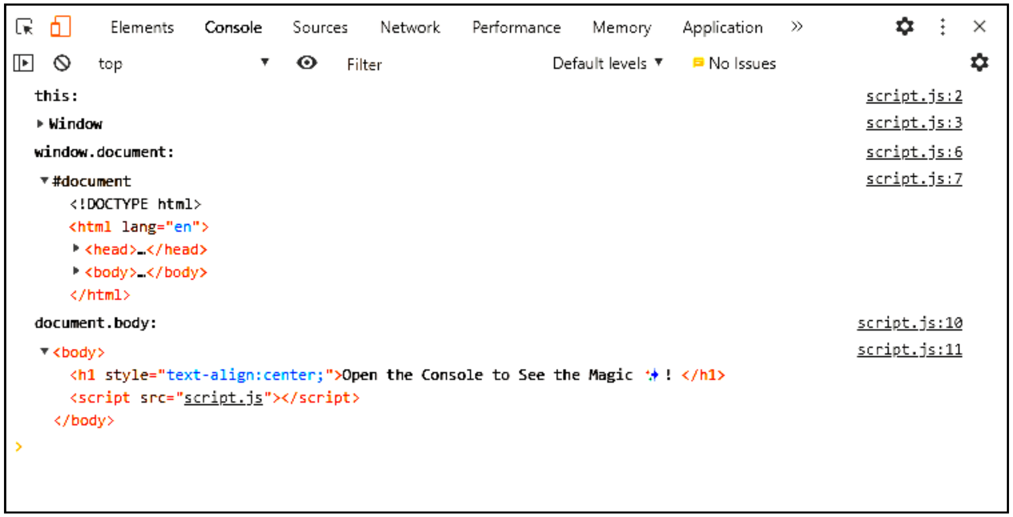
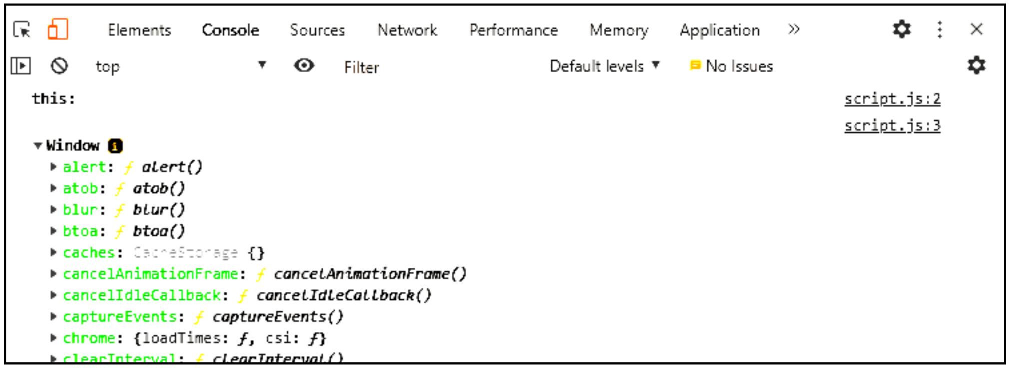
Window Object

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport"
          content="width=device-width,
          initial-scale=1.0">
  <title>Window Object</title>
</head>
<body>
  <h1 style="text-align:center;">
          Open the Console to See the Magic □! </h1>

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

```
// Logs window object using this
console.log("this: ");
console.log(this);

// Logs the document object
console.log("window.document: ");
console.log(window.document);

// Logs body of document
console.log("document.body: ");
console.log( document.body);
```



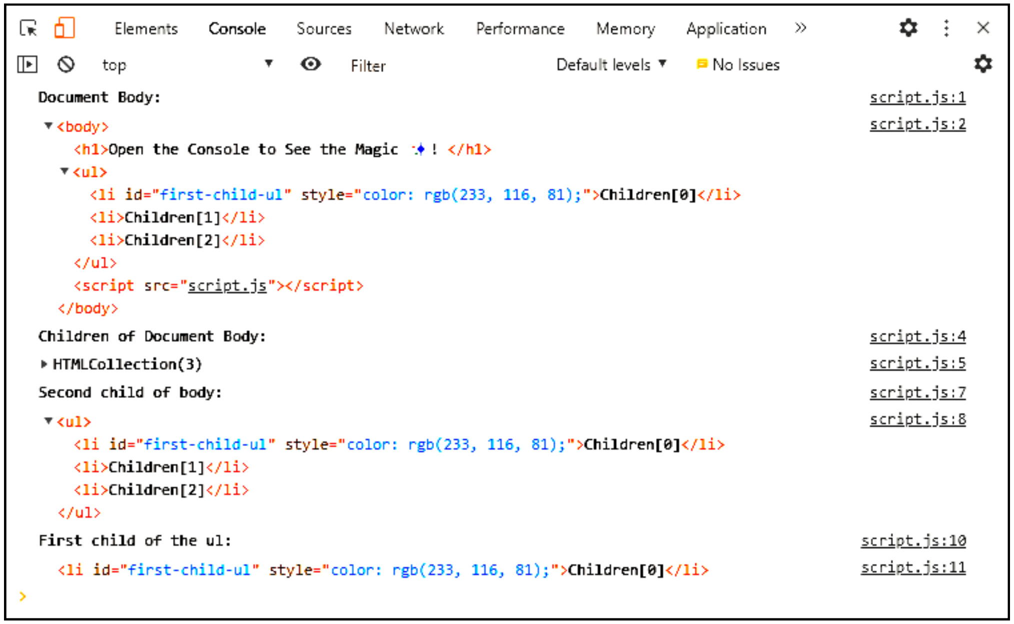
```
this:                                                                    script.js:2
▸ Window                                                                 script.js:3
window.document:                                                         script.js:6
▾ #document                                                              script.js:7
    <!DOCTYPE html>
    <html lang="en">
    ▸ <head>…</head>
    ▸ <body>…</body>
    </html>
document.body:                                                           script.js:10
▾ <body>                                                                 script.js:11
    <h1 style="text-align:center;">Open the Console to See the Magic ✨! </h1>
    <script src="script.js"></script>
  </body>
>
```



```
this:                                                       script.js:2
                                                            script.js:3
▾ Window ℹ
  ▸ alert: ƒ alert()
  ▸ atob: ƒ atob()
  ▸ blur: ƒ blur()
  ▸ btoa: ƒ btoa()
  ▸ caches: CacheStorage {}
  ▸ cancelAnimationFrame: ƒ cancelAnimationFrame()
  ▸ cancelIdleCallback: ƒ cancelIdleCallback()
  ▸ captureEvents: ƒ captureEvents()
  ▸ chrome: {loadTimes: ƒ, csi: ƒ}
  ▸ clearInterval: ƒ clearInterval()
```

