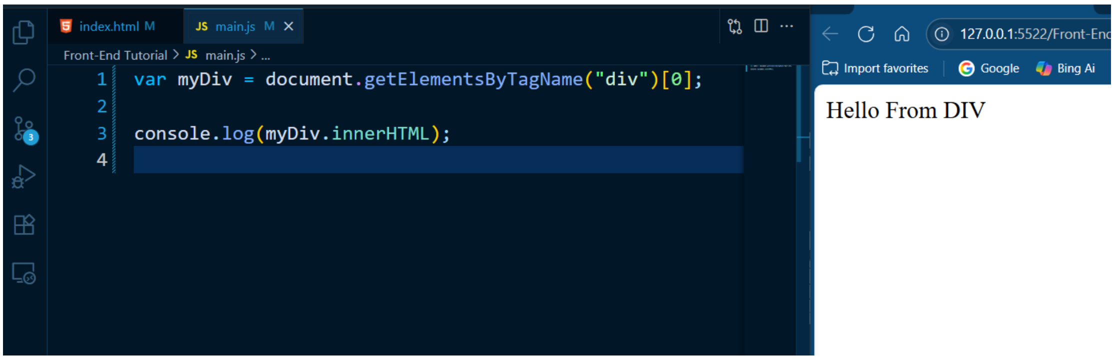
1. Select the first <div> on the page using getElementsByTagName and log its innerHTML.



index.html M · main.js M

Front-End Tutorial > main.js > ...

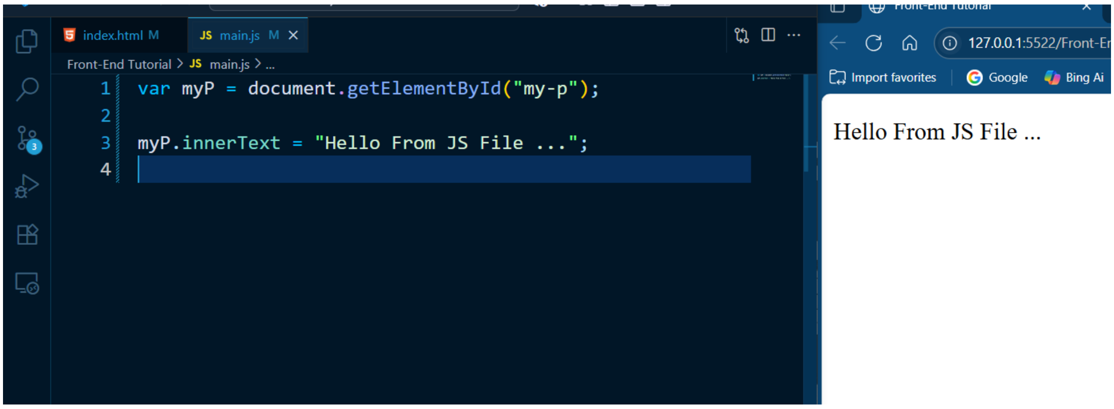
```
var myDiv = document.getElementsByTagName("div")[0];

console.log(myDiv.innerHTML);

```

127.0.0.1:5522/Front-End

Import favorites · Google · Bing Ai

Hello From DIV

2. Using getElementById change the text of #my-p to "Hello DOM".



index.html M · main.js M

Front-End Tutorial > main.js > ...

```
var myP = document.getElementById("my-p");

myP.innerText = "Hello From JS File ...";

```

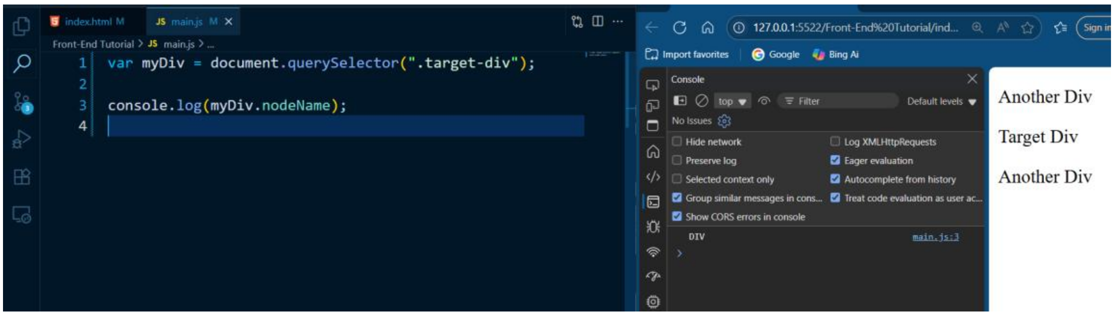
Front-End Tutorial

127.0.0.1:5522/Front-E

Import favorites · Google · Bing Ai

Hello From JS File ...

3. Use querySelector to select the element with class "target-div" and log its nodeName.



index.html M · main.js M

Front-End Tutorial > main.js > ...

```
var myDiv = document.querySelector(".target-div");

console.log(myDiv.nodeName);

```

127.0.0.1:5522/Front-End%20Tutorial/ind...

Import favorites · Google · Bing Ai

Console

top · Filter · Default levels

No Issues

☐ Hide network
☐ Preserve log
☐ Selected context only
☑ Group similar messages in cons...
☑ Show CORS errors in console
☐ Log XMLHttpRequests
☑ Eager evaluation
☑ Autocomplete from history
☑ Treat code evaluation as user ac...

