Javascript Examples

Let's see a couple of examples

Last 2 sessions we talked about Javascript

Today:

- Review Javascript basics by going through some examples
- In the process we will also learn some new Javascript functionality

Example 1: Add two numbers

We want to create a web page that looks like this:

X + Y = ?

When the user enters numbers in text boxes and clicks the button, the sum of the two numbers will appear:

123 + 4431 = 4554

HTML

First let's create the HTML structure and some simple CSS

We need to create two textboxes

And a button

And a div to show the result

HTML

First let's create the HTML structure and some simple CSS

We need to create two textboxes

```
<input type="text" id="first">
<input type="text" id="second">
```

And a button

```
<button>Show me the sum!</button>
```

And a div to show the result

```
<div id="result">X + Y = ?</div>
```

CSS

Now that we have the HTML elements we needed, let's apply some css styling to:

- bring everything to the center of the page and 200px from the top
- show the result under the text boxes and make it bigger

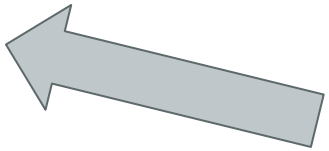
CSS

Now that we have the HTML elements we needed, let's apply some css styling to:

- bring everything to the center of the page and 200px from the top
- show the result under the text boxes and make it bigger

```
.container {
  margin: 200px auto;
  width: 500px;
}

#result {
  font-size: 2em;
  text-align: center;
  padding-top: 2em;
}
```



Note that I use a container to put all HTML elements inside, and apply margin and width properties to the container

Javascript

Perfect! We created a webpage that looks exactly as we wanted.

But it doesn't do anything! Let's add some functionality to it by using javascript

What was the main purpose of this page?

Javascript

Perfect! We created a webpage that looks exactly as we wanted.

But it doesn't do anything! Let's add some functionality to it by using javascript

What was the main purpose of this page?

Adding two numbers

So let's write a javascript function to add two numbers

But first, we need to create a js file and include it in our HTML page

addTwo function

Our function is named **addTwo** and should accept 2 input numbers, add them up and return the result

