



SWCON104
Web & Python Programming

JavaScript

Department of Software Convergence

Today

- Static web pages
 - Dynamic web pages
 - Why do we use Javascript to create web pages?
-
- Write a simple Javascript program
 - Practice Javascript syntax
 - Alert and prompts
 - Events
 - Gathering user input
 - Dynamically modify HTML

Practice

- Practice_19_JavaScript

Static web pages

Experimenting with Links and Text

Department of Software Convergence

Practicing HTML is fun.

So far we have learned **HTML** is a language made of tags that help us describe how we want our webpages to look. We have tried the *head*, *title*, *H(heading)*, and paragraph tags so far.

Very few people know all the tags in HTML, but that is okay because we can do a lot with just some basic tags. We can learn about new tags when we need them.

Dynamic web pages

- Server-side dynamic web page

오늘의 발견 | 오늘 쿠팡이 엄선한 가장 HOT한 상품!

식품

행복촌
캘리포니아 생체리

10,400원

>

여성의류

오델리 에스닉 와이드
롱 치마바지

27,000원

>

디지털

오아아이브릭
블루투스스피커

19,900원

>

잡화

썸머 왕골
빅백 14색

4,900원

>

유아동

바니바니
폼폼 원피스

7,900원

>

유아동

라인하우스
3D메쉬 낮잠 베개

13,900원

>

Dynamic web pages: button example

- Client-side dynamic web page → Generally use Javascript!



Button example: source code

The diagram illustrates the state transition of a button click. It shows two states of a rectangular container divided by a vertical line. In the first state, the left side contains the text "Let's see a button" and the right side contains a "Click Me" button. In the second state, after a click, the right side now displays the text "Thanks for adding text!".

html5.html

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>My Dynamic Details Page</title>
5
6      <script type="text/javascript">
7
8          function addParagraphText()
9          {
10              document.getElementById("para").innerHTML = "Thanks for adding text!";
11          }
12      </script>
13
14  </head>
15
16  <body>
17
18      <H1>Let's see a button</H1>
19
20      <button onclick="addParagraphText();>Click Me</button>
21
22      <p id="para"></p>
23
24  </body>
25
26  </html>
```

Button example: source code