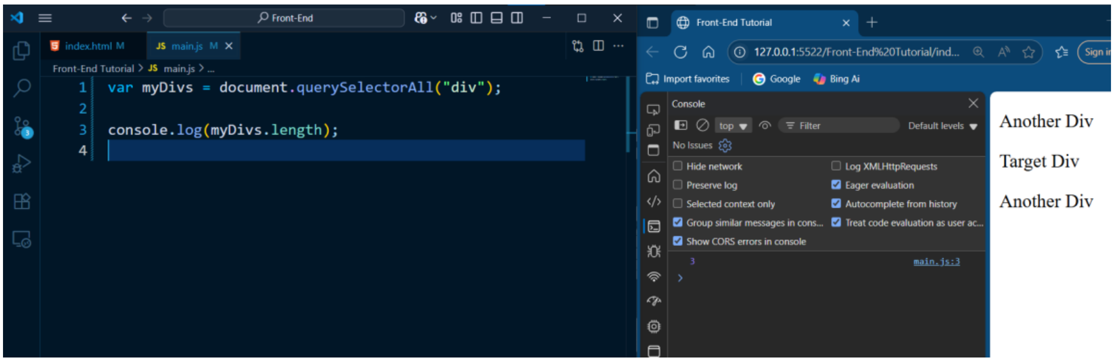
```
DIV                                   main.js:3
```

Another Div

Target Div

Another Div

4. Use querySelectorAll to count how many <div> elements exist; log the count.



```
var myDivs = document.querySelectorAll("div");

console.log(myDivs.length);

```

Console

☐ Hide network ☐ Log XMLHttpRequests
☐ Preserve log ☑ Eager evaluation
☐ Selected context only ☑ Autocomplete from history
☑ Group similar messages in cons... ☑ Treat code evaluation as user ac...
☑ Show CORS errors in console

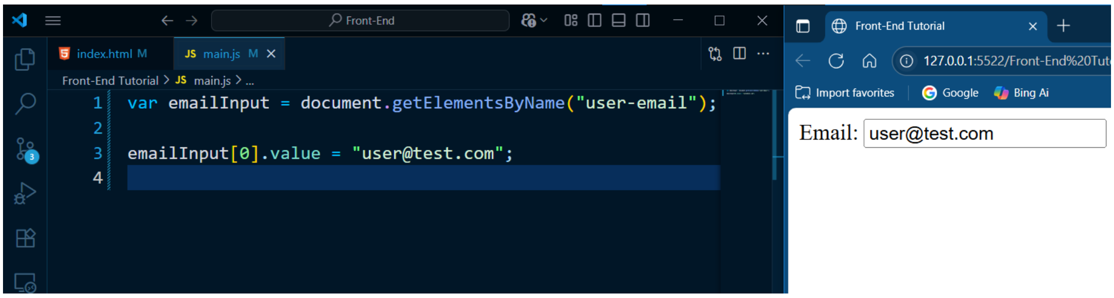
```
3                                   main.js:3
```

Another Div

Target Div

Another Div

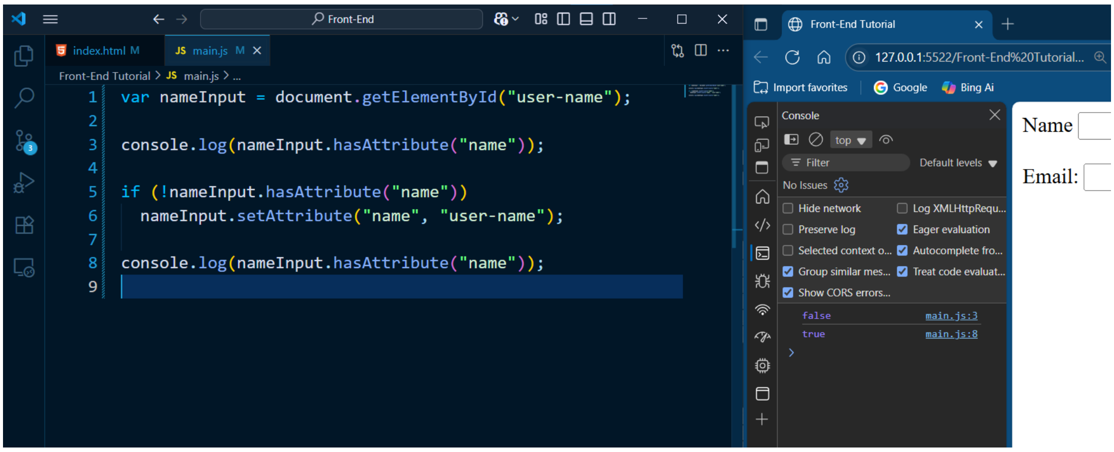
5. Use getElementsByName on the email input (name="user-email") and set its value to "user@test.com".



```
var emailInput = document.getElementsByName("user-email");

emailInput[0].value = "user@test.com";

```

Email: user@test.com

6. Check if the text input has a "name" attribute; if not add name="user-name" via setAttribute.



```
var nameInput = document.getElementById("user-name");

console.log(nameInput.hasAttribute("name"));

if (!nameInput.hasAttribute("name"))
  nameInput.setAttribute("name", "user-name");

console.log(nameInput.hasAttribute("name"));