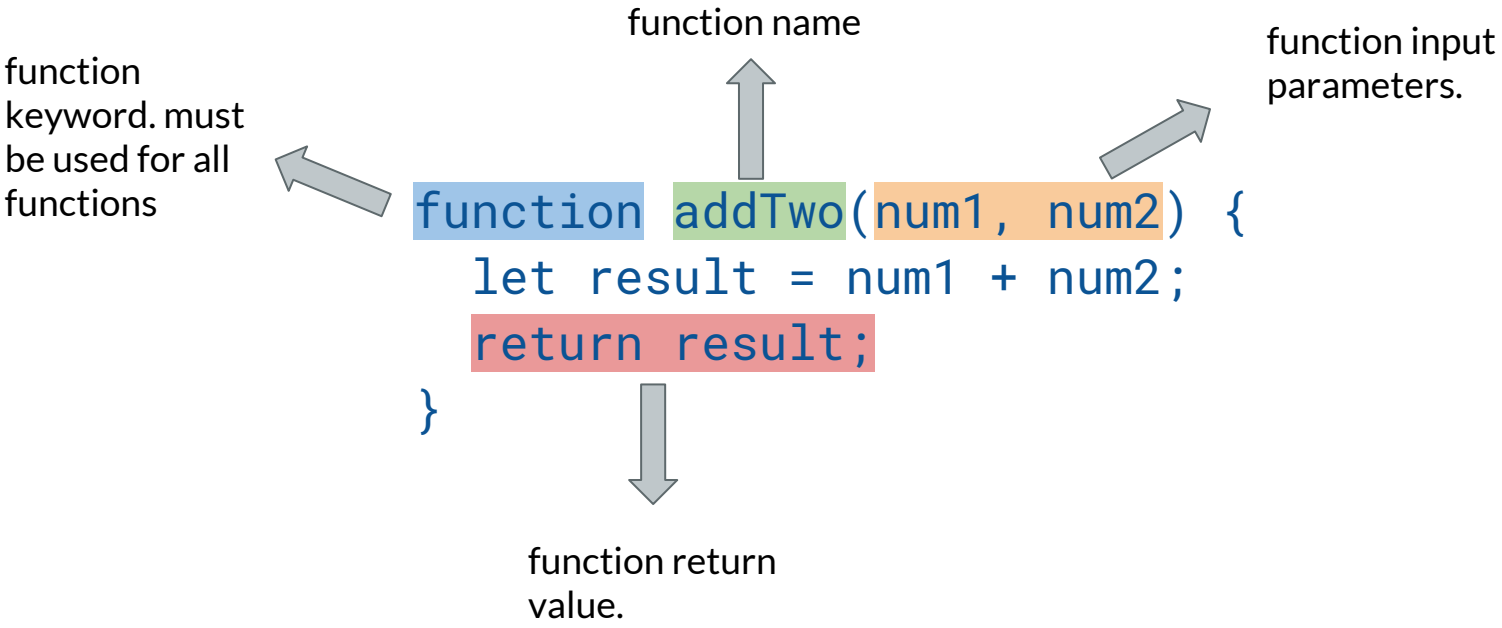
addTwo function

Our function is named **addTwo** and should accept 2 input numbers, add them up and return the result

```
function addTwo(num1, num2) {
  let result = num1 + num2;
  return result;
}
```

addTwo function

Our function is named **addTwo** and should accept 2 input numbers, add them up and return the result



Calling The Function

In order to use the function we just wrote, we should call it. something like this:

```
myResult = addTwo(myValue1, myValue2);
```

Let's test if our function actually works, by using two numbers and showing the result in console:

```
myResult = addTwo(11, 22);

console.log(myResult);
```

Calling The Function

In order to use the function we just wrote, we should call it. something like this:

```
myResult = addTwo(myValue1, myValue2);
```

Let's test if our function actually works, by using two numbers and showing the result in console:

```
myResult = addTwo(11, 22);

console.log(myResult);
```

It Works!

Is that it?

We are pretty sure that our function works! But we only used it for two fixed values and saw the result in console.

What we wanted to do was to **get the input numbers from the text boxes** on our HTML page and **show the result** right below them, **when the button is clicked**.

Last time, we learned that clicking a button fires an event and we can capture that event, by creating an event Listener and attaching it to the button.

Let's do that for this example.