```
html5.html x
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>My Dynamic Details Page</title>
5
6      <script type="text/javascript">
7
8          function addParagraphText()
9          {
10              document.getElementById("para").innerHTML = "Thanks for adding text!";
11          }
12      </script>
13
14  </head>
15
16  <body>
17
18      <H1>Let's see a button</H1>
19
20      <button onclick="addParagraphText();>Click Me</button>
21
22      <p id="para"></p>
23
24  </body>
25
26  </html>
27
```

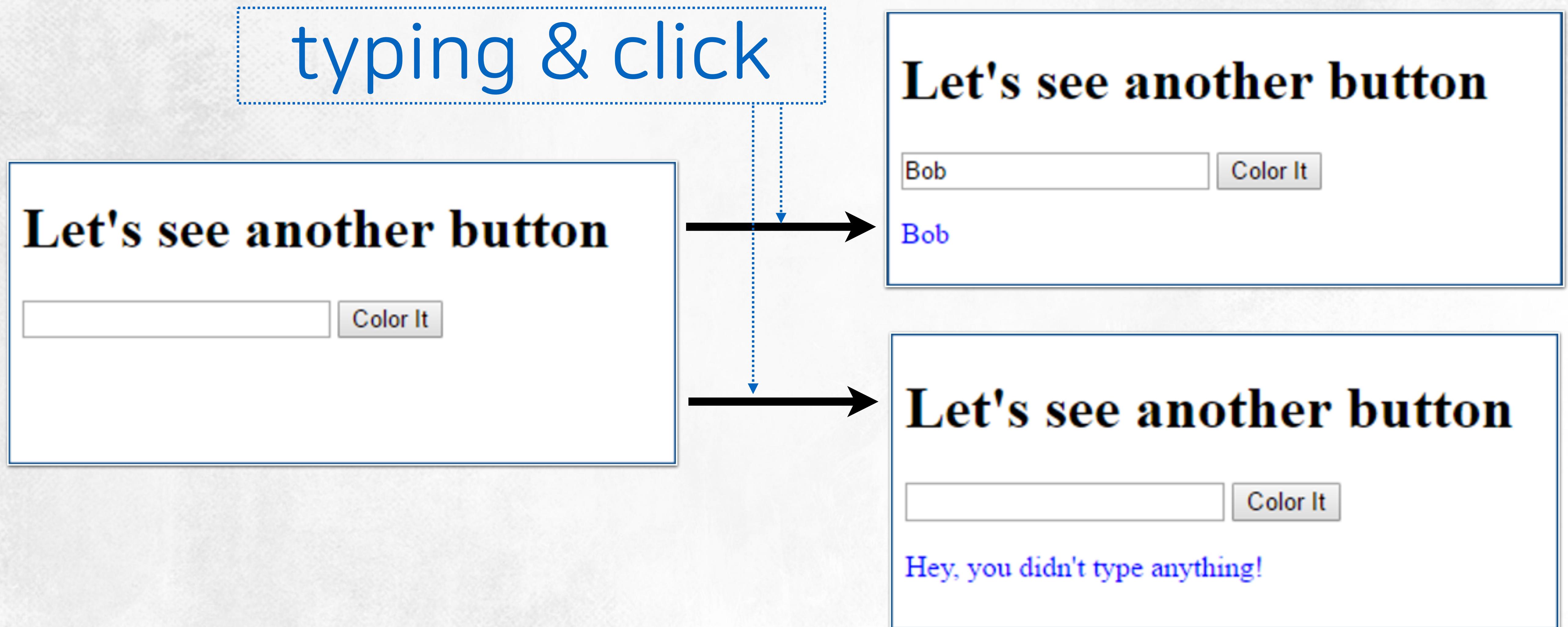
onclick = an event

addParagraphText() = a program (function) to execute

“para” = the id that we’re supplying

innerHTML = “Thanks for adding text!”

User input example



The value of JavaScript

- HTML is static



- We want our website to be responsive

- User takes an action
- Respond with action
- We can't do that with plain text HTML

Responsive pages

- HTML is not a programming language
- HTML is a markup language that uses tags

- If we use a programming language
 - Take steps and actions in a sequence
 - Do things repetitively
 - Make decisions based on data

- JavaScript is a programming language
 - We can make interactive pages
 - We can interact with users/ visitors

Button example: again

```
html5.html
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>My Dynamic Details Page</title>
5
6   <script type="text/javascript">
7     function addParagraphText()
8     {
9       document.getElementById("para").innerHTML = "Thanks for adding text!";
10      }
11
12 </script>
13
14 </head>
15
16 <body>
17
18 <H1>Let's see a button</H1>
19
20 <button onclick="addParagraphText();>Click Me</button>
21
22 <p id="para"></p>
23
24 </body>
25
26 </html>
```

JavaScript

JavaScript's assignment operator

JavaScript ends with a semicolon

JavaScript

Button example: again