```

Console

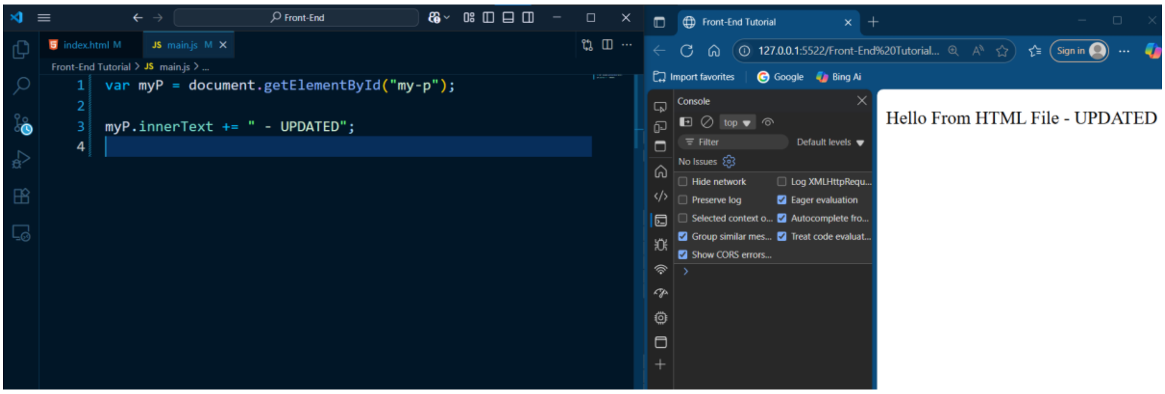
☐ Hide network ☐ Log XMLHttpRequ...
☐ Preserve log ☑ Eager evaluation
☐ Selected context o... ☑ Autocomplete fro...
☑ Group similar mes... ☑ Treat code evaluat...
☑ Show CORS errors...

```
false                 main.js:3
true                  main.js:8
```

Name

Email:

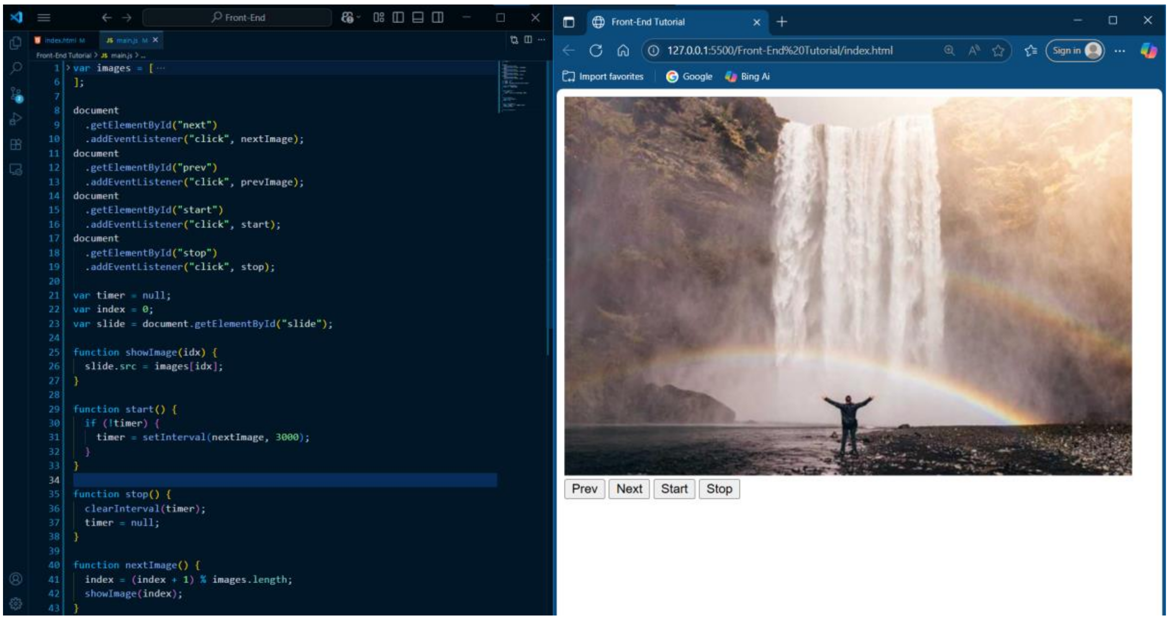
7. Append the string " - UPDATED" to the existing innerText of #my-p (keep original text).



```
var myP = document.getElementById("my-p");

myP.innerText += " - UPDATED";
```

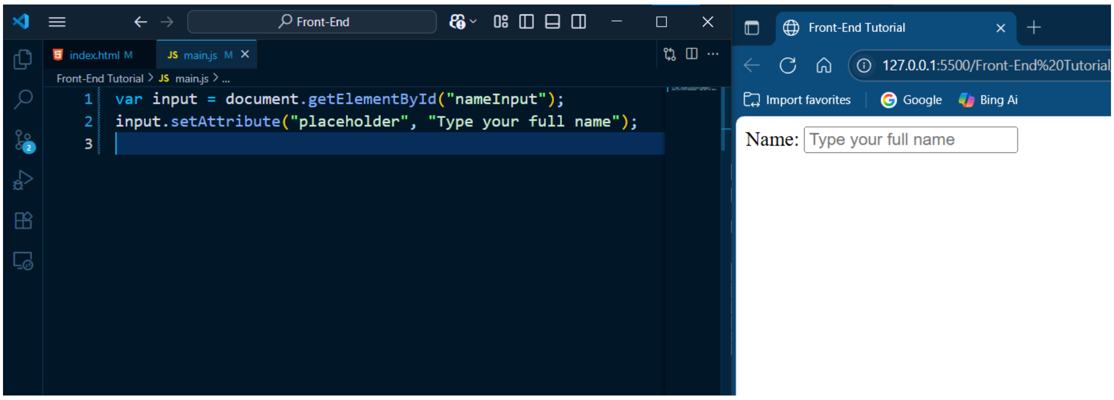
Hello From HTML File - UPDATED

8. Create Images slider that work automatically and with next/prev/start/stop buttons.



```
var images = [···
];

document
  .getElementById("next")
  .addEventListener("click", nextImage);
document
  .getElementById("prev")
  .addEventListener("click", prevImage);
document
  .getElementById("start")
  .addEventListener("click", start);
document
  .getElementById("stop")
  .addEventListener("click", stop);

var timer = null;
var index = 0;
var slide = document.getElementById("slide");

function showImage(idx) {
  slide.src = images[idx];
}

function start() {
  if (!timer) {
    timer = setInterval(nextImage, 3000);
  }
}

function stop() {
  clearInterval(timer);
  timer = null;
}

function nextImage() {
  index = (index + 1) % images.length;
  showImage(index);
}
```