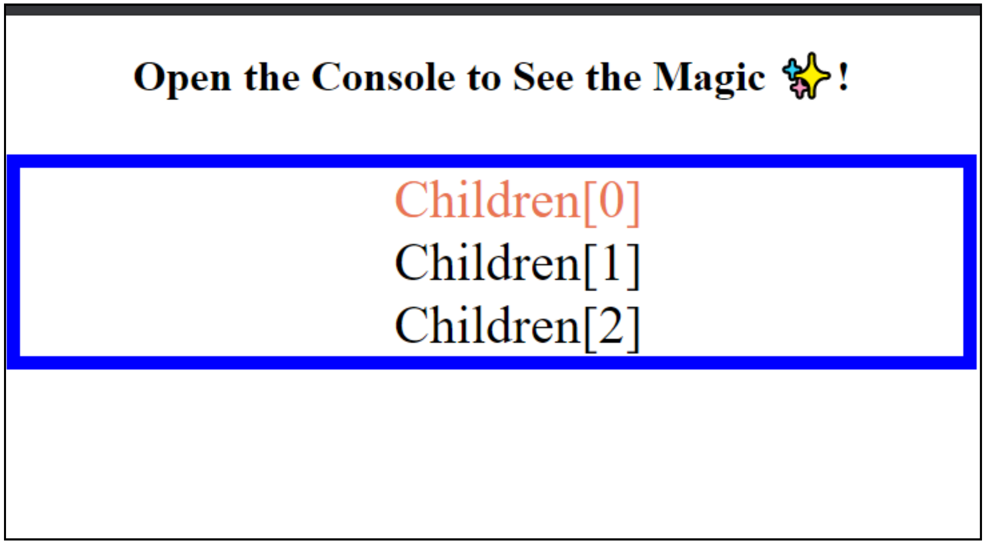
Traverse-DOM

```html
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport"
            content="width=device-width, initial-scale=1.0">
  <title>DOM Traversal</title>
  <style>
     h1 {
       text-align:center;
     }
     ul {
       text-align:center;
       list-style-type:none;
       font-size: 40px;
       border: 10px solid blue
     }
  </style>
</head>
<body>
  <h1>Open the Console to See the Magic □! </h1>
  <ul>
    <li id ="first-child-ul">Children[0]</li>
    <li>Children[1]</li>
    <li>Children[2]</li>
  </ul>
  <script src= "script.js"></script>
</body>
</html>
```



```
Document Body:                                                              script.js:1
 ▾<body>                                                                    script.js:2
    <h1>Open the Console to See the Magic ✨! </h1>
  ▾<ul>
      <li id="first-child-ul" style="color: rgb(233, 116, 81);">Children[0]</li>
      <li>Children[1]</li>
      <li>Children[2]</li>
    </ul>
    <script src="script.js"></script>
  </body>
Children of Document Body:                                                  script.js:4
▸HTMLCollection(3)                                                          script.js:5
Second child of body:                                                       script.js:7
 ▾<ul>                                                                      script.js:8
    <li id="first-child-ul" style="color: rgb(233, 116, 81);">Children[0]</li>
    <li>Children[1]</li>
    <li>Children[2]</li>
  </ul>
First child of the ul:                                                     script.js:10
  <li id="first-child-ul" style="color: rgb(233, 116, 81);">Children[0]</li> script.js:11
>
```

```javascript
console.log("Document Body: ")
console.log(document.body);
console.log("Children of Document Body: ")
console.log(document.body.children);
console.log("Second child of body: ")
console.log(document.body.children[1]);
console.log("First child of the ul: ")
console.log(document.body.children[1].children[0]);

// Accessing element by id
var firstChildUl = document.getElementById("first-child-ul");

// Setting style of element
firstChildUl.style.color = "#e97451";
```

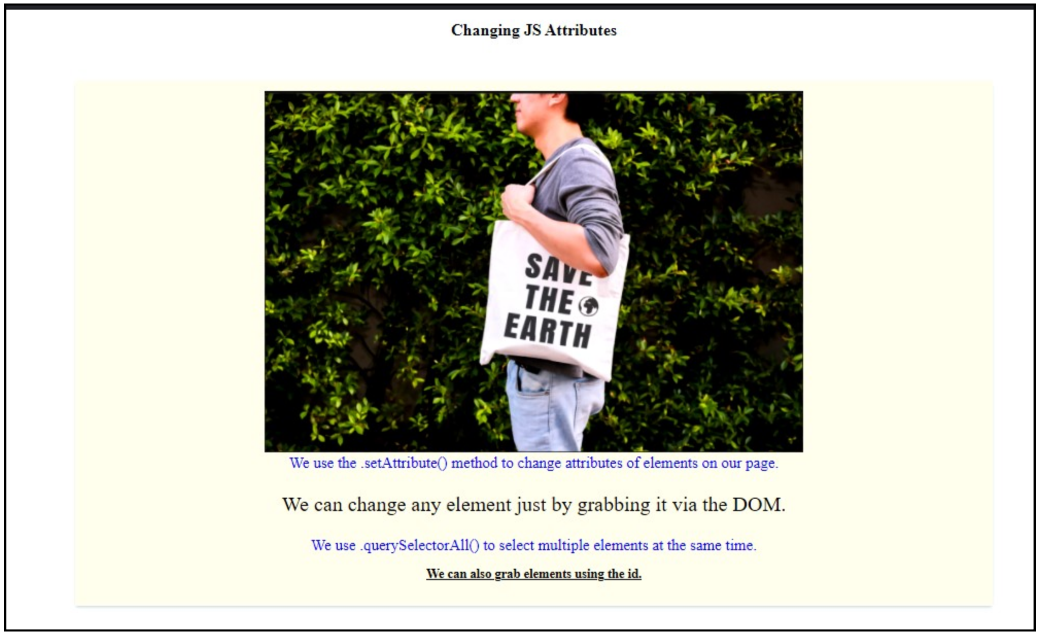


Open the Console to See the Magic ✨!

Children[0]
Children[1]
Children[2]

Setting Attributes

```html
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport"
            content="width=device-width, initial-scale=1.0">
   <meta http-equiv="X-UA-Compatible" content="ie=edge">
   <title>Setting Attributes</title>
   <link rel="stylesheet" href="./assets/css/style.css">
</head>
<body>
   <h1 class="title">Changing JS Attributes</h1>
   <section class="container">
   <a>
      <img>
   </a>
   <div>We use the .setAttribute()
            method to change attributes
            of elements on our page.</div>
   <p>We can change any element
            just by grabbing it via the DOM.</p>
   <div class="change1">We use .querySelectorAll()
            to select multiple elements at the same time.</div>
   <p id="change2">We can also grab elements using the id.</p>
   </section>
   <script src="assets/js/script.js"></script>
</body>
</html>
```



Changing JS Attributes

We use the .setAttribute() method to change attributes of elements on our page.

We can change any element just by grabbing it via the DOM.

We use .querySelectorAll() to select multiple elements at the same time.

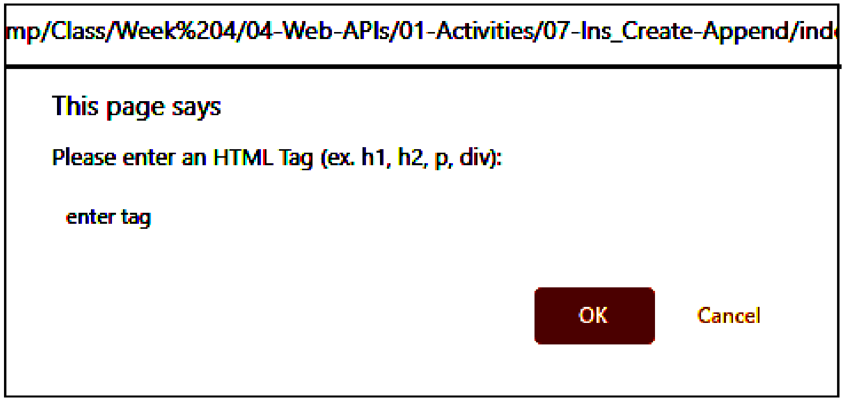
We can also grab elements using the id.

```css
body {
   background-color: #eff5f8
  }
.title {
   text-align: center;
}
.container {
   display: block;
   margin: 80px auto;
   text-align: center;
   box-shadow: 0 4px 8px 0 rgba(0,0,0,0.2);
   transition: 0.3s;
   width: 60vw;
   padding: 20px;
   background-color: #f0ead6;
  }
  @media screen and (min-width: 600px) {
   img {
      min-width: 60%;
   }
  }
```

```javascript
// Access multiple elements using .querySelectorAll()
 var divTags = document.querySelectorAll("div");
 var pTags = document.querySelectorAll("p");
 var imgEl = document.querySelectorAll("img");

// Access element by ID using .querySelector()
 var changeP = document.querySelector("#change2");

// Sets first pTags to have a font-size of 40px
 pTags[0].setAttribute("style", "font-size: 40px;");

// Sets changeP to have multiple style attributes
 changeP.setAttribute("style", "font-size: 25px; font-weight: bold; text-decoration:underline; ");

 // Sets image source of the first image
 imgEl[0].setAttribute("src", "./assets/images/image_1.png");

// Adds size and width styling to image
 imgEl[0].setAttribute("style", "width:50%");

// Loops through divTags to set each one to have the color blue and the font size of 30px
for (var i = 0; i < divTags.length; i++) {
  divTags[i].setAttribute("style", "color:blue; font-size: 30px");
 }
```