```
html5.html
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>My Dynamic Details Page</title>
5
6      <script type="text/javascript">
7
8          function addParagraphText()
9          {
10              document.getElementById("para").innerHTML = "Thanks for adding text!";
11          }
12      </script>
13
14  </head>
15
16  <body>
17
18      <H1>Let's see a button</H1>
19
20      <button onclick="addParagraphText();">Click Me</button>
21
22      <p id="para"></p>
23
24  </body>
25
26  </html>
```

Get an element which has an id =“para”

Assign the string to the element’s innerHTML

Element

User input example

- var
- function { }
- if (condition){ }
- else { }
- input
- Semicolon;
- Colon: not used

html6.html

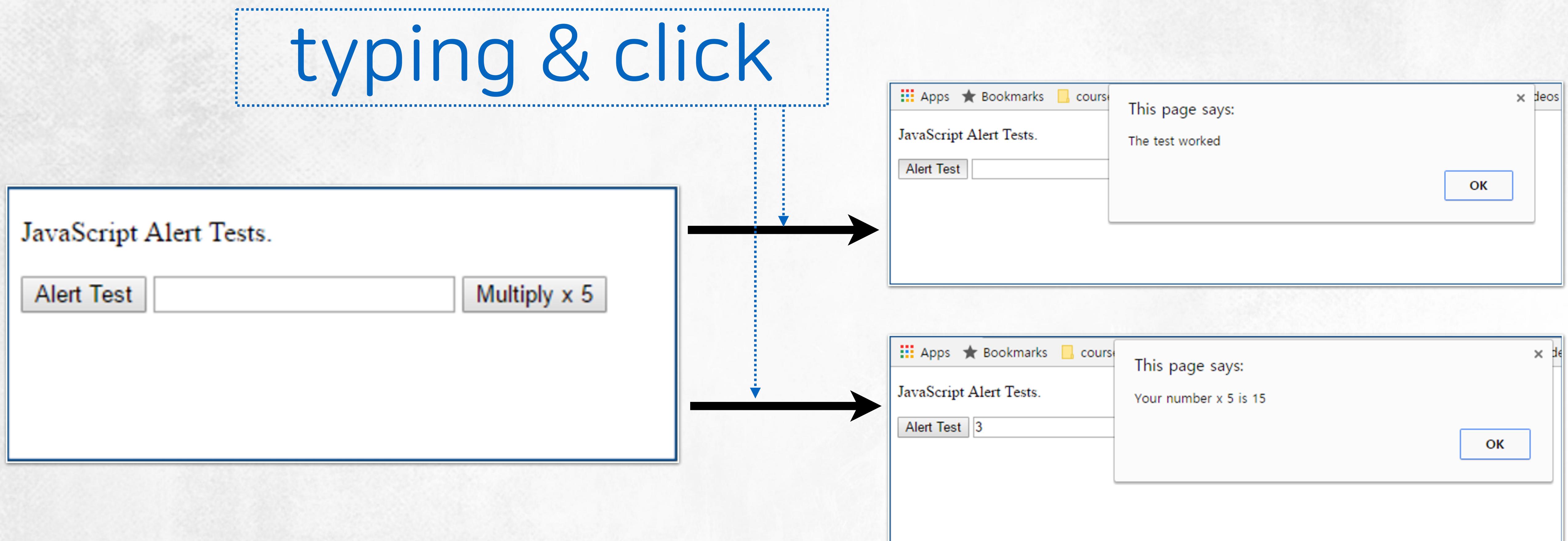
```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>My Dynamic Page</title>
5