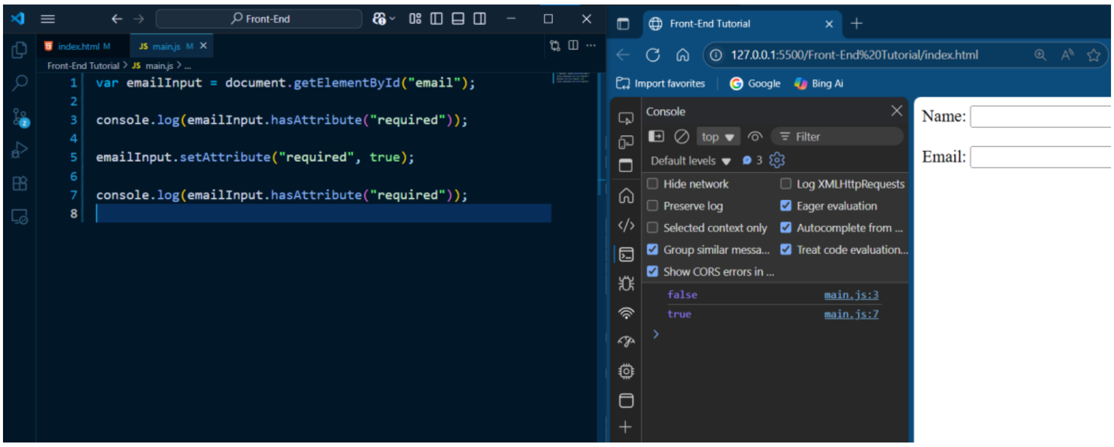
9. Set the placeholder of the text input to "Type your full name" using setAttribute.



```javascript
var input = document.getElementById("nameInput");
input.setAttribute("placeholder", "Type your full name");
```

Name: Type your full name

10. Log whether the email input has attribute "required"; if missing add it.



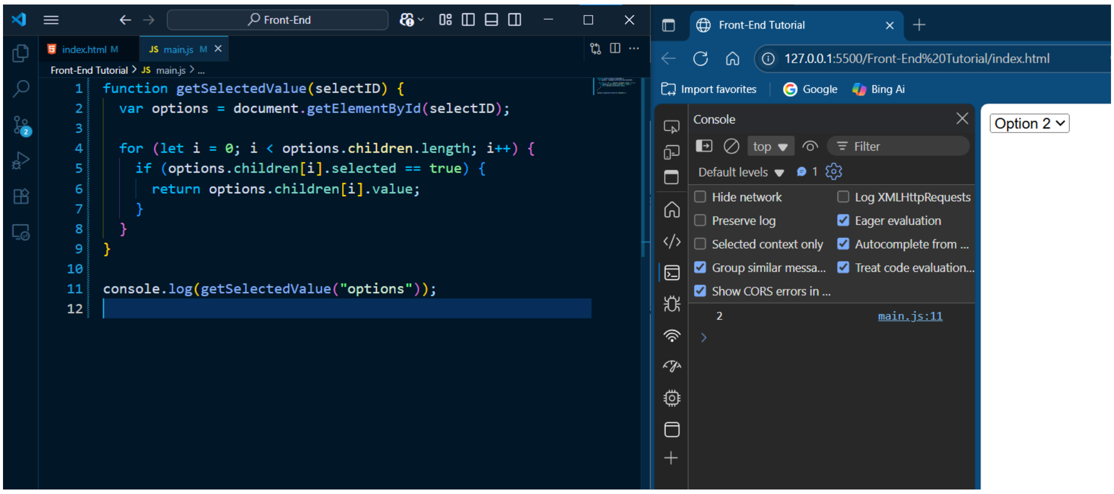
```javascript
var emailInput = document.getElementById("email");

console.log(emailInput.hasAttribute("required"));

emailInput.setAttribute("required", true);

console.log(emailInput.hasAttribute("required"));
```

```
false                                   main.js:3
true                                    main.js:7
```

Name:

Email:

11. Write function getSelectedValue(selectId) returning the current selected option value.

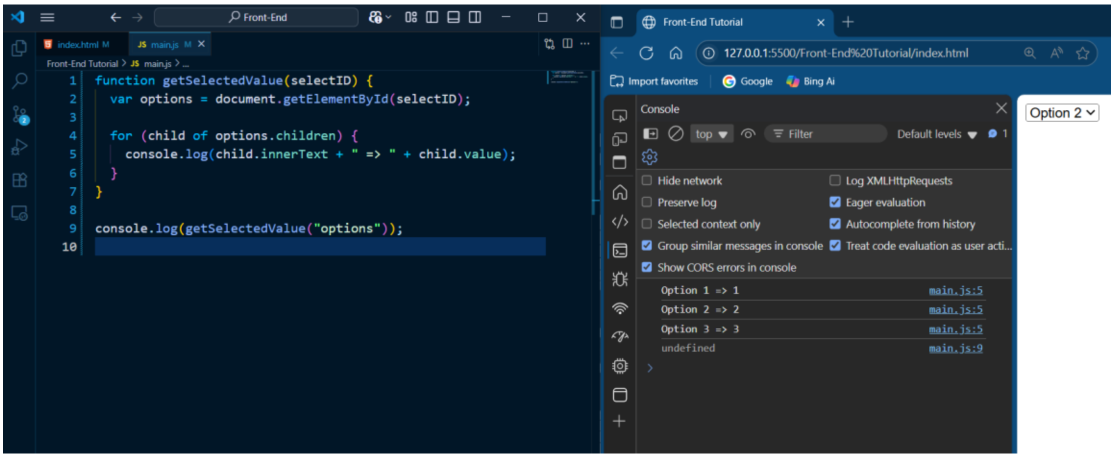


```javascript
function getSelectedValue(selectID) {
  var options = document.getElementById(selectID);

  for (let i = 0; i < options.children.length; i++) {
    if (options.children[i].selected == true) {
      return options.children[i].value;
    }
  }
}

console.log(getSelectedValue("options"));
```

```
2                                       main.js:11
```