Create & Append

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport"
            content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Create and Append</title>
</head>
<body>
  <script src="./script.js"></script>
</body>
</html>
```

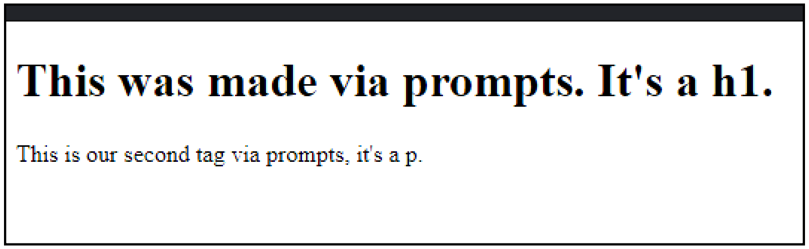


mp/Class/Week%204/04-Web-APIs/01-Activities/07-Ins_Create-Append/ind

This page says

Please enter an HTML Tag (ex. h1, h2, p, div):

enter tag



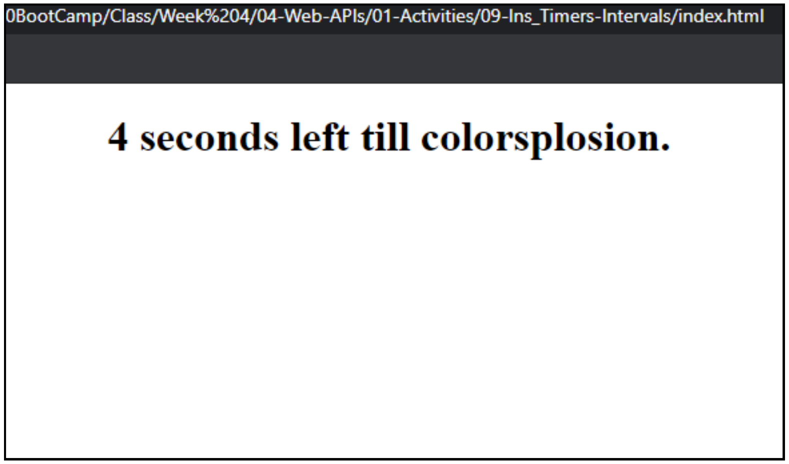
This was made via prompts. It's a h1.

This is our second tag via prompts, it's a p.

```
// Stores user response in variable
var tagName = prompt("Please enter an HTML Tag (ex. h1, h2, p, div):", "enter tag");
if (tagName !== "h1" && tagName !== "h2" && tagName !== "p" && tagName !== "div") {
  alert("please enter a valid tag");
} else {
// Creates element based on tag entered by user
  var tag = document.createElement(tagName);

// Adds text content to created tag
  tag.textContent = "This was made via prompts. It's a " + tagName + ".";

// Appends tag as child of document body
  document.body.appendChild(tag);
}
var nextTag = confirm("Would you like to add another tag?");
if (nextTag === true) {
  var secondTagName = prompt("Please enter another  HTML Tag (ex. h1, h2, p, div):", "enter tag here");
  if(secondTagName !== "h1" && secondTagName !== "h2" && secondTagName !== "p" && secondTagName !== "div") {
    alert("please enter a valid tag");
  } else {
    var secondTag = document.createElement(secondTagName);
    secondTag.textContent = "This is our second tag via prompts, it's a " + secondTagName + ".";
    document.body.appendChild(secondTag);
  }
}
```

Timers-Intervals

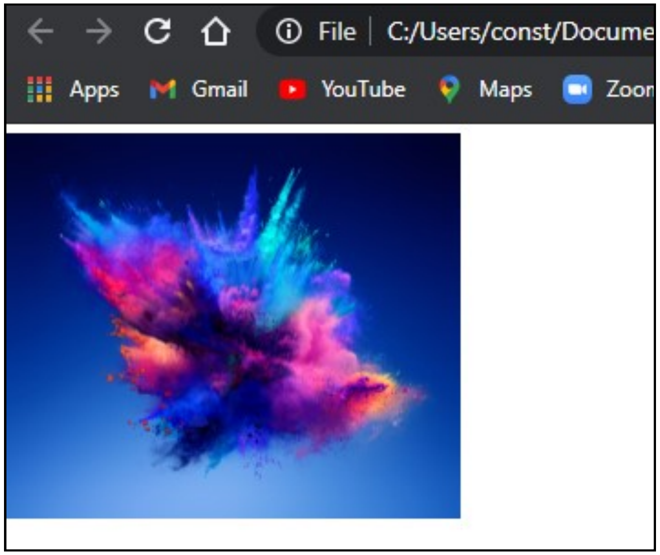
```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport"
            content="width=device-width, initial-scale=1.0">
   <meta http-equiv="X-UA-Compatible" content="ie=edge">
   <title>Timers and Intervals</title>
</head>
<body>
   <h1 class="time" style="text-align:center;"></h1>
   <div id="main"></div>
   <script src="./script.js"></script>
</body>
</html>
```



0BootCamp/Class/Week%204/04-Web-APIs/01-Activities/09-Ins_Timers-Intervals/index.html

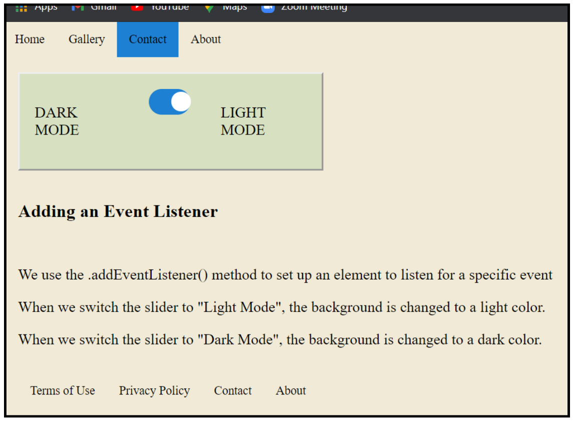
4 seconds left till colorsplosion.

```
// Selects element by class
var timeEl = document.querySelector(".time");

// Selects element by id
var mainEl = document.getElementById("main");
var secondsLeft = 10;
function setTime() {

  // Sets interval in variable
  var timerInterval = setInterval(function() {
    secondsLeft--;
    timeEl.textContent = secondsLeft + " seconds left till colorsplosion.";
    if(secondsLeft === 0) {

      // Stops execution of action at set interval
      clearInterval(timerInterval);

      // Calls function to create and append image
      sendMessage();
    }
  }, 1000);
}