6   <script type="text/javascript">
7
8     function displayInput() // Function will display a blue version of input.
9     {
10       var testInput = document.getElementById("name").value;
11
12       if (testInput.length == 0)
13       {
14         document.getElementById("para").innerHTML = "Hey, you didn't type anything!";
15       }
16       else
17       {
18         document.getElementById("para").innerHTML = testInput;
19       }
20     }
21   </script>
22
23 </head>
24
25 <body>
26
27   <H1 style="color:blue">Let's see another button</H1>
28
29   <input id="name" type="text"/>
30
31   <button onclick="displayInput();">Color It</button>
32
33   <p id="para" style="color:blue;"></p>
34
35 </body>
36
37 </html>
38
```

Let's see another button

Color It

Hey, you didn't type anything!

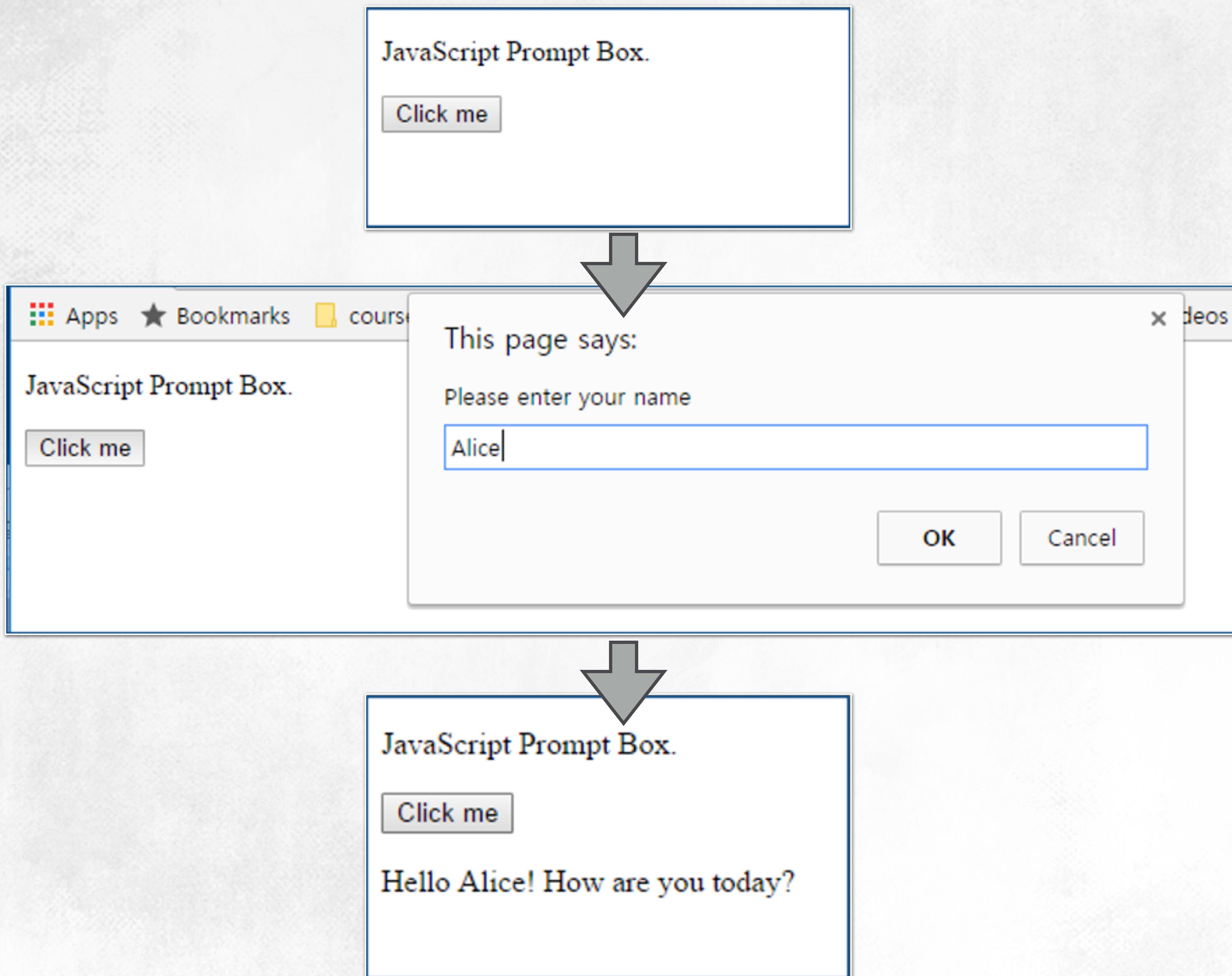
Alert example



Alert example: source code