Option 2

12. Loop through all options of the select and log each option's text and value.

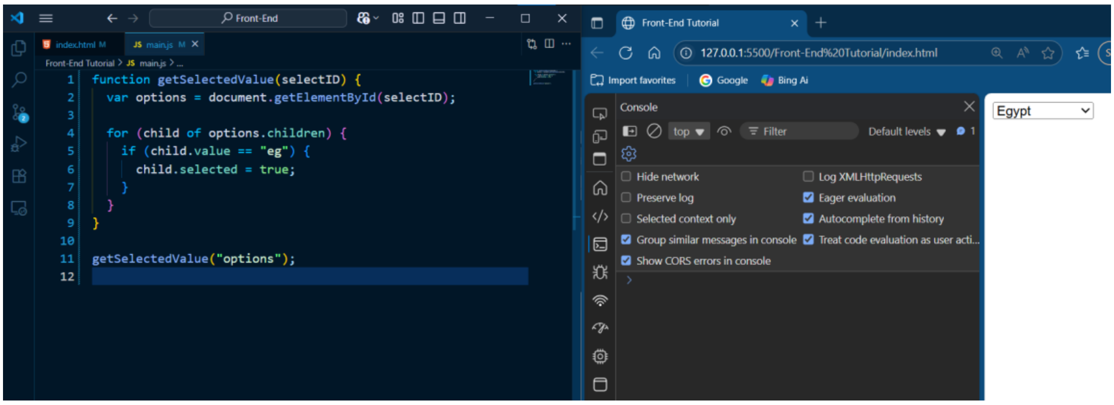


```
function getSelectedValue(selectID) {
  var options = document.getElementById(selectID);

  for (child of options.children) {
    console.log(child.innerText + " => " + child.value);
  }
}

console.log(getSelectedValue("options"));
```

□ Hide network　□ Log XMLHttpRequests
□ Preserve log　☑ Eager evaluation
□ Selected context only　☑ Autocomplete from history
☑ Group similar messages in console　☑ Treat code evaluation as user acti...
☑ Show CORS errors in console

```
Option 1 => 1                                   main.js:5
Option 2 => 2                                   main.js:5
Option 3 => 3                                   main.js:5
undefined                                       main.js:9
```

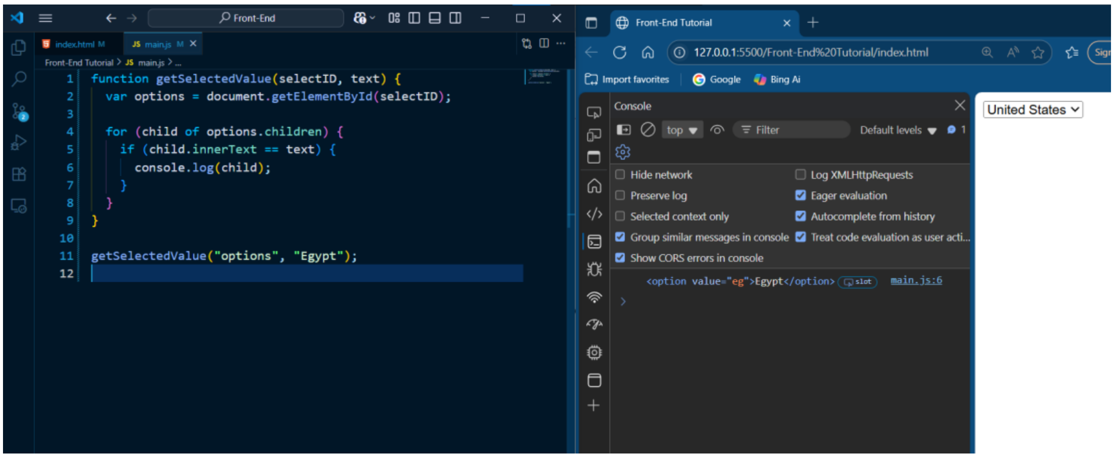
13. Programmatically select the option with value "EG".



```
function getSelectedValue(selectID) {
  var options = document.getElementById(selectID);

  for (child of options.children) {
    if (child.value == "eg") {
      child.selected = true;
    }
  }
}

getSelectedValue("options");
```

□ Hide network　□ Log XMLHttpRequests
□ Preserve log　☑ Eager evaluation
□ Selected context only　☑ Autocomplete from history
☑ Group similar messages in console　☑ Treat code evaluation as user acti...
☑ Show CORS errors in console

