EXAMPLE 52: USING THE DOM TO MODIFY THE HTML OF A PAGE

EXAMPLE 52: USING THE DOM TO MODIFY THE HTML OF A PAGE

We all love Closures!

Closures are SO cool, I just love closures.

Agree?

Yes! No!

DOM = "DOCUMENT-OBJECT-MODEL"

THE DOM IS A WAY OF ACCESSING ANY PART OF THE HTML OF A WEBPAGE AS A JAVASCRIPT OBJECT.

```
var code = document.getElementById("cod
```

DOM = "DOCUMENT-OBJECT-MODEL"

THE DOM IS A WAY OF ACCESSING ANY PART OF THE HTML OF A WEBPAGE AS A JAVASCRIPT OBJECT.

"code"

"code"

```
<script id = "code">
    var code = document.getElementById("code");
    function nullExample() {
        var code = document.getElementById("code");
        if(code != null) {
            console.log("The code in this webpage
contains this html: " + code.innerHTML);
        }
        else {
            console.log("code is null!");
        }
```

DOM = "DOCUMENT-OBJECT-MODEL"

THE DOM IS A WAY OF ACCESSING ANY PART OF THE HTML OF A WEBPAGE AS A JAVASCRIPT OBJECT.

```javascript
        var code = document.getElementById("cod
console.log("The code in this webpage contains this html: "
code.innerHTML);
```

DOM = "DOCUMENT-OBJECT-MODEL"

THE DOM IS A WAY OF ACCESSING ANY PART OF THE HTML OF A WEBPAGE AS A JAVASCRIPT OBJECT.
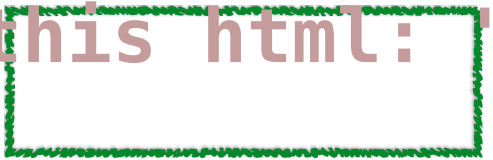
```javascript
var code = document.getElementById("cod
console.log("The code in this webpage contains this html: "
code.innerHTML);
```

DOM = "DOCUMENT-OBJECT-MODEL"

THE DOM IS A WAY OF ACCESSING ANY PART OF THE HTML OF A WEBPAGE AS A JAVASCRIPT OBJECT.

```
var code = document.getElementById("cod
console.log("The code in this webpage contains this html: "
code.innerHTML);
```

WE CAN NOW ACCESS THE HTML VIA AN OBJECT PROPERTY!!

```
var code = document.getElementById("cod
```

IF THERE IS NO HTML ELEMENT WITH ID = "CODE",
THEN `document.getElementById`
RETURNS `null`

We all love Closures!

Closures are SO cool, I just love closures.

Agree?

Yes! No!

THE "YES" BUTTON WILL KEEP TRACK OF THE YES VOTES USING A CLOSURE

WE HAVE 2 BUTTONS IN THE HTML

Yes! No!

```
<button id = "yesButton"> Yes! </b
<button id = "noButton"> No! </but
```

EACH HAS AN ID THAT WE CAN USE FROM THE JAVASCRIPT USING `document.getElementById`

```
window.onload = function(){

  var yesButton = document.getElementById("yesButton");
  var noButton = document.getElementById("noButton");
  var yesVotes = 0;

  noButton.onclick = countNoVotes;

  yesButton.onclick = function() {
    yesVotes++;
    var yesVotesMessage =
document.getElementById("yesVotes");
    yesVotesMessage.innerHTML = "# Yes votes = <b>" +
yesVotes + "</b><br/>";


  }

}

function countNoVotes() {
  noVotes++;
  var noVotesMessage = document.getElementById("noVotes");
  noVotesMessage.innerHTML = "# No votes = <b>" + noVotes +
"</b><br/>";
```

```
window.onload = function(){

    var yesButton = document.getElementById("yesButton");
    var noButton = document.getElementById("noButton");
    var yesVotes = 0;


    noButton.onclick = countNoVotes;


    yesButton.onclick = function() {
        yesVotes++;
        var yesVotesMessage =
document.getElementById("yesVotes");
        yesVotesMessage.innerHTML = "# Yes votes = <b>" +
yesVotes + "</b><br/>";

    }

}

function countNoVotes() {
    noVotes++;
    var noVotesMessage = document.getElementById("noVotes");
    noVotesMessage.innerHTML = "# No votes = <b>" + noVotes
"</b><br/>";
```