```
html1.html x html2.html x html3.html x html4.html x javaScriptAlert.html x
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>Alert Page</title>
5
6      <script>
7
8          function myAlertMath()
9          {
10              var userNumber = document.getElementById("mathInput").value;
11
12              alert("Your number x 5 is " + userNumber * 5);
13          }
14      </script>
15
16  </head>
17
18  <body>
19
20      <p>JavaScript Alert Tests.</p>
21
22      <button onclick="alert('The test worked');">Alert Test</button>
23
24      <input id="mathInput" type="number"/> <button onclick="myAlertMath()">Multiply x 5</button>
25
26  </body>
27
28  </html>
29 |
```

Prompt example



Prompt example: source code

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>Prompt Page</title>
5
6      <script>
7
8          function myPrompt()
9          {
10              var person = prompt("Please enter your name"); // The second string optional
11              if (person != "")
12              {
13                  document.getElementById("test").innerHTML =
14                      "Hello " + person + "! How are you today?";
15              }
16          }
17      </script>
18
19  </head>
20
21  <body>
22
23      <p>JavaScript Prompt Box.</p>
24
25      <button onclick="myPrompt();">Click me</button>
26
27      <p id="test"></p>
28
29  </body>
30
31  </html>
32
```

Prompt example: default value

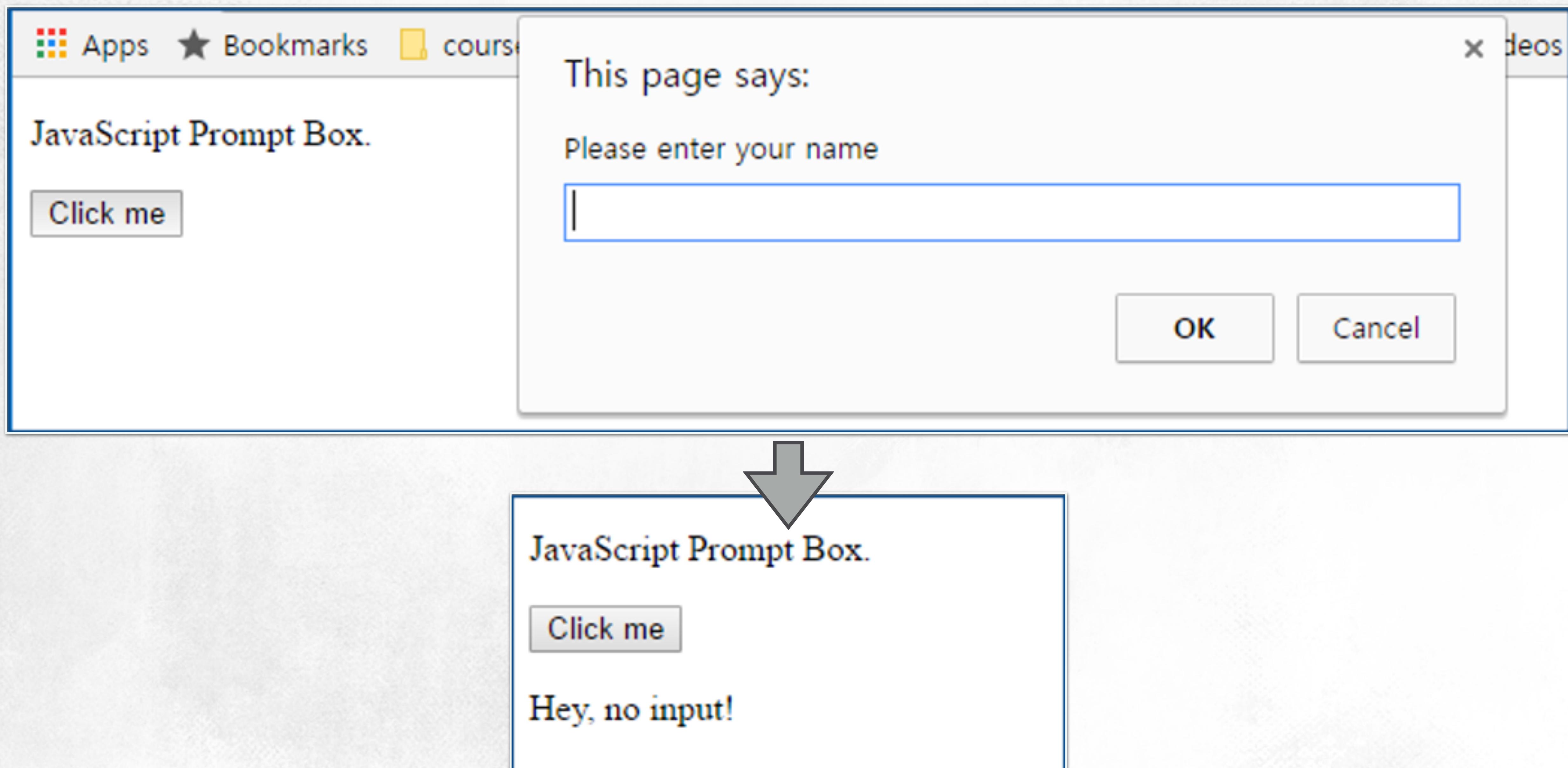
```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>Prompt Page</title>
5
6      <script>
7
8          function myPrompt()
9          {
10              var person = prompt("Please enter your name", "Fred"); // The second string optional
11              if (person != "")
12              {
13                  document.getElementById("test").innerHTML =
14                      "Hello " + person + "! How are you today?";
15              }
16          }
17      </script>
18
19  </head>
20
21  <body>
22
23      <p>JavaScript Prompt Box.</p>
24
25      <button onclick="myPrompt()">Click me</button>
26
27      <p id="test"></p>
28
29  </body>
30
31  </html>
32
```

Alert and Prompt

- Both an alert and a prompt display a pop-up to the user.
- An alert simply displays a message to the user.
- A Prompt asks the user for some sort of input.