14. Create function selectByText(selectId, text) that selects the first option whose text matches exactly.

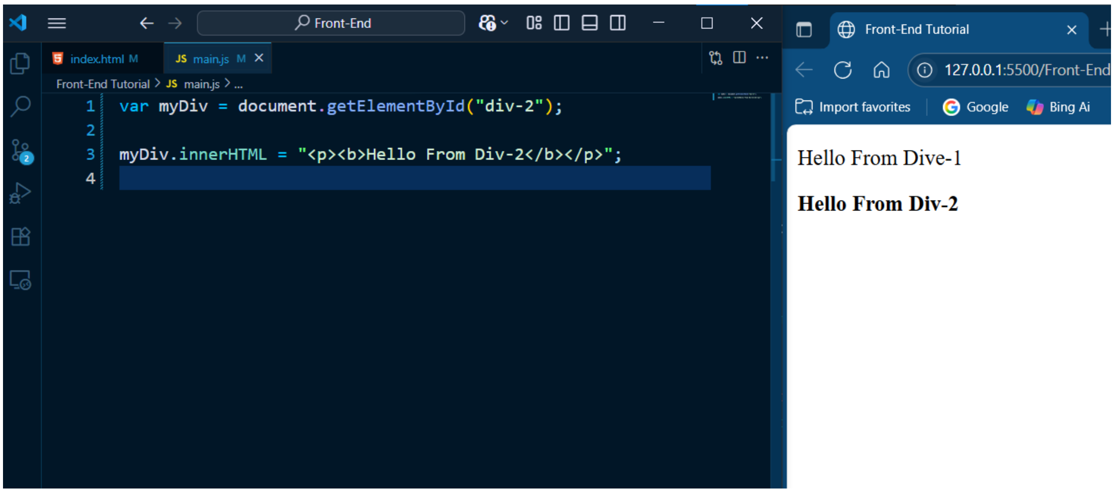


```
function getSelectedValue(selectID, text) {
  var options = document.getElementById(selectID);

  for (child of options.children) {
    if (child.innerText == text) {
      console.log(child);
    }
  }
}

getSelectedValue("options", "Egypt");
```

□ Hide network　□ Log XMLHttpRequests
□ Preserve log　☑ Eager evaluation
□ Selected context only　☑ Autocomplete from history
☑ Group similar messages in console　☑ Treat code evaluation as user acti...
☑ Show CORS errors in console

```
<option value="eg">Egypt</option> slot          main.js:6
```

15. Replace innerHTML of #div-2 with a <p><b>Bold Text</b></p> (ensure bold renders, not printed literally).

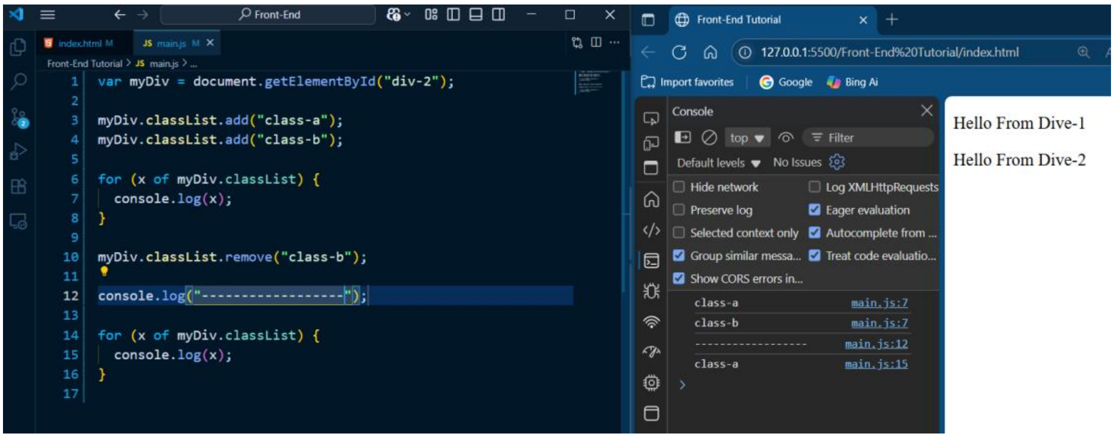


```
var myDiv = document.getElementById("div-2");

myDiv.innerHTML = "<p><b>Hello From Div-2</b></p>";
```

Hello From Dive-1

Hello From Div-2

16. Add classes class-a and class-b to #div-2 then remove class-b (using classList).



```
var myDiv = document.getElementById("div-2");

myDiv.classList.add("class-a");
myDiv.classList.add("class-b");

for (x of myDiv.classList) {
  console.log(x);
}

myDiv.classList.remove("class-b");

console.log("------------------");

for (x of myDiv.classList) {
  console.log(x);
}

```

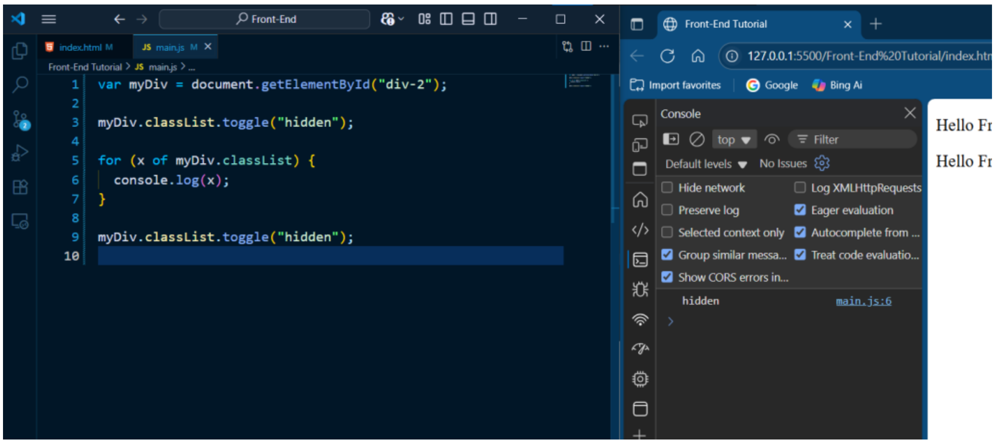
□ Hide network ☐ Log XMLHttpRequests
□ Preserve log ☑ Eager evaluation
□ Selected context only ☑ Autocomplete from ...
☑ Group similar messa... ☑ Treat code evaluatio...
☑ Show CORS errors in...

```
class-a                 main.js:7
class-b                 main.js:7
------------------      main.js:12
class-a                 main.js:15
```