WE HAVE A LOCAL (NOT GLOBAL) VARIABLE TO COUNT THE NUMBER OF YES VOTES

AND A NESTED FUNCTION THAT DISPLAYS A MESSAGE WHEN SOMEONE VOTES "YES"

```
    var yesButton = document.getElementById("yesButton");
    var noButton = document.getElementById("noButton");
  var yesVotes = 0;


  noButton.onclick = countNoVotes;


  yesButton.onclick = function() {
    yesVotes++;
    var yesVotesMessage =
document.getElementById("yesVotes");
    yesVotesMessage.innerHTML = "# Yes votes = <b>" +
yesVotes + "</b><br/>";

  }


}

function countNoVotes() {
  noVotes++;
  var noVotesMessage = document.getElementById("noVotes");
  noVotesMessage.innerHTML = "# No votes = <b>" + noVotes
"</b><br/>";
```

CLOSURE = SAY WE HAVE A NESTED + VARIABLES LOCAL TO THE OUTER SCOPE "REFERENCING

THE "YES" BUTTON WILL KEEP TRACK OF THE YES VOTES USING A CLOSURE

```
var yesVotes = 0;

noButton.onclick = countNoVotes;

yesButton.onclick = function() {
    yesVotes++;
    var yesVotesMessage =
document.getElementById("yesVotes");
    yesVotesMessage.innerHTML = "# Yes votes = <b>" +
yesVotes + "</b><br/>";

}
```

<button id = "yesButton"> Yes! </button>
<button id = "noButton"> No! </button>

CLOSURE = SAY WE HAVE A NESTED +

VARIABLES LOCAL TO THE OUTER SCOPE

"REFERENCING

LET'S BUILD THE CODE FOR THIS LITTLE POLL IN 2 WAYS.

We all love Closures!

Closures are SO cool, I just love closures.

Agree?

Yes! No!

THE "YES" BUTTON WILL KEEP TRACK OF THE YES VOTES USING A CLOSURE

LET'S BUILD THE CODE FOR THIS LITTLE POLL IN 2 WAYS.

We all love Closures!

Closures are SO cool, I just love closures.

Agree?

Yes! No!

THE "NO" BUTTON WILL KEEP TRACK OF THE NO VOTES WITHOUT USING CLOSURES

```javascript
    var noVotes = 0;

    window.onload = function(){

        var yesButton = document.getElementById("yesButton");
        var noButton = document.getElementById("noButton");
        var yesVotes = 0;

        noButton.onclick = countNoVotes;

        yesButton.onclick = function() {
          yesVotes++;
          var yesVotesMessage = document.getElementById("yesVotes");
          yesVotesMessage.innerHTML = "# Yes votes = <b>" + yesVotes +
    >";

        }

    }

    function countNoVotes() {
      noVotes++;
      var noVotesMessage = document.getElementById("noVotes");
      noVotesMessage.innerHTML = "# No votes = <b>" + noVotes + "</b>
    }
```

```
var noVotes = 0;

window.onload = function(){

    var yesButton = document.getElementById("yesButton");
    var noButton = document.getElementById("noButton");
    var yesVotes = 0;


    noButton.onclick = countNoVotes;


    yesButton.onclick = function() {
        yesVotes++;
        var yesVotesMessage = document.getElementById("yesVotes");
        yesVotesMessage.innerHTML = "# Yes votes = <b>" + yesVotes + "</b><br/>";

    }


}


function countNoVotes() {
    noVotes++;
    var noVotesMessage = document.getElementById("noVotes");
    noVotesMessage.innerHTML = "# No votes = <b>" + noVotes + "</b><br/>";
}
```

THE "NO" BUTTON WILL KEEP TRACK OF THE NO VOTES WITHOUT USING CLOSURES

TO DO THIS WITHOUT CLOSURES, WE END UP NEEDING A GLOBAL VARIABLE!

We all love Closures!

Closures are SO cool, I just love closures.

Agree?

Yes! No!

\# Yes votes = **20**
\# No votes = **1**

We all love Closures!

Closures are SO cool, I just love closures.

Agree?

[Yes!] [No!]

Yes votes = **20**
No votes = **1**