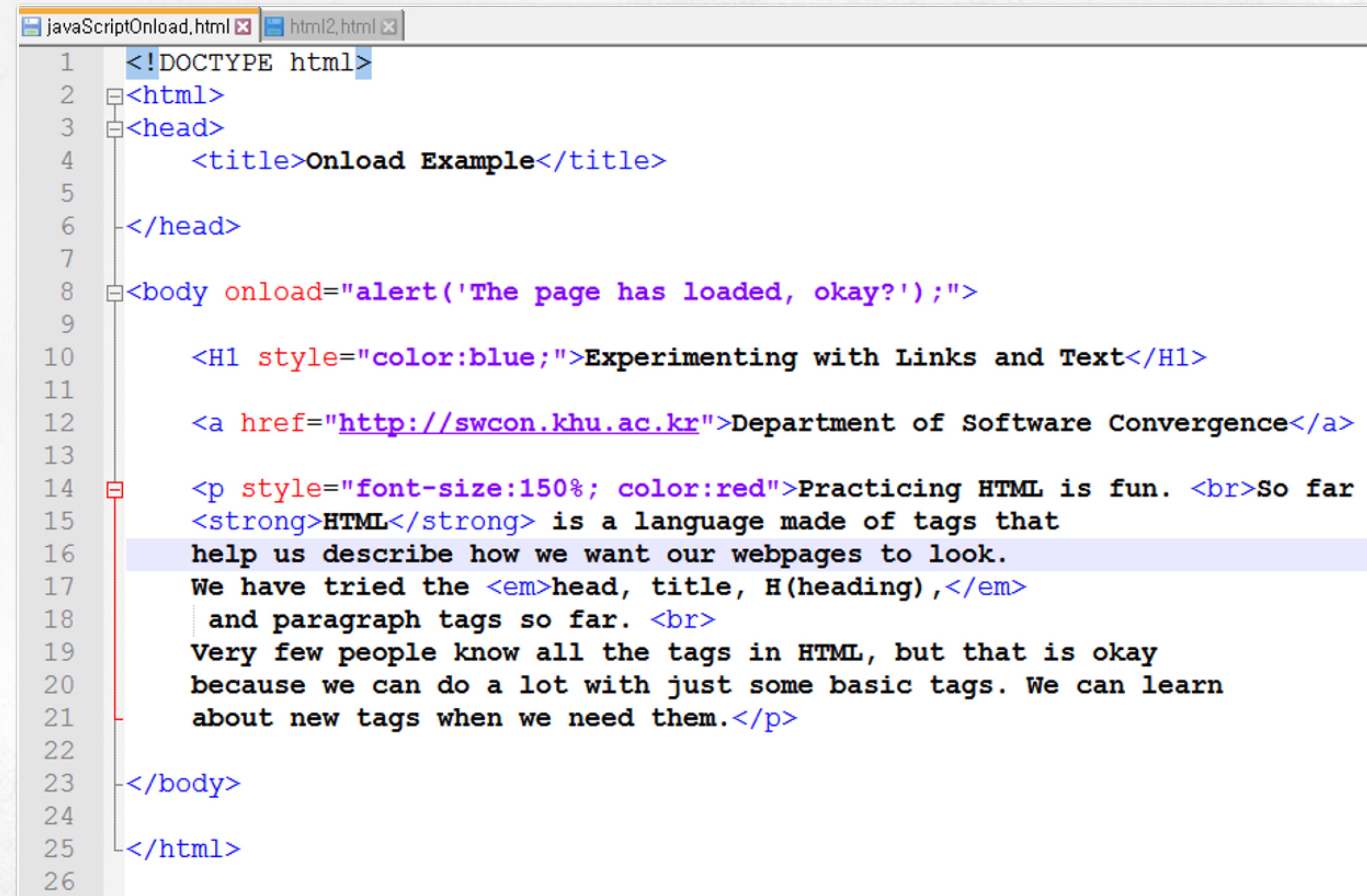
DIY Exercise

- If nothing is entered through the prompt,
 - show “Hey, no input” on the web page.