Hello From Dive-1

Hello From Dive-2

17. Toggle class "hidden" on #div-2 twice; comment final visibility.



```
var myDiv = document.getElementById("div-2");

myDiv.classList.toggle("hidden");

for (x of myDiv.classList) {
  console.log(x);
}

myDiv.classList.toggle("hidden");
```

Console

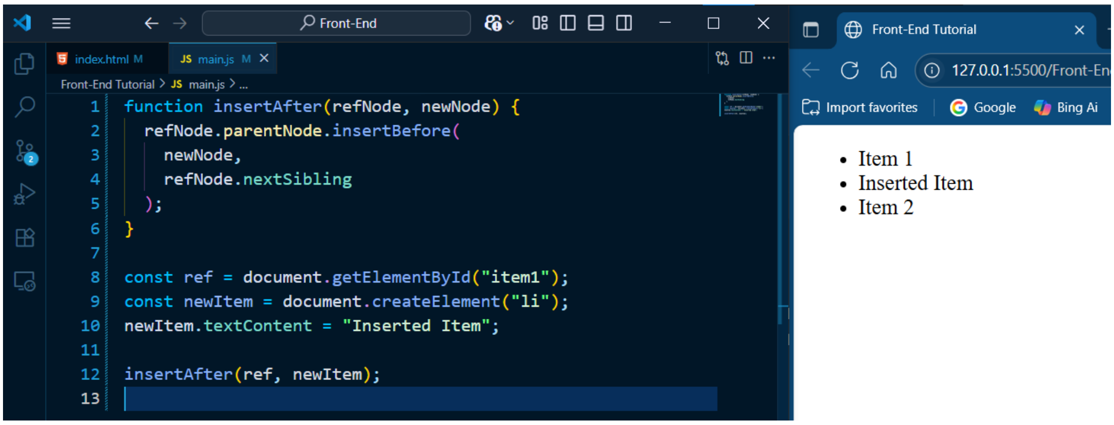
☐ Hide network ☐ Log XMLHttpRequests
☐ Preserve log ☑ Eager evaluation
☐ Selected context only ☑ Autocomplete from ...
☑ Group similar messa... ☑ Treat code evaluatio...
☑ Show CORS errors in...

```
hidden                                   main.js:6
```

Hello Fr

Hello Fr

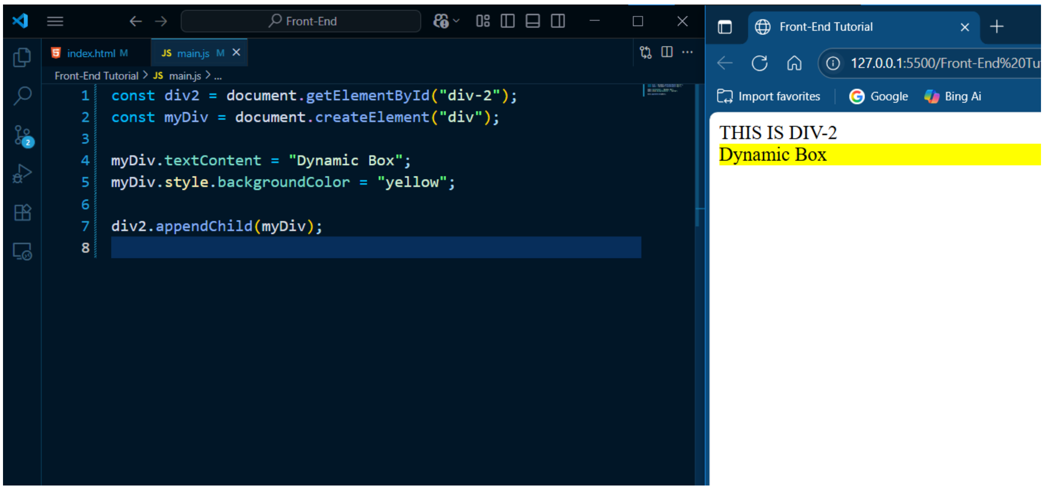
18. Create function insertAfter(refNode, newNode) that inserts newNode immediately after refNode.



```
function insertAfter(refNode, newNode) {
  refNode.parentNode.insertBefore(
    newNode,
    refNode.nextSibling
  );
}

const ref = document.getElementById("item1");
const newItem = document.createElement("li");
newItem.textContent = "Inserted Item";

insertAfter(ref, newItem);
```