// Function to create and append colorsplosion image
function sendMessage() {
  timeEl.textContent = " ";
  var imgEl = document.createElement("img");
  imgEl.setAttribute("src", "images/image_1.jpg");
  mainEl.appendChild(imgEl);
}

setTime();
```



Event-Listeners

```html
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Click Event</title>
  <link rel="stylesheet" type="text/css" href="./assets/css/style.css">
</head>
<body>
  <div class="container dark">
    <div class="menu">
      <ul>
        <li><a href="#home">Home</a></li>
        <li><a href="#gallery">Gallery</a></li>
        <li><a href="#contact">Contact</a></li>
        <li><a href="#about">About</a></li>
      </ul>
    </div>
    <div class="row toggle">
      <div class="col">
        <p>DARK MODE</p>
      </div>
      <label class="switch col">
        <input type="checkbox" id="theme-switcher">
        <span class="slider round"></span>
      </label>
      <div class="col">
        <p>LIGHT MODE</p>
      </div>
    </div>
    <div class="row">
      <h2>Adding an Event Listener</h2>
    </div>
    <div class="row">
      <div>
        <p>We use the .addEventListener() method to set up an element to listen for a specific event</p>
        <p id= "light-mode">When we switch the slider to "Light Mode", the background is changed to a light color.</p>
        <p id = "dark-mode">When we switch the slider to "Dark Mode", the background is changed to a dark color.</p>
      </div>
      <div class="footer">
        <ul>
          <li><a href="#home">Terms of Use</a></li>
          <li><a href="#gallery">Privacy Policy</a></li>
          <li><a href="#contact">Contact</a></li>
          <li><a href="#about">About</a></li>
        </ul>
      </div>
    </div>
  <script src="./assets/js/script.js"></script>
</body>
</html>
```

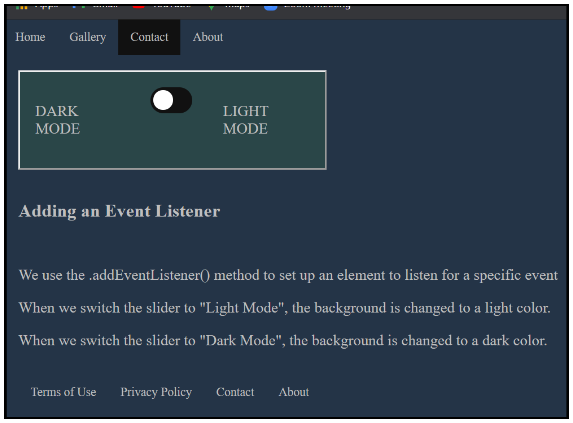


Adding an Event Listener

We use the .addEventListener() method to set up an element to listen for a specific event

When we switch the slider to "Light Mode", the background is changed to a light color.

When we switch the slider to "Dark Mode", the background is changed to a dark color.



Adding an Event Listener

We use the .addEventListener() method to set up an element to listen for a specific event

When we switch the slider to "Light Mode", the background is changed to a light color.

When we switch the slider to "Dark Mode", the background is changed to a dark color.

```css
body {
  display: block;
  margin: 0px;
}
ul {
  list-style-type: none;
  margin: 0;
  padding: 0;
  overflow: hidden;
}
li {
  float: left;
}
li a {
  display: block;
  text-align: center;
  padding: 14px 16px;
  text-decoration: none;
}
p {
  font-size: 20px;
}
.menu {
  width: 100%;
}
.footer {
  width: 100%;
  position: absolute;
  padding-bottom: 100px;
  bottom: 0px;
}
```

```css
.toggle {
  border-style: outset;
  width: 400px;
  background-color: rgba
(76, 175, 80, 0.15)
}

/* Dark Mode */
.dark {
  background-color:#243447;
  height: 100vh;
  color: #bebebe
}
.dark li a {
  color: #bebebe;
}
.dark li a:hover {
  background-color: #111;
}

/* Light Mode */
.light {
  background-color: #f0ead6;
  color: black;
  height: 100vh;
}
.light li a {
  color: black;
}
.light li a:hover {
  background-color: rgb(29, 128, 210);
}
```

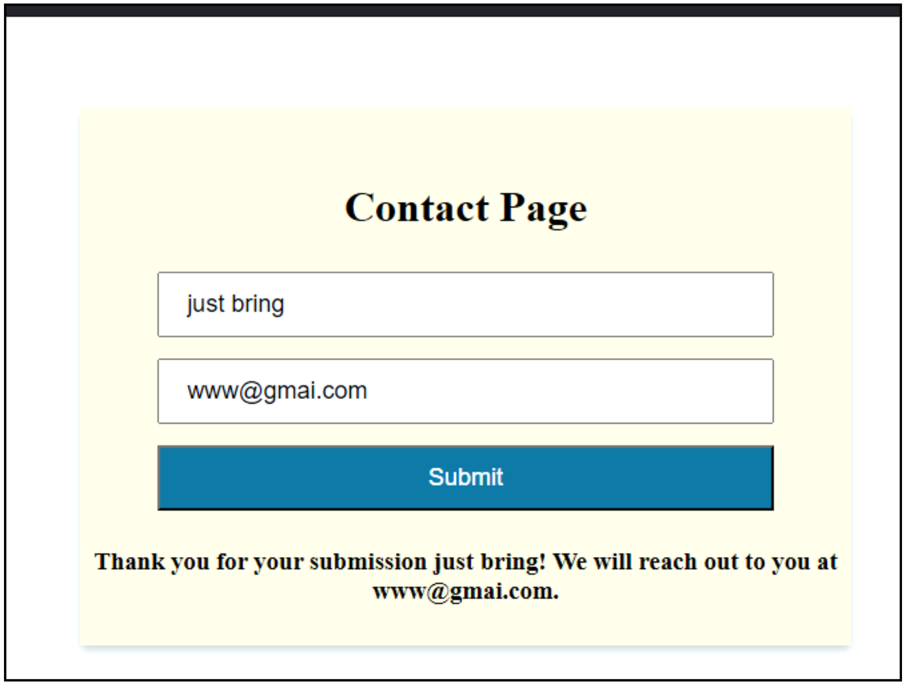
```css
/* Styling the Switch button */
.switch {
  position: relative;
  display: inline-block;
  width: 60px;
  height: 34px;
}
/* Hide default HTML checkbox */
.switch input {
  opacity: 0;
  width: 0;
  height: 0;
}
/* The slider */
.slider {
  position: absolute;
  cursor: pointer;
  top: 0;
  left: 0;
  right: 0;
  bottom: 0;
  background-color: #111;
  -webkit-transition: .4s;
  transition: .4s;
}
.slider:before {
  position: absolute;
  content: "";
  height: 26px;
  width: 26px;
  left: 4px;
  bottom: 4px;
  background-color: white;
  -webkit-transition: .4s;
  transition: .4s;
}
```

```css
input:checked + .slider {
  background-color: rgb(29, 128, 210);
}
input:focus + .slider {
  box-shadow: 0 0 1px rgb
(29, 128, 210);
}
input:checked + .slider:before {
  -webkit-transform: translateX(26px);
  -ms-transform: translateX(26px);
  transform: translateX(26px);
}

/* Rounded sliders */
.slider.round {
  border-radius: 34px;
}
.slider.round:before {
  border-radius: 50%;
}
.row {
  margin: 20px;
  display: flex;
}
.col {
  margin: 20px;
}
```

```javascript
// Access toggle switch HTML element
var themeSwitcher = document.querySelector("#theme-switcher");
var container = document.querySelector(".container");

// Set default mode to dark
var mode = "dark";

// Listen for a click event on toggle switch
themeSwitcher.addEventListener("click", function() {

// If mode is dark, apply light background
  if (mode === "dark") {
    mode = "light";
    container.setAttribute("class", "light");
  }

// If mode is light, apply dark background
  else {
    mode = "dark";
    container.setAttribute("class", "dark");
  }
});
```

Preventing Default

```html
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Event Prevent Default Demo</title>
  <link rel="stylesheet" href="./assets/css/style.css">
</head>
<body>
  <div class="wrapper">
    <div class="container">
      <div class= "card-contents"></div>
        <h1>Contact Page</h1>
        <form>
          <input placeholder="name" id="name" />
          <input placeholder="email" id="email" />
          <button id="submit">Submit</button>
          <div>
            <h3 id="response"></h3>
          </div>
        </form>
      </div>
    </div>
  </div>
  <script src="./assets/js/script.js"></script>
</body>
</html>
```



Contact Page

just bring

www@gmai.com

Thank you for your submission just bring! We will reach out to you at www@gmai.com.

