classList Methods : .remove()

```
<body>
     <h1 class="heading">Hello</h1>
      <input class="check" type="checkbox">
     <button id="click" class="btn">Click here</button><br>
     <ul class="list">
        <li id="google" class="list item">Google</li>
        <li id="uni" class="list item">COMSATS</li>
        <li id="department" class="list item">Comp: Eng:</li>
     </ul>
 </body>
```

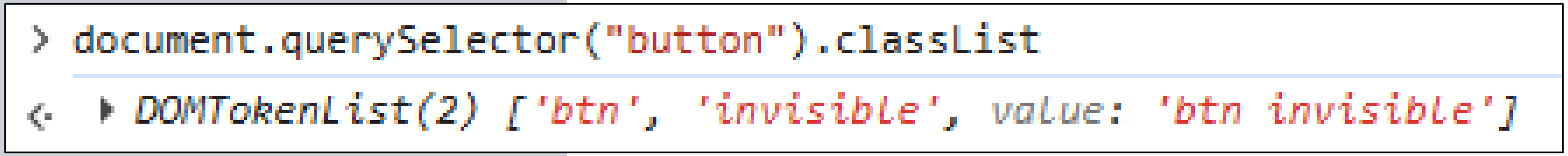


Hello

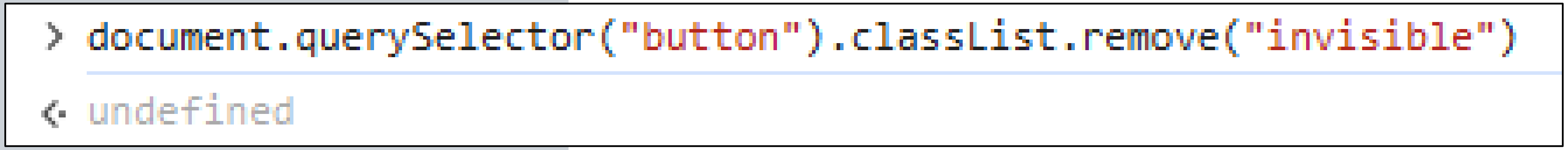
□

- Google
- COMSATS
- Comp: Eng:

```
.invisible{
   visibility: hidden;
}
```



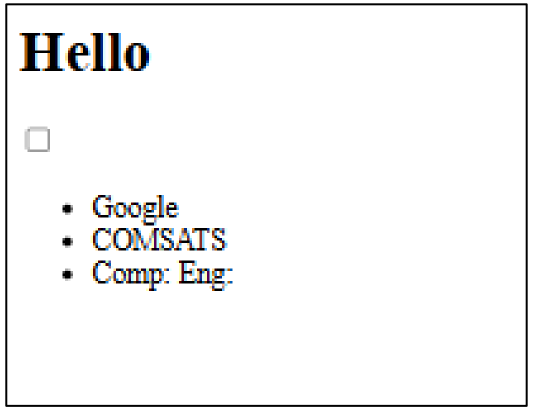
```
> document.querySelector("button").classList
<· ▸ DOMTokenList(2) ['btn', 'invisible', value: 'btn invisible']
```



```
> document.querySelector("button").classList.remove("invisible")
<· undefined
```

classList Methods : .remove()

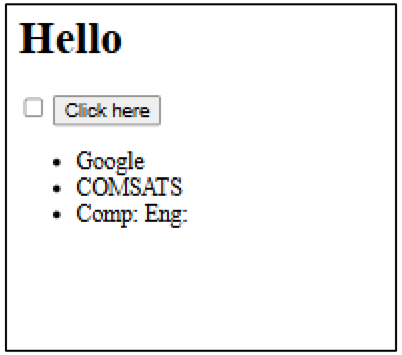
```
<body>
     <h1 class="heading">Hello</h1>
      <input class="check" type="checkbox">
     <button id="click" class="btn">Click here</button><br>
     <ul class="list">
        <li id="google" class="list item">Google</li>
        <li id="uni" class="list item">COMSATS</li>
        <li id="department" class="list item">Comp: Eng:</li>
     </ul>
 </body>
```



