

Coding Academy Challenge

Hi There!

Thanks for expressing your interest in our special bootcamp program, we wish you a great success and hope to see you in our class soon!

We have experienced the very basics of HTML, CSS and programming in Javascript, and we have used MS Code (<https://code.visualstudio.com/>) as our editing application.

Your Challenge

Through this challenge you will expand your knowledge of those languages, create a page of your own with your personal details and also have a better memory game that keeps the highest score, being able to restart the game, and more.

HTML

- HTML is in charge of the structure of our page.
- It is a markup language where each element has an opening tag and a closing tag, for example:

```
<p> I am a paragraph and I'm proud of it </p>
```

CSS

- CSS is in charge of the look of our page
- We put our CSS in a different file
- With CSS we use [selectors](#) to select some [elements](#) on the page and apply some [styling rules](#) to them
- For example:

```
/* Selects all <h1> elements and make their color blue*/
h1 {
  color: blue;
}
/* Selects elements that have the class "box" */
.box {
  background-color: green;
}

/* Selects elements that have the class "back" and are contained in
an element that have both classed: "card" and "flipped" - and hide
it */
.card.flipped .back{
  display:none;
}
```

Javascript

Javascript is a programming language. It is in charge of the behavior of our page. Many modern applications relay heavily on Javascript.

- For example:

```
var age = prompt('Whats your age dude?');
if (age > 18) {
    alert('Carry on');
} else {
    alert('Too Young');
}
```

- Variable is like a box with a name in which we can store a value
- We use conditions to implement our desired logic
- `prompt` is a built-in function that we can use to get input from the user
- `alert` is also built-in function in Javascript
- We can write our own functions
- We can bind a click event from the HTML to a function we created
- For example:

```
<!-- In the HTML: -->
<button onclick="sayHello()">Feeling Lonely?</button>

// In the Javascript:
function sayHello() {
    var userName = prompt('Whats your Name?');
    alert('Hello ' + userName + '!');
}
```

Task 1 – Take a deep breath

Review the current game code.

- Play with it
- Take the time to understand things you don't understand
- Make some simple CSS changes such as colors
- I've added some sounds to the sounds folder, use them:
 - when user is wrong play the wrong.mp3
 - when user got it right, play the right.mp3

Task 2 – Homepage and Navigation

Create a homepage and navigation

1. Create an *index.html* page, this should be your homepage with your details:

- a. Name
- b. Picture
- c. Phone
- d. Email (use a *mailto link* , so it's become clickable and easy to email you)
- e. Hobbies, Dreams, or anything else

Learn the following HTML elements and use them in your solution:

- `` Span
- `<div>` Division
- `<a>` Anchor
- `` & `` Unordered List & List Item
- `` Image
- Headings `<h1>`, `<h2>`...
- Line Break `
`
- Horizontal Line `<hr />`
- `<table>` Table
- button

Learn the following CSS rules and use them in your solution:

- font-size
 - padding & margin & border
 - width & height
 - border-radius
 - position: absolute
2. Create a link to the memory game page
 3. In the memory game, add a link back to the home page

Task 3 – Logic

HTML5 *localStorage*, is a simple mechanism that allow us to keep some data in the browser, this data is kept until the user clear its history (just like a cookie).

You can view the *localStorage* by opening the developer console in chrome (F12 will do that):



Using the localStorage from Javascript is easy enough. For example:

```
localStorage.setItem('schoolName', 'Coding Academy');
var schoolName = localStorage.getItem('schoolName');
alert(schoolName);
```

- Tip: You can simply copy and paste this code to a developer-tools console and see it in action.
- When page load, prompt the user for its name and save it to localStorage
- When the user clicks the first card, clock starts
(Note: You are not requested to show the actual clock to the user)
- For representing time in programming languages we often use timestamps
- In Javascript, a timestamp is the amount of **milliseconds** passed from 1970, see here: <http://www.currenttimestamp.com/>
Hint: use `Date.now()` to get the current milliseconds.
Hint: Add a global variable to hold the game's start time
- When the user wins, clock stops
- Calculate how much time has passed by subtracting 2 timestamps
- Save the best time in the localStorage
Clue: you will need to compare 2 timestamps
Clue: you may need to use `parseInt` to convert strings to numbers
- Show the best time also on the page itself
Clue: You should use something like: `document.querySelector('.bestTime')` to access an element on the page and change its content.

Task 4 – Replay the Game

Let's make it so we can replay the game!

- When user wins, display a **Play Again** button
 - You can show or hide an element by changing its style, see here: <https://css-tricks.com/snippets/javascript/showhide-element/>

- You need to reset some variables
- You need to flip all the cards
 - To be able to flip the cards we need to be able to select all the cards and remove their flipped class
 - This will require a loop such as:

```
var divs = document.querySelectorAll('div');  
  
for (var i = 0; i < divs.length; ++i) {  
    divs[i].style.color = "green";  
}
```

Task 5 – Finalize

- Make the game look nice
- Clean up the code
- Make it so the back of the card hides the monster completely

Task 5 – Bonus

Get some bonus points!

Fix the Bug

- If we click fast, we can flip more than 2 cards, that's a bug, fix it.
Hint: use another variable: `isProcessing` that we set to true while handling a user action.

Shuffle the cards every game

- this will require code such as:

```
var board = document.querySelector('.board');  
for (var i = board.children.length; i >= 0; i--) {  
    board.appendChild(board.children[Math.random() * i | 0]);  
}
```

- Wrap this code in a function so you can run it when user wins.

Remember the player name

- When page loads check the localStorage to see if we know his name, if not:
 - Prompt for the player name
 - Use localStorage to store the player name
- Show the player name on the page
- Show a "Not You?" button
 - Clicking it should prompt to ask for the player name.

Start getting yourself ready for the Bootcamp!

Expand your knowledge, step by step, by completing as much as you can of the following tutorials:

HTML

- <http://www.w3schools.com/html/>
- <https://developer.mozilla.org/en-US/Learn/HTML>

CSS

- <http://www.w3schools.com/css>
- <https://developer.mozilla.org/en-US/Learn/CSS>

Javascript

- <http://www.w3schools.com/js>
- https://developer.mozilla.org/en-US/Learn/Getting_started_with_the_web/JavaScript_basics