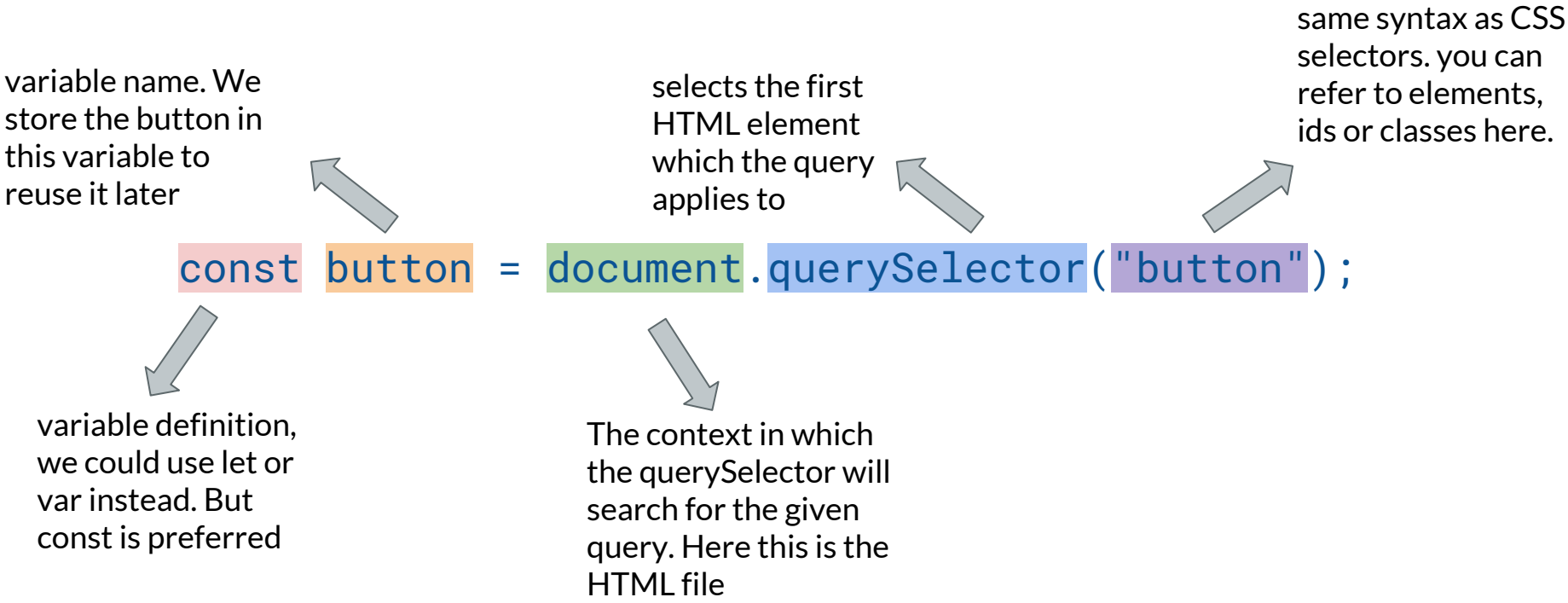
querySelector

First we need to access the button element in our javascript code.

```
const button = document.querySelector("button");
```

querySelector

First we need to access the button element in our javascript code.



addEventListener

Then we need to attach an eventListener to our button

```
const button = document.querySelector("button");
button.addEventListener('click', onClick);
```

This is the object we want to attach the eventListener to.

This is the event we want to capture

This is the name of the function that executes each time the event is fired

onClick Function

Great! So far we made sure that each time the button is clicked, **onClick** function will be executed. But what should **onClick** function actually do?

- read the input values from text boxes
- add them up
- show the result on the web page

onClick Function

1. read the input values from text boxes

```
const num1 = document.querySelector("#first").value;
const num2 = document.querySelector("#second").value;
```

Get the value of the textbox

onClick Function

1. read the input values from text boxes

```
const num1 = document.querySelector("#first").value;
const num2 = document.querySelector("#second").value;
```

2. add them up

```
const res = addTwo(parseInt(num1),parseInt(num2));
```

call the addTwo function with values from textboxes

num1 and num2 are string values. In order for our addTwo function to work, we have to convert them to numbers(integers)

onClick Function

1. read the input values from text boxes

```
const num1 = document.querySelector("#first").value;
const num2 = document.querySelector("#second").value;
```

2. add them up

```
const res = addTwo(parseInt(num1),parseInt(num2));
```

3. show the result on the web page

```
let resultDiv = document.querySelector("#result");
resultDiv.textContent = num1 + " + " + num2 + " = " + res;
```

what are all those + operators? what do they do?

change the text inside the element

onClick Function

```
function onClick() {
  const num1 = document.querySelector("#first").value;
  const num2 = document.querySelector("#second").value;
  const res = addTwo(parseInt(num1),parseInt(num2));
  let resultDiv = document.querySelector("#result");
  resultDiv.textContent = num1 + " + " + num2 + " = " + res;

}
```

We did it!

You can check out the code and how it works here:

http://ww2.cs.fsu.edu/~faizian/cgs3066/sandbox/sumtwo/add.html

Try changing the code to perform other operations on two numbers like subtraction, multiplication, ...

We can even take this a couple of steps further! How about creating a calculator?!!

Example 2: A simple Image Gallery

We want to create a web page that looks like this. It has a number of images. All images are small except one. when the user clicks on any of the small images, it becomes bigger and the current big image becomes small.