Hello

□

- Google
- COMSATS
- Comp: Eng:

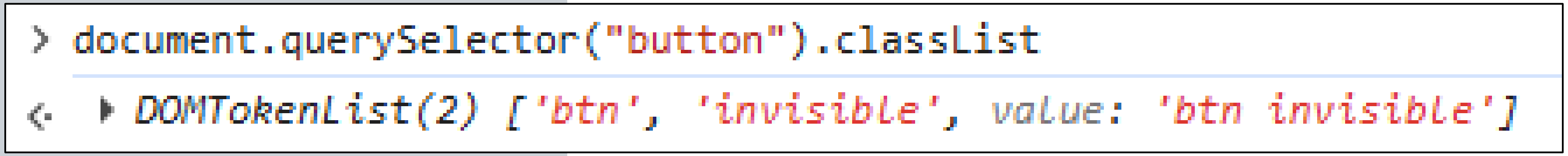


Hello

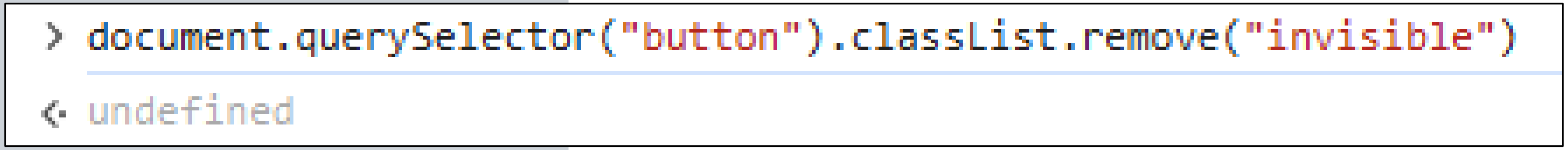
□

- Google
- COMSATS
- Comp: Eng:

```
.invisible{
   visibility: hidden;
}
```



```
> document.querySelector("button").classList
<· ▸ DOMTokenList(2) ['btn', 'invisible', value: 'btn invisible']
```

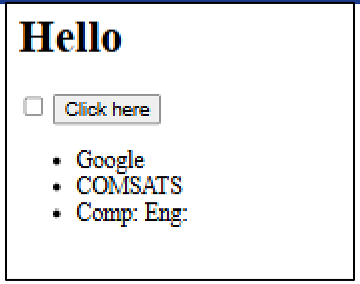


```
> document.querySelector("button").classList.remove("invisible")
<· undefined
```

classList Methods : .toggle()

```
<body>
    <h1 class="heading">Hello</h1>
     <input class="check" type="checkbox">
    <button id="click" class="btn">Click here</button><br>
    <ul class="list">
      <li id="google" class="list item">Google</li>
      <li id="uni" class="list item">COMSATS</li>
      <li id="department" class="list item">Comp: Eng:</li>
    </ul>
 </body>
```

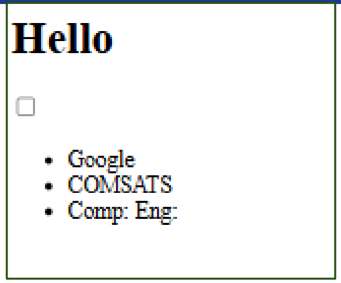
```
.invisible{
   visibility: hidden;
}
```



Hello

□ Click here

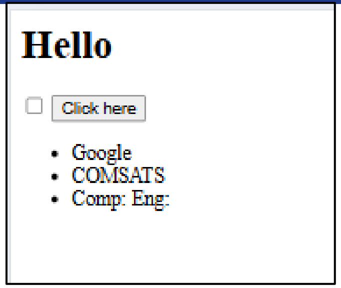
- Google
- COMSATS
- Comp: Eng:



Hello

□

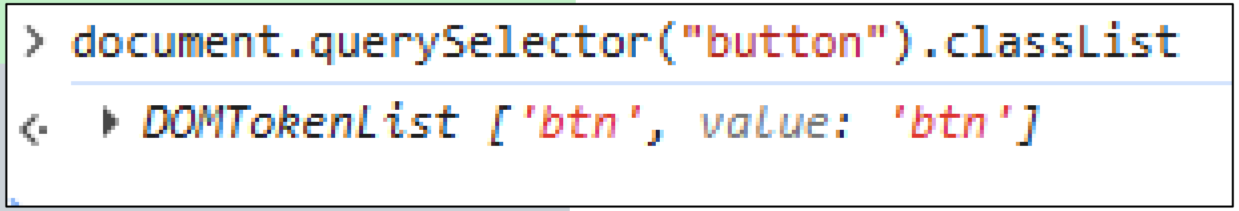
- Google
- COMSATS
- Comp: Eng:



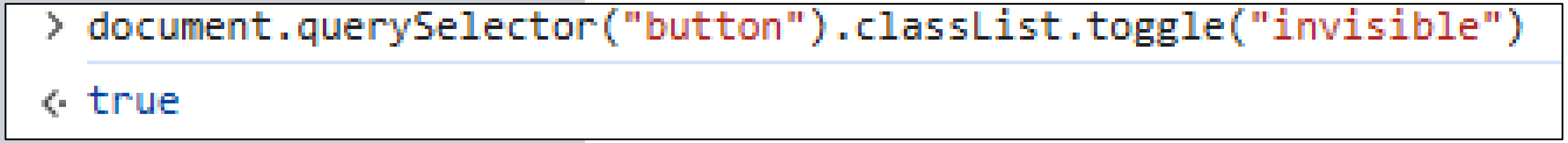
Hello

□ Click here

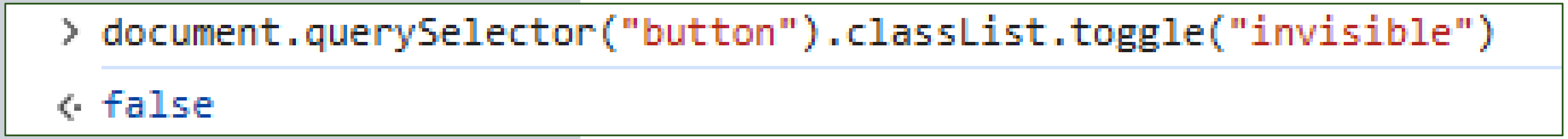
- Google
- COMSATS
- Comp: Eng:



```
> document.querySelector("button").classList
<· ▸ DOMTokenList ['btn', value: 'btn']
```



```
> document.querySelector("button").classList.toggle("invisible")
<· true
```



```
> document.querySelector("button").classList.toggle("invisible")
<· false
```

Practice Problem:

- CSS: Create 'huge' Class .huge → font-size: 10rem; color: red; font-weight: bold;
- JS: document.querySelector('h1').classList.add('huge');

Benefits of This Method

- Keeps styling in CSS for easier debugging
- JS only handles behavior
- More scalable + cleaner code design

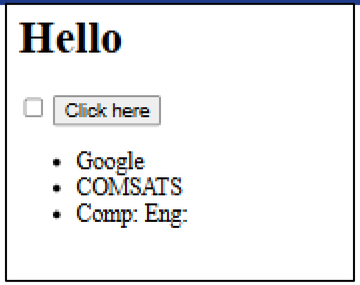
innerHTML vs textContent in JavaScript

We can change text inside selected HTML elements using:

- innerHTML
- textContent

innerHTML vs textContent

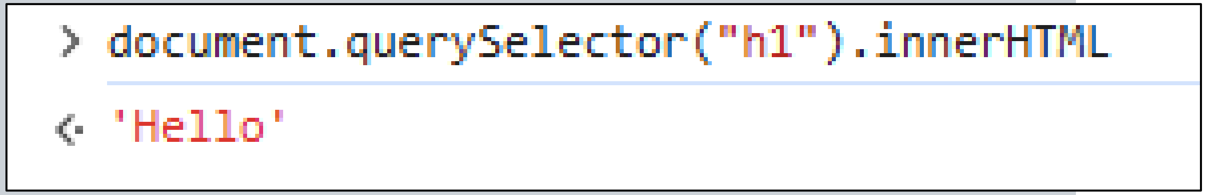
```
<body>
    <h1 class="heading">Hello</h1>
     <input class="check" type="checkbox">
    <button id="click" class="btn">Click here</button><br>
    <ul class="list">
      <li id="google" class="list item">Google</li>
      <li id="uni" class="list item">COMSATS</li>
      <li id="department" class="list item">Comp: Eng:</li>
    </ul>
 </body>
```



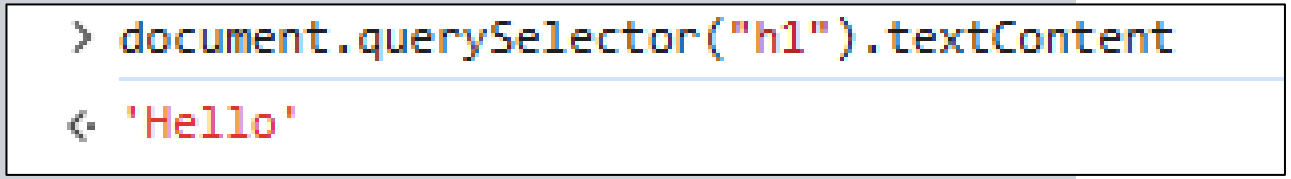
Hello

□ Click here

- Google
- COMSATS
- Comp: Eng:



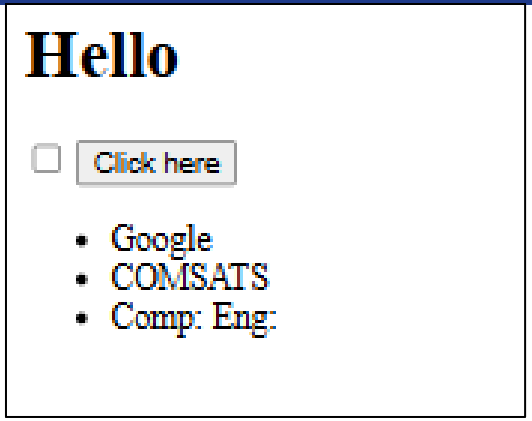
```
> document.querySelector("h1").innerHTML
<· 'Hello'
```



```
> document.querySelector("h1").textContent
<· 'Hello'
```

innerHTML vs textContent

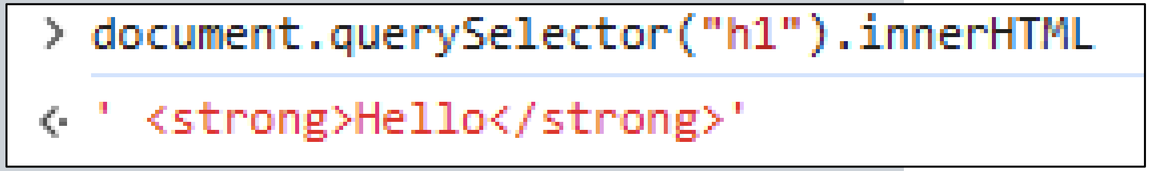
```
<body>
    <h1 > <strong>Hello</strong></h1>
     <input type="checkbox">
    <button>Click here</button><br>
    <ul >
      <li >Google</li>
      <li >COMSATS</li>
      <li >Comp: Eng:</li>
    </ul>
  </body>
```