```css
body {
  background-color: #eff5f8
}
button {
    padding: 10px;
    margin: 24px;
    cursor: pointer;
    font-size: 24px;
    background-color: #247BA0;
    color: white;
  }

input {
  background-color: #f7f7f7;
  }
.wrapper {
  padding-top: 20px;
}
```

```css
.container {
    display: block;
    margin: 40px auto;
    text-align: center;
    box-shadow: 0 4px 8px 0 rgba(0,0,0,0.2);
    transition: 0.3s;
    width: 60vw;
    background-color: #f0ead6;
  }
.card-contents {
    padding: 5px 5px 30px 5px;
  }
#name,
#email,
#submit {
    width: 80%;
    padding: 12px 20px;
    margin: 8px 20px;
    box-sizing: border-box;
    font-size: 18px;
  }
#response {
      padding-bottom: 30px;
  }
```

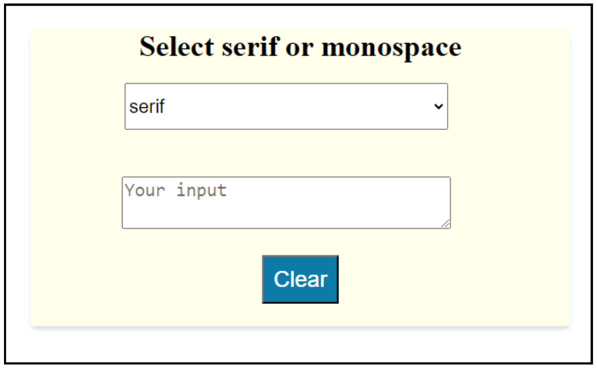
```javascript
var submitEl = document.querySelector("#submit");
var nameInput = document.querySelector("#name");
var emailInput = document.querySelector("#email");
var submissionResponseEl = document.querySelector("#response");

// Action to be performed on click store in named function
function showResponse(event) {

// Prevent default action
  event.preventDefault();
  console.log(event);
  var response = "Thank you for your submission " + nameInput.value +
           "! We will reach out to you at " + emailInput.value + ".";
  submissionResponseEl.textContent = response;
}

// Add listener to submit element
submitEl.addEventListener("click", showResponse);
```

Keyboard Events

```html
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Font Checker</title>
  <link rel="stylesheet" href="./assets/css/style.css">
</head>
<body>
  <div class="container">
    <h1>Select serif or monospace</h1>
    <form>
      <select id="typeface">
        <option value="serif">
          serif
        </option>
        <option value="monospace">
          monospace
        </option>
      </select>
      <div style="margin-top: 50px;">
        <textarea id="textarea" placeholder="Your input" ></textarea>
        <div>
          <button id="clear">Clear</button>
        </div>
      </div>
    </form>
    <div class="elements">
      <h1 id="h1"></h1>
      <h2 id="h2"></h2>
      <h3 id="h3"></h3>
      <p id="p"></p>
    </div>
  </div>

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

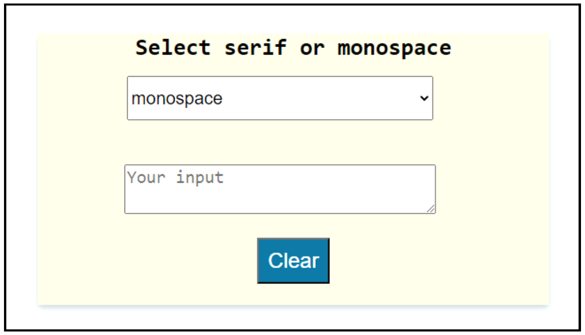
```css
body {
  background-color: #eff5f8
}
button {
  padding: 10px;
  margin: 24px;
  cursor: pointer;
  font-size: 24px;
  background-color: #247BA0;
  color: white;
}
.container {
  display: block;
  margin: 80px auto;
  text-align: center;
  box-shadow: 0 4px 8px 0 rgba(0,0,0,0.2);
  transition: 0.3s;
  width: 60vw;
  background-color: #f0ead6;
}
textarea, select {
  height: 50px;
  margin-right: 30px;
  font-size: 20px;
  width: 60%;
}
```

```javascript
var typefaceEl = document.querySelector('#typeface');
var clearEl = document.querySelector('#clear');
var h1El = document.querySelector('#h1');
var h2El = document.querySelector('#h2');
var h3El = document.querySelector('#h3');
var pEl = document.querySelector('#p');
var textAreaEl = document.querySelector('#textarea');
var elements = [h1El, h2El, h3El, pEl];
var typeface;

// Change event
typefaceEl.addEventListener('change', function (event) {
  event.preventDefault();
  typeface = typefaceEl.value;
  document.querySelector('.container').style.fontFamily = typeface;
});

// Keydown event
textAreaEl.addEventListener('keydown', function (event) {

// Access value of pressed key with key property
  var key = event.key.toLowerCase();
  var alphabetNumericCharacters =
            'abcdefghijklmnopqrstuvwxyz0123456789 '.split(
    ''
  );
  if (alphabetNumericCharacters.includes(key)) {
    for (var i = 0; i < elements.length; i++) {
      elements[i].textContent += event.key;
    }
  }
});
clearEl.addEventListener('click', function (event) {
  event.preventDefault();
  textAreaEl.value = '';
  for (var i = 0; i < elements.length; i++) {
    elements[i].textContent = '';
  }
});
```



Select serif or monospace

serif



Select serif or monospace

monospace

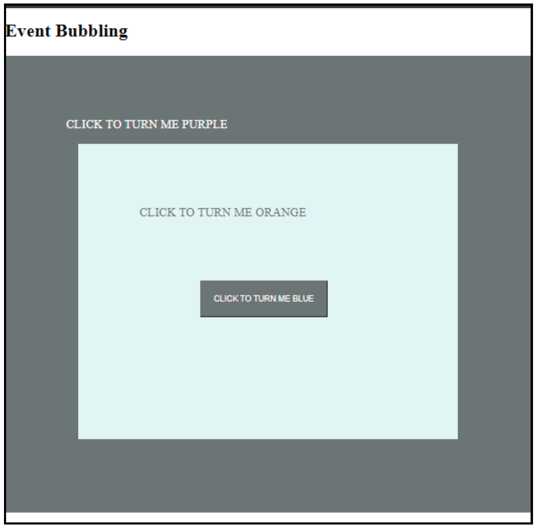
Event Bubbling

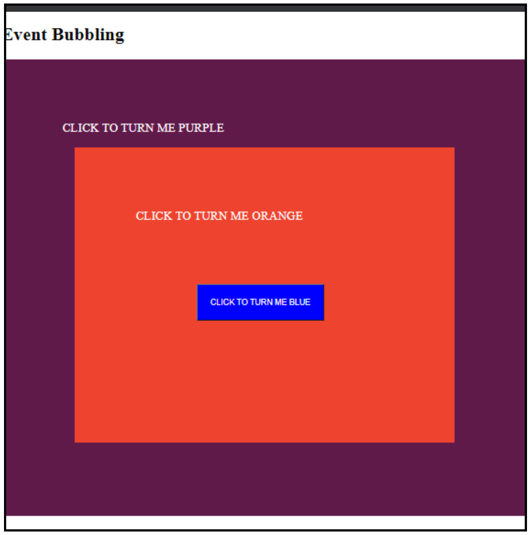
```html
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport"
          content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <link rel="stylesheet" href="./assets/css/style.css">
</head>
<body>
  <div>
    <h2>Event Bubbling </h2>
    <div class="outer-div">CLICK TO TURN ME PURPLE
      <div class="inner-div">
          CLICK TO TURN ME ORANGE<div>
        <button class="button">
          CLICK TO TURN ME BLUE</button>
      </div>
      </div>
    </div>
  </div>
  <script src="./assets/js/script.js" ></script>
</body>
</html>
```

```css
*, *::before, *::after {
  box-sizing: border-box;
}
html, body, .outer-div {
  height: 100%;
  margin: 0;
  padding: 0;
  font-size: 20px;
}
body {
  color: white;
}
h2 {
  color: black;
}
.outer-div {
  background-color: #6c7575;
  cursor: pointer;
  padding: 100px;
}
```