HTML

We have a very simple HTML page with 5 images on it. Lets assume when the user first opens the page, the middle picture is big and the rest of them are small.

```
<img class="small" id="img1" src="img/1.jpg" alt="Image 1">
<img class="small" id="img2" src="img/2.jpg" alt="Image 2">
<img class=  "big" id="img3" src="img/3.jpg" alt="Image 3">
<img class="small" id="img4" src="img/4.jpg" alt="Image 4">
<img class="small" id="img5" src="img/5.jpg" alt="Image 5">
```

CSS

Like the previous example let's center the gallery on the page. Also we have to defined CSS rules for small and big images:

```
.container {
  margin: 200px auto;
  width: 1000px;
  height: 500px;
}
```

```
.small {
  margin: 0 10px;
  border: 1px solid black;
  width: 72px;
  height: 48px;
}

.big {
  margin: 0 10px;
  border: 1px solid black;
  width: 360px;
  height: 240px;
}
```

Javascript

Perfect! We created a webpage that looks exactly as we wanted.

But it doesn't do anything! Let's add some functionality to it by using javascript

What was the main purpose of this page?

Switch between big and small images

Each time the user clicks on a small image:

- make the current big image small
- make the clicked image big

So, again we need to do something when an element is clicked. How?

Switch between big and small images

Each time the user clicks on a small image:

- make the current big image small
- make the clicked image big

So, again we need to do something when an element is clicked. How?

Define a function and attach it to the click event of the element(s)

addEventListener

if we only wanted to attach the event to #img1:

addEventListener

if we only wanted to attach the event to #img1:

```
const img1 = document.querySelector("#img1");
img1.addEventListener('click', onClick);
```

But here we have to attach the event to all small images. How?

addEventListener

if we only wanted to attach the event to #img1:

```
const img1 = document.querySelector("#img1");
img1.addEventListener('click', onClick);
```

But here we have to attach the event to all images. How?

```
const images = document.querySelectorAll("img");
for (let i=0; i<images.length; i++) {
  images[i].addEventListener('click', onClick);
}
```

querySelectorAll

because the query returns multiple elements, this will be an array of elements

returns all elements selected by the query

```
const images = document.querySelectorAll("img");
for (let i=0; i<images.length; i++) {
  images[i].addEventListener('click', onClick);
}
```

The for loop iterates through all img elements and adds the onClick eventHandler to their click event

This is how we get the length of the array, which is the number of elements inside it.

onClick Function

Great! So far we made sure that each time an image is clicked, **onClick** function will be executed. But what should **onClick** function actually do?

- change the current big picture to small
- change the clicked image to big

But how can we achieve this effect?

onClick Function

Great! So far we made sure that each time an image is clicked, **onClick** function will be executed. But what should **onClick** function actually do?

- change the current big picture to small
- change the clicked image to big

But how can we achieve this effect?

One way is to toggle the class of each image between big and small

How to access images in Javascript

before we change the class of each image element, we need to access that image

we already know how to access the big image

```
const bigImage = document.querySelector(".big");
```

But how do we access the small image which the user clicked on? does this work?

```
const smallImage = document.querySelector(".small");
```

How to access images in Javascript

before we change the class of each image element, we need to access that image

we already know how to access the big image

```
const bigImage = document.querySelector(".big");
```

But how do we access the small image which the user clicked on? does this work?

```
const smallImage = document.querySelector(".small");
```

This will only return the first image that belongs to class small.
not the clicked image.

onClick Function

we can access the element that caused the click event by using:

```
function onClick(event) {
  const smallImage = event.currentTarget;
  const bigImage = document.querySelector(".big");

  .
  .
  .
}
```

onClick Function

Great! Now let's remove the **big class** from the **current big image** and assign it to **small class**:

```
bigImage.classList.remove('big');
bigImage.classList.add('small');
```

And the opposite for small image:

```
smallImage.classList.remove('small');
smallImage.classList.add('big');
```

onClick Function

Our final onClick function looks like this:

```
function onClick(event) {
  const smallImage = event.currentTarget;
  const bigImage = document.querySelector(".big");

  bigImage.classList.remove('big');
  bigImage.classList.add('small');
  smallImage.classList.remove('small');
  smallImage.classList.add('big');
}
```

We did it again!

You can check out the code and how it works here:

http://ww2.cs.fsu.edu/~faizian/cgs3066/sandbox/gallery/index.html

Try changing the code to achieve the same effect when hovering over images instead of clicking on them. This time you will need to attach the onClick function to **mouseover event** of the images instead of **click event**.

Try this at home and think about how you can change the gallery to do something more fun!