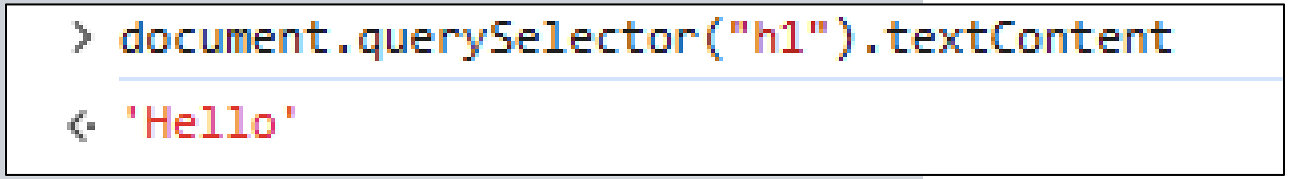
Hello

□ Click here

- Google
- COMSATS
- Comp: Eng:



```
> document.querySelector("h1").innerHTML
<· ' <strong>Hello</strong>'
```



```
> document.querySelector("h1").textContent
<· 'Hello'
```

innerHTML vs textContent

- **innerHTML**
  - Gets or sets HTML + text
  - Can insert elements, formatting, and tags
  - Be careful: can run unwanted HTML/JS if not safe
- **textContent**
  - Gets or sets plain text only
  - Ignores any HTML tags
  - Safer for user input and readable content.

Manipulating HTML Attributes with JavaScript

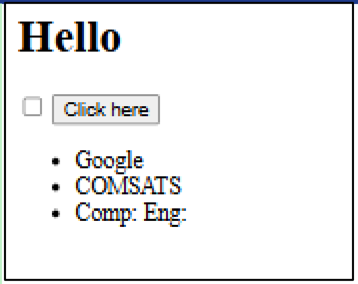
DOM Attribute Access: getAttribute() & setAttribute()

What Are Attributes?

- Attributes provide additional information about HTML elements:
  - Class, href, src, id
- Everything inside the tag besides the tag name is an attribute.

Manipulating HTML Attributes with JavaScript

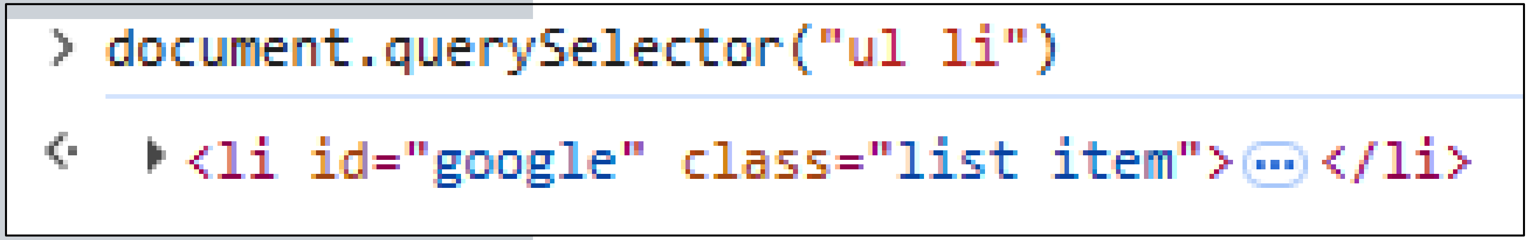
```
<body>
     <h1 class="heading">Hello</h1>
      <input class="check" type="checkbox">
     <button id="click" class="btn">Click here</button><br>
     <ul class="list">
        <li id="google" class="list item">Google</li>
        <li id="uni" class="list item">COMSATS</li>
        <li id="department" class="list item">Comp: Eng:</li>
     </ul>
 </body>
```



Hello

□ Click here

- Google
- COMSATS
- Comp: Eng:

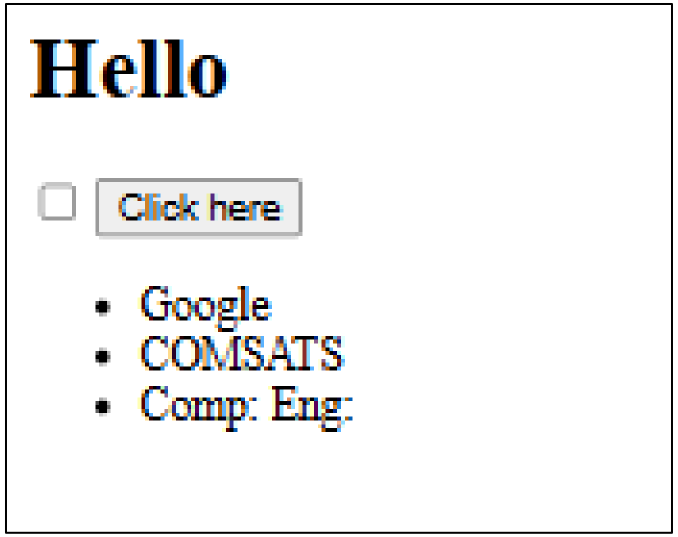


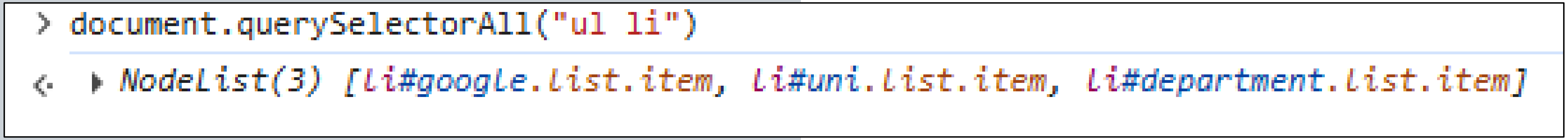
```
> document.querySelector("ul li")
<· ▸<li id="google" class="list item">…</li>
```

- Why only one list item is selected?

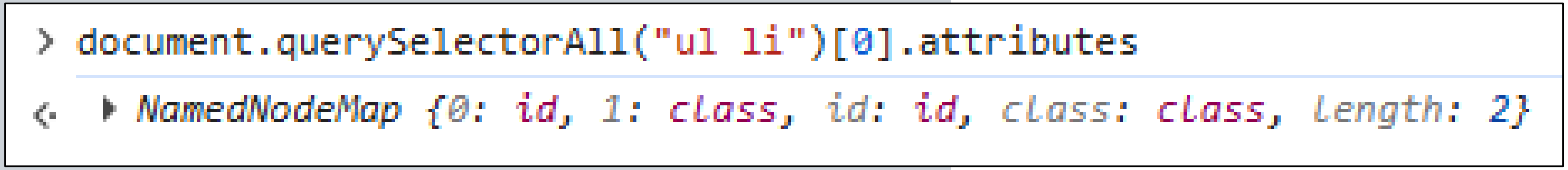
viewing Attributes

```
<body>
    <h1 class="heading">Hello</h1>
     <input class="check" type="checkbox">
    <button id="click" class="btn">Click here</button><br>
    <ul class="list">
      <li id="google" class="list item">Google</li>
      <li id="uni" class="list item">COMSATS</li>
      <li id="department" class="list item">Comp: Eng:</li>
    </ul>
 </body>
```





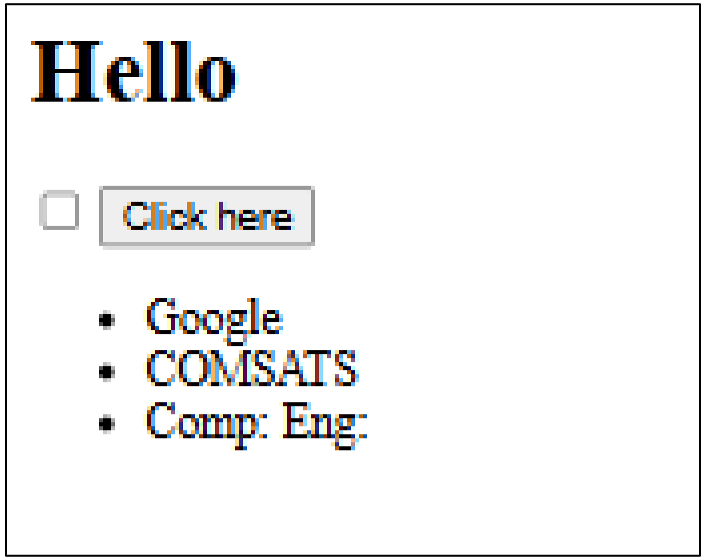
```
> document.querySelectorAll("ul li")
<· ▸ NodeList(3) [li#google.list.item, li#uni.list.item, li#department.list.item]
```



```
> document.querySelectorAll("ul li")[0].attributes
<· ▸ NamedNodeMap {0: id, 1: class, id: id, class: class, length: 2}
```