Event

- onclick
- onload
- onblur
- onmouseover



```
javaScriptOnLoad.html x html2.html x
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Onload Example</title>
5 
6 </head>
7 
8 <body onload="alert('The page has loaded, okay?');">
9 
10  <H1 style="color:blue;">Experimenting with Links and Text</H1>
11 
12  <a href="http://swcon.khu.ac.kr">Department of Software Convergence</a>
13 
14  <p style="font-size:150%; color:red">Practicing HTML is fun. <br>So far
15  <strong>HTML</strong> is a language made of tags that
16  help us describe how we want our webpages to look.
17  We have tried the <em>head, title, H(heading),</em>
18  and paragraph tags so far. <br>
19  Very few people know all the tags in HTML, but that is okay
20  because we can do a lot with just some basic tags. We can learn
21  about new tags when we need them.</p>
22 
23 </body>
24 
25 </html>
26
```

DIY Testing

The screenshot shows a browser window with two tabs: "javaScriptTesting.html" and "html2.html". The left pane displays the source code of "javaScriptTesting.html", which contains an HTML file with a title "Onblur Event Page" and a script block. The script defines a function "doMath" that calculates the result of $x * y / z$. It then sets the innerHTML of a paragraph element with id "output" to "The result is" followed by the calculated value. The right pane shows four separate "JavaScript Testing" boxes, each with a "Do the Math" button. The first box shows the result "The result is Infinity", the second shows "The result is 5", and the third shows "The result is NaN". This demonstrates how different values for variables x, y, and z lead to different results.

```
<!DOCTYPE html>
<html>
<head>
    <title>Onblur Event Page</title>
    <script>
        function doMath()
        {
            var x = 2;
            var y = 10;
            var z = 0;
            var answer = x * y / z;

            document.getElementById("output").innerHTML = "The result is " + answer;
        }
    </script>
</head>
<body>
    <p>JavaScript Testing.</p>
    <button onclick="doMath();">Do the Math</button>
    <p id="output" style="color:green;font-size:200%;"></p>
</body>
</html>
```