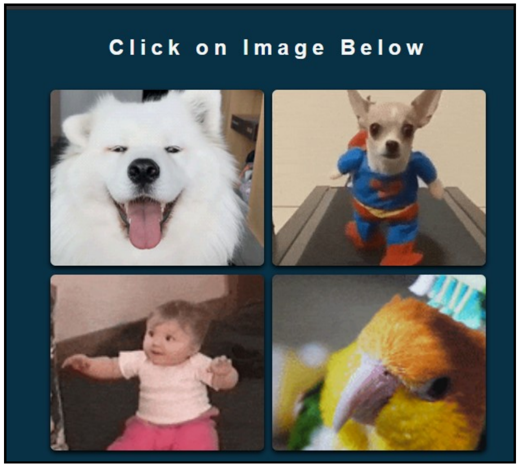
```css
.inner-div {
  height: 60%;
  background-color: #e1f5f4 ;
  padding: 100px;
  margin: 20px;
  color: #6c7575;
}
.button {
  background-color:#6c7575;
  color: white;
  padding: 20px;
  margin: 100px;
}
```

```javascript
var outer = document.querySelector(".outer-div");
var inner = document.querySelector(".inner-div");
var button = document.querySelector(".button");

function changeBlue(event) {
// event.stopPropagation();
  event.currentTarget.setAttribute(
    "style",
    "background-color: blue"
  );
}

function changePurple(event) {
// event.stopPropagation();
  event.currentTarget.setAttribute(
    "style",
    "background-color: #601A4A"
  );
}

function changeOrange(event) {
// event.stopPropagation();
  event.currentTarget.setAttribute(
    "style",
    "background-color: #EE442F; color: white;"
  );
}
outer.addEventListener("click", changePurple);
inner.addEventListener("click", changeOrange);
button.addEventListener("click", changeBlue);
```

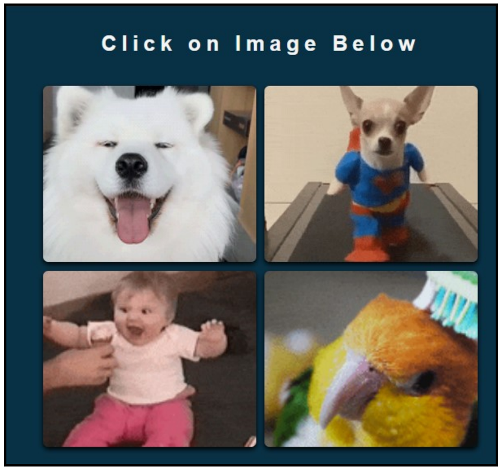




Data Attributes



Click on Image Below



Click on Image Below

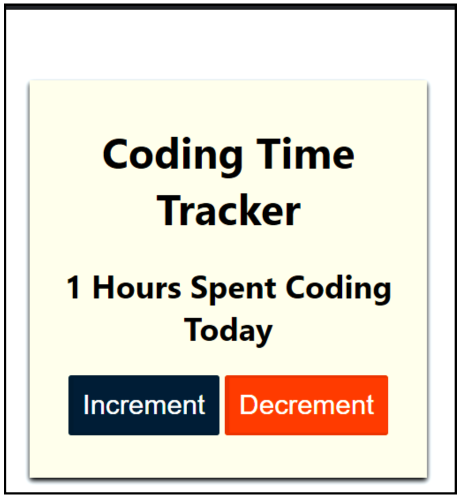
```html
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport"
          content="width=device-width, initial-scale=1.0" />
    <meta http-equiv="X-UA-Compatible" content="ie=edge" />
    <link rel="stylesheet" href="./assets/css/style.css" />
    <title>Gif Player</title>
  </head>
  <body>
    <div class="img-container">
      <h1>Click on Image Below</h1>
<!-- Save references to the image's animated and still versions
          as attributes on the element itself -->
      <img src="https://media0.giphy.com/
          media/4Zo41lhzKt6iZ8xff9/200w_s.gif"
        data-animate="https://media0.giphy.com/
          media/4Zo41lhzKt6iZ8xff9/200w.gif"
        data-still="https://media0.giphy.com/
          media/4Zo41lhzKt6iZ8xff9/200w_s.gif"
        data-state="still"
        alt="Funny dog flapping ears and tilting head side to side"/>
      <img src="https://media2.giphy.com/media/
          RQSuZfuylVNAY/200w_s.gif"
        data-animate="https://media2.giphy.com/media/
          RQSuZfuylVNAY/200w.gif"
        data-still="https://media2.giphy.com/media/
          RQSuZfuylVNAY/200w_s.gif"
        data-state="still"
        alt="Funny dog wearing superhero outfit walking on treadmill"/>
      <img src="https://media0.giphy.com/media/
          AGGz7y0rCYxdS/200w_s.gif"
        data-animate="https://media0.giphy.com/media/
          AGGz7y0rCYxdS/200w.gif"
        data-still="https://media0.giphy.com/media/
          AGGz7y0rCYxdS/200w_s.gif"
        data-state="still"
        alt="Funny baby excited about being presented with ice cream cone"/>
      <img src="https://media2.giphy.com/media/
          vsGnvQD0ZcQco/200w_s.gif"
        data-animate="https://media2.giphy.com/media/
          vsGnvQD0ZcQco/200w.gif"
        data-still="https://media2.giphy.com/media/
          vsGnvQD0ZcQco/200w_s.gif"
        data-state="still"
        alt="Funny bird getting top of head brushed with toothbrush"/>
    </div>
    <script src="./assets/js/script.js"></script>
  </body>
</html>
```

```css
*,
*::before,
*::after {
  box-sizing: border-box;
}
body {
  font-family: sans-serif;
  background-color: #093145;
}
h1 {
  text-align: center;
  margin: 40px auto;
  letter-spacing: 0.25em;
  color: whitesmoke;
}
.img-container {
  max-width: 680px;
  margin: 20px auto;
}
.img-container::after {
  content: " ";
  display: block;
  clear: both;
}
img {
  display: block;
  float: left;
  width: 48%;
  height: 270px;
  margin: 1%;
  cursor: pointer;
  object-fit: cover;
  border-radius: 8px;
  box-shadow: 0 4px 8px 0 rgba(0,0,0, 1);
  transition: 0.3s;
}
```

```javascript
var imageContainer = document.querySelector(".img-container");

// Listen for any clicks within the img-container div
imageContainer.addEventListener("click", function(event) {
  var element = event.target;

// Check if the clicked element was an image
  if (element.matches("img")) {

// Get the current value of the image's data-state attribute
    var state = element.getAttribute("data-state");
    if (state === "still") {

// Change the data-state attribute's value
// There are two different ways this attribute can be set
      element.dataset.state = "animate";
      element.setAttribute("data-state", "animate");

// Update the image's source to the string being stored
          in the data-animate attribute
      element.setAttribute("src", element.dataset.animate);
    } else {

// Change the attributes back to their non-animated values
      element.dataset.state = "still";
      element.setAttribute("src", element.dataset.still);
    }
  }
});
```

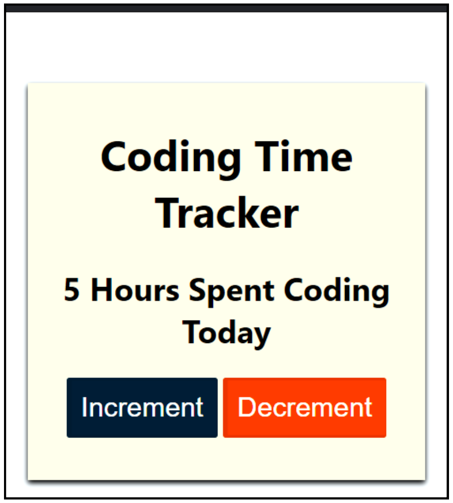
Local Storage

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport"
            content="width=device-width, initial-scale=1.0" />
    <meta http-equiv="X-UA-Compatible" content="ie=edge" />
    <link rel="stylesheet" href="./assets/css/style.css" />
    <title>Coding Timetracker</title>
  </head>

  <body>
    <div class="wrapper">
      <div class="container">
        <h1>Coding Time Tracker</h1>
        <h2><span id="counter">0</span>
            Hours Spent Coding Today</h2>
        <button id="add">Increment</button>
        <button id="subtract">Decrement</button>
      </div>
    </div>