Get a Single Attribute Value

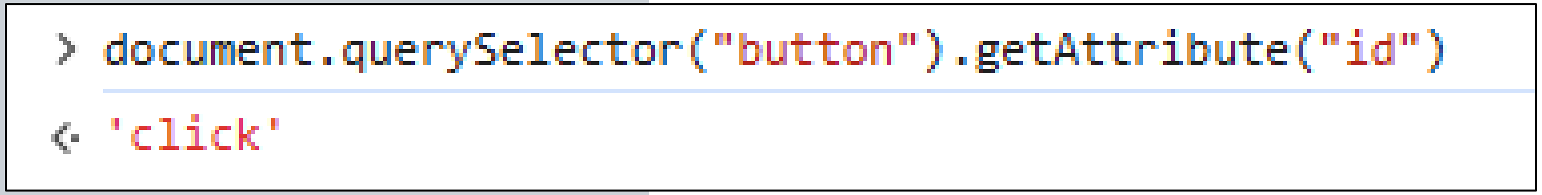
```
<body>
    <h1 class="heading">Hello</h1>
     <input class="check" type="checkbox">
    <button id="click" class="btn">Click here</button><br>
    <ul class="list">
      <li id="google" class="list item">Google</li>
      <li id="uni" class="list item">COMSATS</li>
      <li id="department" class="list item">Comp: Eng:</li>
    </ul>
 </body>
```



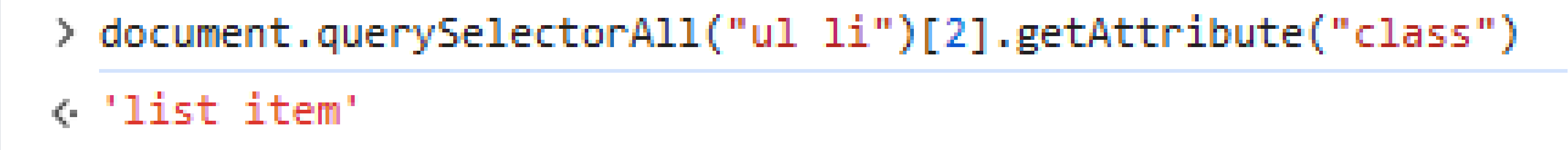
Hello

☐ Click here

- Google
- COMSATS
- Comp: Eng:



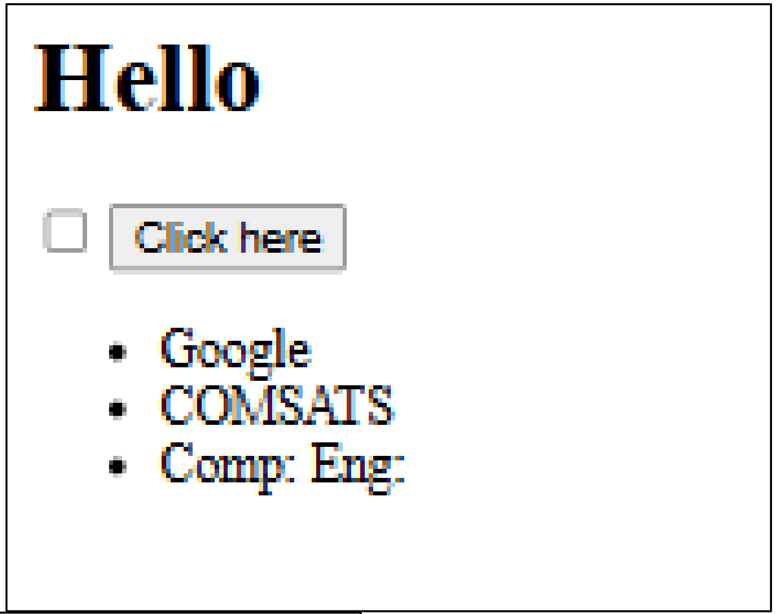
```
> document.querySelector("button").getAttribute("id")
<· 'click'
```

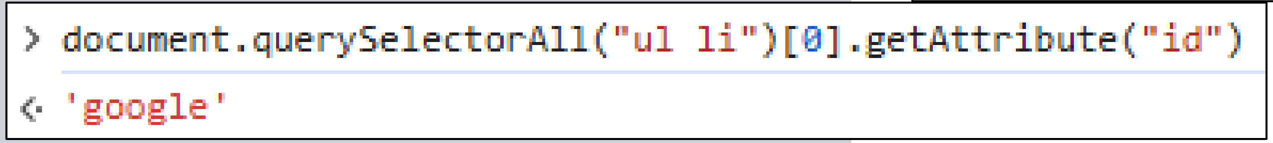


```
> document.querySelectorAll("ul li")[2].getAttribute("class")
<· 'list item'
```

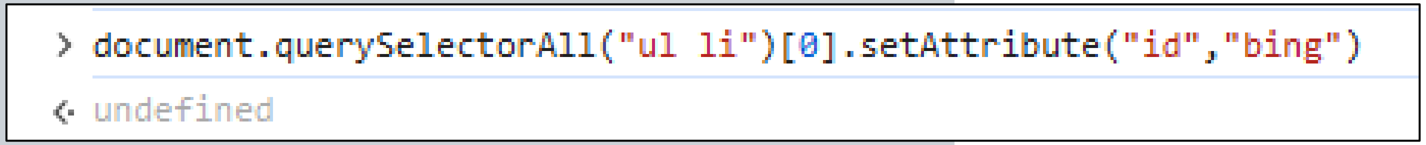
Set or Change Attribute Value

```
<body>
      <h1 class="heading">Hello</h1>
       <input class="check" type="checkbox">
      <button id="click" class="btn">Click here</button><br>
      <ul class="list">
         <li id="google" class="list item">Google</li>
         <li id="uni" class="list item">COMSATS</li>
         <li id="department" class="list item">Comp: Eng:</li>
      </ul>
 </body>
```

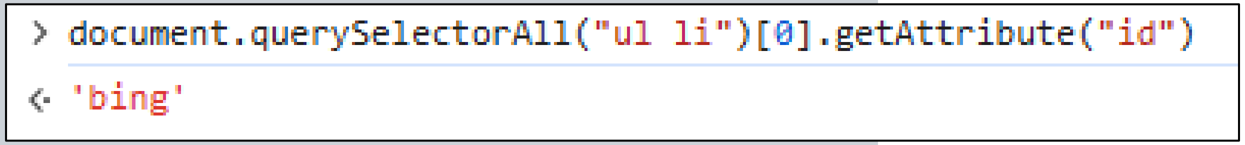




```
> document.querySelectorAll("ul li")[0].getAttribute("id")
<· 'google'
```



```
> document.querySelectorAll("ul li")[0].setAttribute("id","bing")
<· undefined
```



```
> document.querySelectorAll("ul li")[0].getAttribute("id")
<· 'bing'
```

Summary

- attributes → list of attribute objects
- getAttribute(name) → read value
- setAttribute(name, value) → update value
- Useful for links, images, classes, IDs, etc.