JavaScript Testing.
Do the Math

Reference

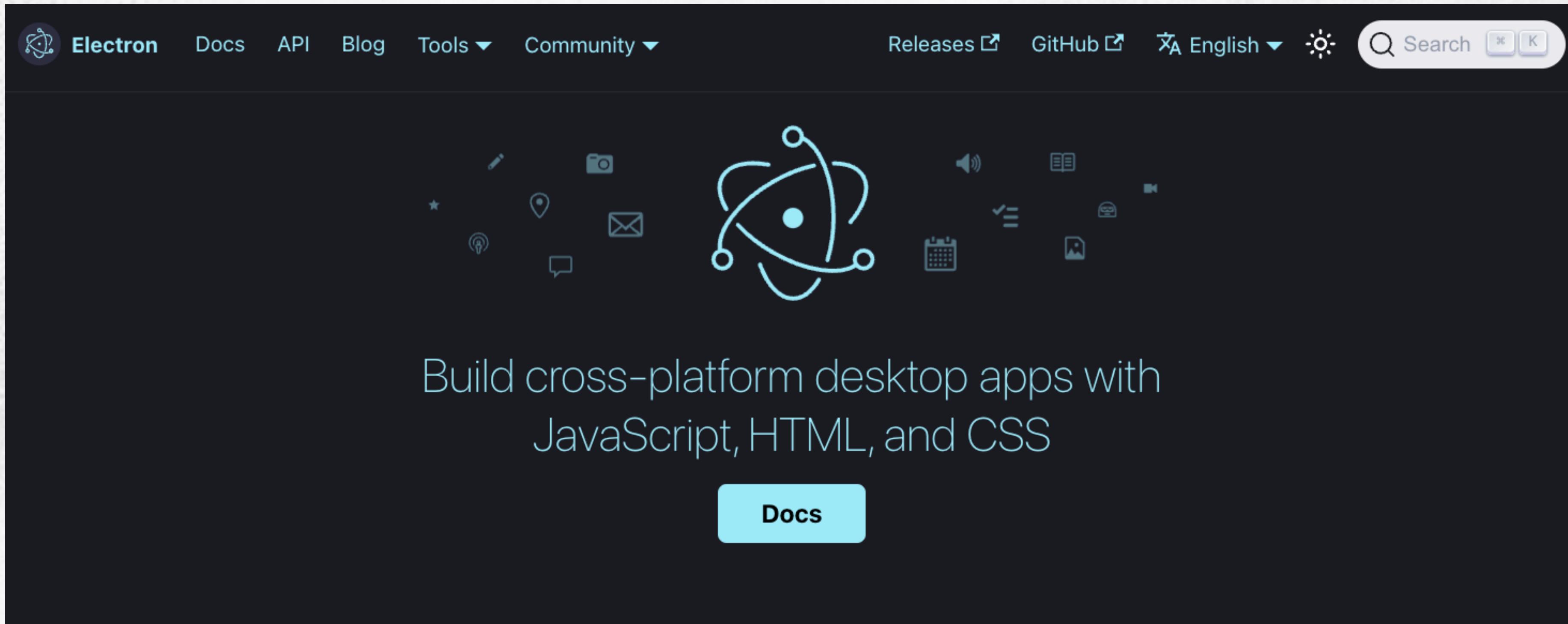
HTML/CSS/JS for Cross-Platform App Dev



A screenshot of the React Native mobile application development interface. It shows two smartphones side-by-side. The left phone displays a "Welcome to React" screen with instructions: "Step One: Edit App.js to change this screen and then come back to see your edits.", "See Your Changes: Double tap R on your keyboard to reload your app's code.", and "Debug: Press Cmd+R in the terminal to see what's happening in your app's code.". The right phone also displays the same "Welcome to React" screen. To the right of the phones is a section titled "Create native apps for Android, iOS, and more using React". It explains that React Native combines native development with React and provides instructions for using it in existing projects or creating new ones from scratch.

Reference

HTML/CSS/JS for Cross-Platform Desktop Dev



Web Technologies

Electron embeds Chromium and Node.js to enable web developers to create desktop applications.



Cross Platform

Compatible with macOS, Windows, and Linux, Electron apps run on three platforms across all supported architectures.



Open Source

Electron is an open source project maintained by the [OpenJS Foundation](#) and an active community of contributors.



Reference

JS for Server Dev

The screenshot shows the Node.js website's download page for macOS. At the top, there is a dark header bar with the Node.js logo and navigation links in Korean: 홈 | ABOUT | 다운로드 | 문서 | 참여하기 | 보안 | 뉴스 | CERTIFICATION. To the right of the header are icons for dark mode and accessibility.

Below the header, a large text states: "Node.js®는 Chrome V8 JavaScript 엔진으로 빌드된 JavaScript 런타임입니다."

The main section is titled "다운로드 - macOS". It features two prominent download buttons:

- 18.12.1 LTS** (Light Green Button)
안정적, 신뢰도 높음
- 19.1.0 현재 버전** (Dark Green Button)
최신 기능

Below each button, there are links to "다른 운영 체제 | 변경사항 | API 문서".

A note at the bottom says: "LTS 일정은 [여기서 확인하세요.](#)"

Reference

www.w3schools.com

The screenshot shows the w3schools.com homepage. At the top, there's a navigation bar with links for Tutorials, References, Exercises, Videos, and a search bar. Below the navigation is a large dark banner with the text "Learn to Code" and "With the world's largest web developer site.". A search bar and a "Not Sure Where To Begin?" button are also present. On the left side, there's a sidebar with the title "JavaScript" and the subtitle "The language for programming web pages". It includes buttons for "Learn JavaScript", "JavaScript Reference", and "Get Certified". In the center, there's a white box titled "JavaScript Example:" containing the code shown in the image below. A "Try it Yourself" button is at the bottom of this box.

```
<button onclick="myFunction()">Click Me!</button>

<script>
function myFunction() {
    let x = document.getElementById("demo");
    x.style.fontSize = "25px";
    x.style.color = "red";
}
</script>
```

This screenshot shows a detailed JavaScript example page. At the top, it says "JavaScript Example:". Below that is the same code as in the previous image. At the bottom, there's a large green "Try it Yourself" button.

```
<button onclick="myFunction()">Click Me!</button>

<script>
function myFunction() {
    let x = document.getElementById("demo");
    x.style.fontSize = "25px";
    x.style.color = "red";
}
</script>
```



Reference





Thank you