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



Coding Time Tracker

1 Hours Spent Coding Today



Coding Time Tracker

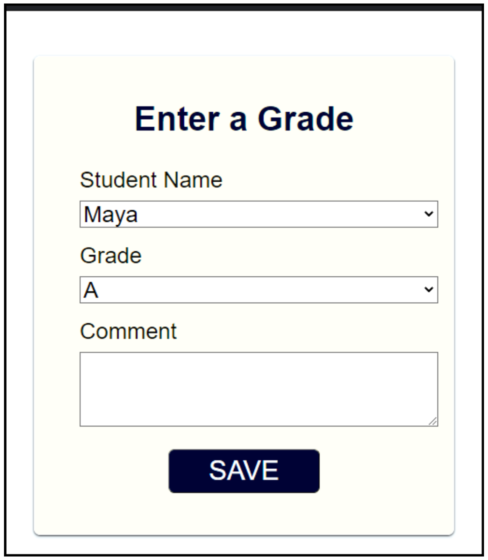
5 Hours Spent Coding Today

```
*, *::before, *::after {
  box-sizing: border-box;
}
html {
  font-size: 16px;
  font-family: 'Segoe UI', Tahoma, Geneva, Verdana,
            sans-serif;
}
body {
  margin: 0;
  padding: 0;
  font-weight: normal;
  background-color: #eff5f8;
}
.container {
  display: block;
  margin: 80px auto;
  text-align: center;
  font-size: 30px;
  box-shadow: 0 4px 8px 0 rgba(0,0,0,1);
  transition: 0.3s;
  width: 60vw;
  background-color: #f0ead6;
  padding-top: 20px;
}
.wrapper {
  padding-top: 20px;
}
h1 {
  font-size: 2em;
}
#add, #subtract {
  border-radius: 4px;
  border-style: none;
  box-shadow: rgba(34,36,38,.15) 3px 0 0 4px inset;
  color: #fff;
  cursor: pointer;
  font-family: "Helvetica Neue",Arial,Helvetica,
            sans-serif;
  font-size: 1.3em;
  padding: 20px;
  margin-bottom: 60px;;
}
#add {
  background-color: #093145;;
}
#subtract {
  background-color: #f64900
}
```

```
var counter = document.querySelector("#counter");
var addButton = document.querySelector("#add");
var subtractButton = document.querySelector("#subtract");
var count = localStorage.getItem("count");
counter.textContent = count;
addButton.addEventListener("click", function() {
  if (count < 24) {
    count++;
    counter.textContent = count;
    localStorage.setItem("count", count);
  }
});
subtractButton.addEventListener("click", function() {
  if (count > 0) {
    count--;
    counter.textContent = count;
    localStorage.setItem("count", count);
  }
});
```

Local Storage Objects

```html
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <meta http-equiv="X-UA-Compatible" content="ie=edge" />
    <link rel="stylesheet" href="./assets/css/style.css" />
    <title>Gradebook</title>
  </head>
  <body>
    <div class="card">
      <h2>Enter a Grade</h2>
      <form>
        <ul>
          <li>
            <label for="name">Student Name</label>
            <select id="student-names">
              <option value="Maya">Maya</option>
              <option value="Alex">Alex</option>
              <option value="Sojung">Sojung</option>
              <option value="Tomas">Tomas</option>
            </select>
          </li>
          <li>
            <label for="grade">Grade</label>
            <select id="grades">
              <option value="A">A</option>
              <option value="B">B</option>
              <option value="C">C</option>
              <option value="Incomplete">Incomplete</option>
            </select>
          </li>
          <li>
            <label for="msg">Comment</label>
            <textarea id="msg" name="comments"></textarea>
          </li>
        </ul>
        <button id="save">SAVE</button>
      </form>
      <h3 class="message"></h3>

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

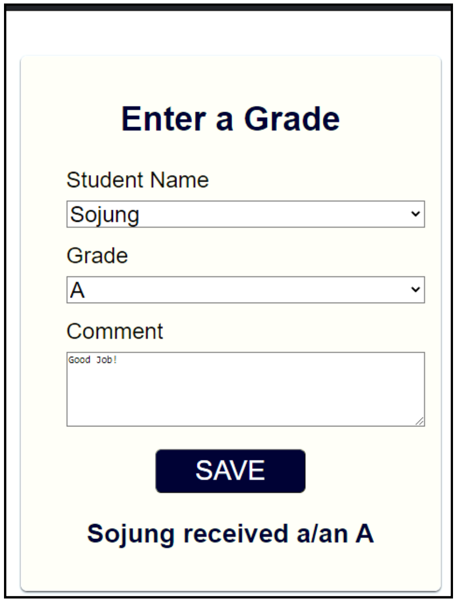


Enter a Grade

Student Name: Maya

Grade: A

Comment:



Enter a Grade

Student Name: Sojung

Grade: A

Comment: Good Job!

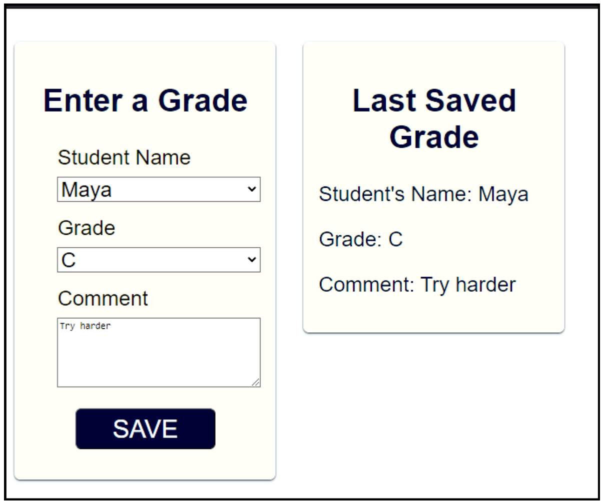
Sojung received a/an A

```css
*,
*::before,
*::after {
  box-sizing: border-box;
}
body {
  font-size: 30px;
  font-family: "Roboto", sans-serif;
  background-color: #eff5f8;
  color: #001b44;
}
h2,
h3 {
  text-align: center;
}
li {
  list-style: none;
  margin-top: 20px;
  display: block;
}
button {
  background-color: #001b44;
  border-radius: 8px;
  color: #fff;
  cursor: pointer;
  font-size: 1.2em;
  letter-spacing: 0.21px;
  line-height: 16px;
  max-width: 255px;
  padding: 20px 52px;
  text-align: center;
  transition-duration: 0.15s;
  transition-property: background-color;
  transition-timing-function: linear;
  margin: 0.6em auto;
  display: block;
}
label {
  color: #212121;
}
select {
  font-size: 30px;
  width: 100%;
  margin-top: 10px;
  background-color: whitesmoke;
}
.card {
  background-color: #FFFFE0;
  border: 1px solid #eceff1;
  border-radius: 8px;
  box-shadow: rgba(60, 64, 67, 2) 0 1px 2px 0,
    rgba(60, 64, 67, 0.15) 0 2px 4px 3px;
  margin: 60px auto;
  padding: 20px;
  width: 60vw;
  display: block;
}
.card::after {
  content: " ";
  display: block;
  clear: both;
}
#msg {
  display: block;
  min-height: 100px;
  width: 100%;
  margin-top: 10px;
  background-color: whitesmoke;
}
```

```javascript
var student = document.getElementById("student-names");
var grade = document.getElementById("grades");
var comment = document.getElementById("msg");
var saveButton = document.getElementById("save");
var savedName = document.getElementById("saved-name");
saveButton.addEventListener("click", function(event) {
event.preventDefault();
var studentGrade = {
  student: student.value,
  grade: grade.value,
  comment: comment.value.trim()
};
localStorage.setItem("studentGrade", JSON.stringify
(studentGrade));
renderMessage();
});
function renderMessage() {
  var lastGrade = JSON.parse(localStorage.getItem
("studentGrade"));
  if (lastGrade !== null) {
    document.querySelector
(".message").textContent = lastGrade.student +
    " received a/an " + lastGrade.grade
  }
}
```

Local Object Todos

```html
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <meta http-equiv="X-UA-Compatible" content="ie=edge" />
    <link rel="stylesheet" href="./assets/css/style.css" />
    <title>Gradebook</title>
  </head>
  <body>
    <div class="card">
      <h2>Enter a Grade</h2>
      <form>
        <ul>
          <li>
            <label for="name">Student Name</label>
            <select id="student-names">
              <option value="Maya">Maya</option>
              <option value="Alex">Alex</option>
              <option value="Sojung">Sojung</option>
              <option value="Tomas">Tomas</option>
            </select>
          </li>
          <li>
            <label for="grade">Grade</label>
            <select id="grades">
              <option value="A">A</option>
              <option value="B">B</option>
              <option value="C">C</option>
              <option value="Incomplete">Incomplete</option>
            </select>
          </li>
          <li>
            <label for="msg">Comment</label>
            <textarea id="msg" name="comments"></textarea>
          </li>
        </ul>
        <button id="save">SAVE</button>
      </form>
    </div>
    <div class="card">
      <h2>Last Saved Grade</h2>
      <p>Student's Name: <span id="saved-name"></span></p>
      <p>Grade: <span id="saved-grade"></span></p>
      <p>Comment: <span id="saved-comment"></span></p>
    </div>

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



Enter a Grade

Student Name: Maya

Grade: C

Comment: Try harder

Last Saved Grade

Student's Name: Maya

Grade: C

Comment: Try harder

```css
*,
*::before,
*::after {
  box-sizing: border-box;
}
body {
  font-size: 30px;
  font-family: "Roboto", sans-serif;
  background-color: #eff5f8;
  color: #001b44;
}
h2 {
  text-align: center;
}
li {
  list-style: none;
  margin-top: 20px;
  display: block;
}
button {
  background-color: #001b44;
  border-radius: 8px;
  color: #fff;
  cursor: pointer;
  font-size: 1.2em;
  letter-spacing: 0.21px;
  line-height: 16px;
  max-width: 255px;
  padding: 20px 52px;
  text-align: center;
  transition-duration: 0.15s;
  transition-property: background-color;
  transition-timing-function: linear;
  margin: 0.6em auto;
  display: block;
}
label {
  color: #212121;
}
select {
  font-size: 30px;
  width: 100%;
  margin-top: 10px;
  background-color: whitesmoke;
}
.card {
  background-color: #FFFFE0;
  border: 1px solid #eceff1;
  border-radius: 8px;
  box-shadow:
          rgba(60, 64, 67, 2) 0 1px 2px 0,
          rgba(60, 64, 67, 0.15) 0 2px 4px 3px;
  margin: 40px 20px;
  padding: 20px;
  width: 40vw;
  float: left;
}
.card::after {
  content: " ";
  display: block;
  clear: both;
}
#msg {
  display: block;
  min-height: 100px;
  width: 100%;
  margin-top: 10px;
  background-color: whitesmoke;
}
@media only screen and (max-width: 800px) {
 .card {
   width: 80vw;
 }
}
```

```javascript
var student = document.getElementById("student-names");
var grade = document.getElementById("grades");
var comment = document.getElementById("msg");
var saveButton = document.getElementById("save");
var savedName = document.getElementById("saved-name");
function saveLastGrade() {
// Save related form data as an object
  var studentGrade = {
    student: student.value,
    grade: grade.value,
    comment: comment.value.trim()
  };
// Use .setItem() to store object in storage
                  and JSON.stringify to convert it as a string
  localStorage.setItem("studentGrade", JSON.stringify
(studentGrade));
}
function renderLastGrade() {
// Use JSON.parse() to convert text to JavaScript object
  var lastGrade = JSON.parse(localStorage.getItem("studentGrade"));
// Check if data is returned, if not exit out of the function
  if (lastGrade !== null) {
  document.getElementById("saved-
name").innerHTML = lastGrade.student;
  document.getElementById("saved-
grade").innerHTML = lastGrade.grade;
  document.getElementById("saved-
comment").innerHTML = lastGrade.comment;
  } else {
    return;
  }
}
saveButton.addEventListener("click", function(event) {
event.preventDefault();
saveLastGrade();
renderLastGrade();
});
// The init() function fires when the page is loaded
function init() {
// When the init function is executed,
                  the code inside renderLastGrade function will also execute
  renderLastGrade();
}
init();
```

Git Pull Request

A **pull request** allows us to collaborate on features after they're finished and before they're merged into a development branch or main codebase. Once a feature branch is finished and ready for review, we can open a pull request and invite team members to review the code, comment on it, and even push their own commits. Pull requests not only provide a chance to catch bugs before merging but also enable developers to discuss new features and any necessary modifications before moving forward.

In this activity, you'll open a pull request for a feature branch. For now, you'll only leave comments on and review your own code. In future activities, you'll learn more about collaboration on GitHub and have the chance to work with others.

Once the review is finished, you'll merge the pull request and delete the feature branch.

Create a New Feature Branch

Once the repository is done cloning to your machine, navigate to the cloned repo's directory by using the `cd` command.

In the directory of this repo, create and switch to a new branch by running the following command:

~~~bash
 git checkout -b 01-my-new-feature
```:
```bash
  git add -A
  git commit -m "<commit message>"
  git push origin 01-my-new-feature
~~~

Open a Pull Request in the Browser

After making these changes to the new feature, you'll need to open a pull request. Navigate to the repo page in your browser. You can click on the "Pull requests" tab at the top of the screen.

This will take you to the pull request hub. On this page, you can open a pull request by clicking on the green "New pull request" button

Now you should see a form for opening a pull request. First, make sure that you are comparing the feature branch that you've been working on to the correct base. In a larger dev environment, you'll likely base your feature branch off of a `dev` branch to avoid corrupting the core codebase.

In this case, because the project is so tiny, you'll compare `01-my-new-feature` to the base branch, `main`.

Give your pull request a descriptive title, and leave comments or reviews for teammates. A comment should include details about the changes in your feature branch that you want to merge.

The side panel offers other optional settings for your pull request. You can assign a reviewer (in this case, you'll review your own work), add assignees, or add labels to help organize your pull requests. These options are especially useful when working on bigger projects, but you can leave them blank for now.

Once you've filled out the form, click on the green "Create pull request" button at the bottom.

Now you're ready to review your pull request!

Review Pull Request

Once you've successfully opened your pull request, select the "Files changed" tab, which should take you to a page where you can view all of the changes that have been made. You can comment on these changes by clicking the plus sign (+) that appears when the cursor points at the code.

You can also begin a review of the changes on this page.

Click on the green "Review changes" button on the right. changes" button, a box will appear. Enter a comment in the comment box (maybe something like, "Looks good to me!").

You can choose to provide feedback without approval, approve the merge, or request changes to be completed before merging. Select the first option, "Comment", and then click the green "Submit review" button.

On submitting, you'll be redirected to the "Conversation" tab, where you'll see the comment that you just left in your review, a history of commits, and a button to merge the pull request.

Merge and Close Pull Request

Now that you've left a review and you don't need to make any more changes to your feature branch, you can merge and close your pull request!

Select the green "Merge pull request" button. Then you should get a request asking you to confirm the merge.

Select the green "Confirm merge" button.

Success!

At this point, Git offers the option to delete the remote feature branch. We're finished with the feature, so go ahead and select "Delete branch".

Great! Now let's delete the feature branch locally.

Delete Local Branch

Imagine how many branches you might open when working on a larger project with many features. To stay organized, it's good practice to delete a local branch once you've merged your feature.

You've already deleted the remote feature branch in the browser, so now you'll delete the local feature branch.

Navigate to the repo directory in your CLI, if you aren't there already. Run the following commands to switch to `main`, pull changes, and begin your next feature:

```bash
git checkout main
git pull origin main
git branch -d 01-my-new-feature
git checkout -b 02-my-next-feature
```