- Item 1
- Inserted Item
- Item 2

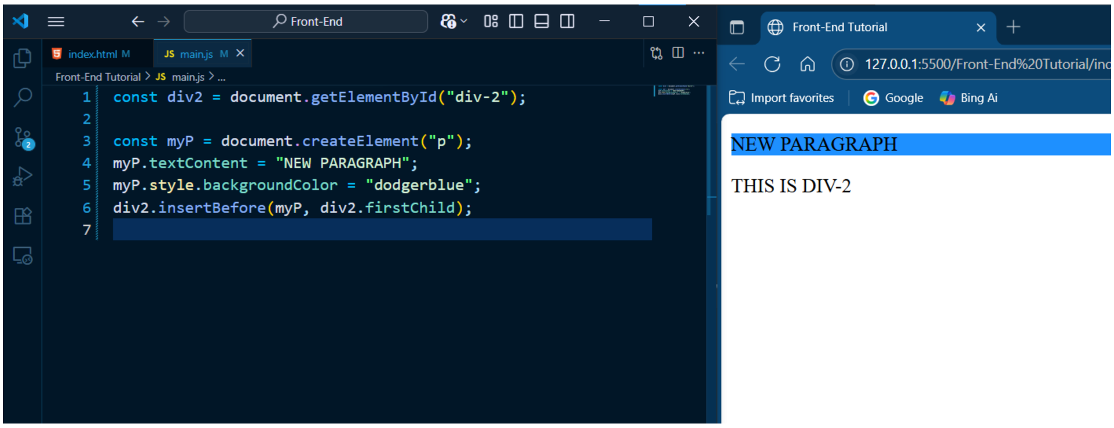
19. Create a new <div> saying "Dynamic Box" with yellow background and append inside #div-2.



```
const div2 = document.getElementById("div-2");
const myDiv = document.createElement("div");

myDiv.textContent = "Dynamic Box";
myDiv.style.backgroundColor = "yellow";

div2.appendChild(myDiv);
```

THIS IS DIV-2
Dynamic Box

20. Insert a new <p> BEFORE the first child of #div-2 (insertBefore).

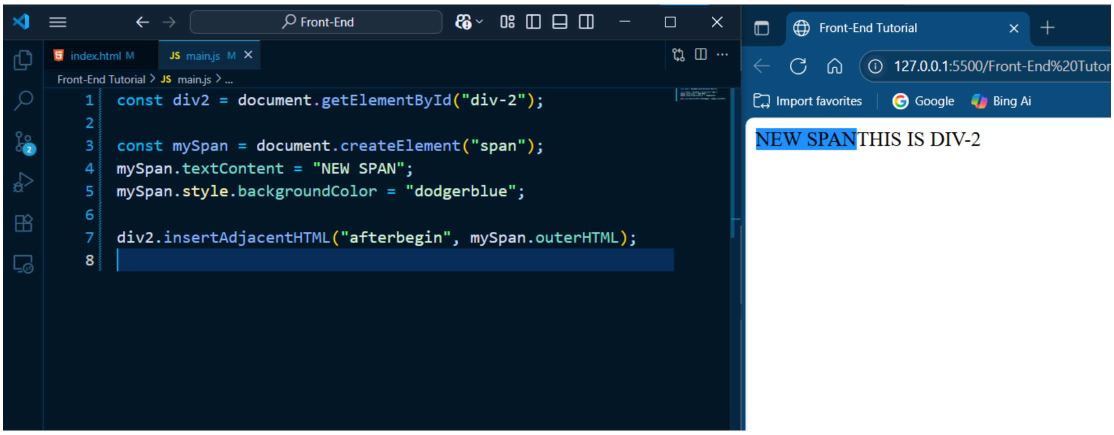


```
const div2 = document.getElementById("div-2");

const myP = document.createElement("p");
myP.textContent = "NEW PARAGRAPH";
myP.style.backgroundColor = "dodgerblue";
div2.insertBefore(myP, div2.firstChild);
```

NEW PARAGRAPH

THIS IS DIV-2

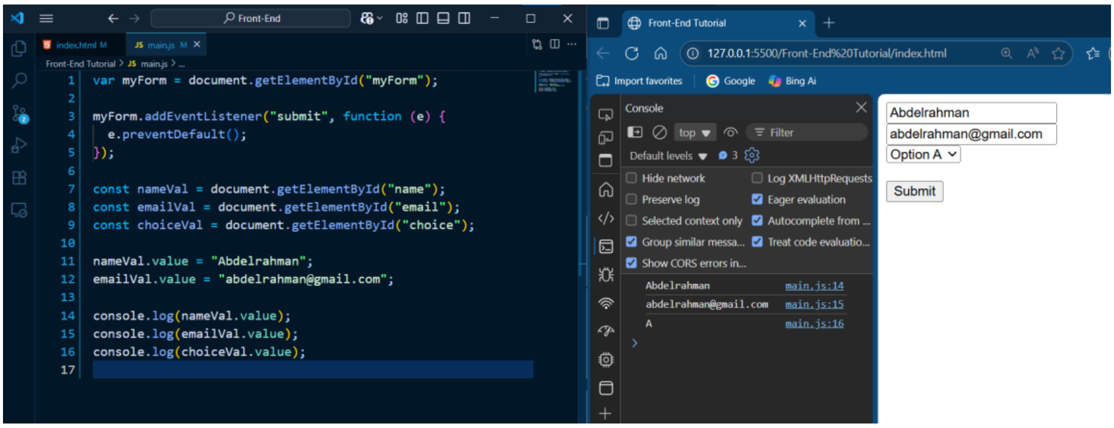
21. Insert a <span> with text "AFTER END" right after #div-2 using insertAdjacentHTML.



```
const div2 = document.getElementById("div-2");

const mySpan = document.createElement("span");
mySpan.textContent = "NEW SPAN";
mySpan.style.backgroundColor = "dodgerblue";

div2.insertAdjacentHTML("afterbegin", mySpan.outerHTML);
```

NEW SPANTHIS IS DIV-2

22. Form onsubmit: prevent default and log values of text, email, and select inputs.



```
var myForm = document.getElementById("myForm");

myForm.addEventListener("submit", function (e) {
  e.preventDefault();
});

const nameVal = document.getElementById("name");
const emailVal = document.getElementById("email");
const choiceVal = document.getElementById("choice");

nameVal.value = "Abdelrahman";
emailVal.value = "abdelrahman@gmail.com";

console.log(nameVal.value);
console.log(emailVal.value);
console.log(choiceVal.value);
```

```
Abdelrahman                main.js:14
abdelrahman@gmail.com      main.js:15
A                          main.js:16
```

Abdelrahman

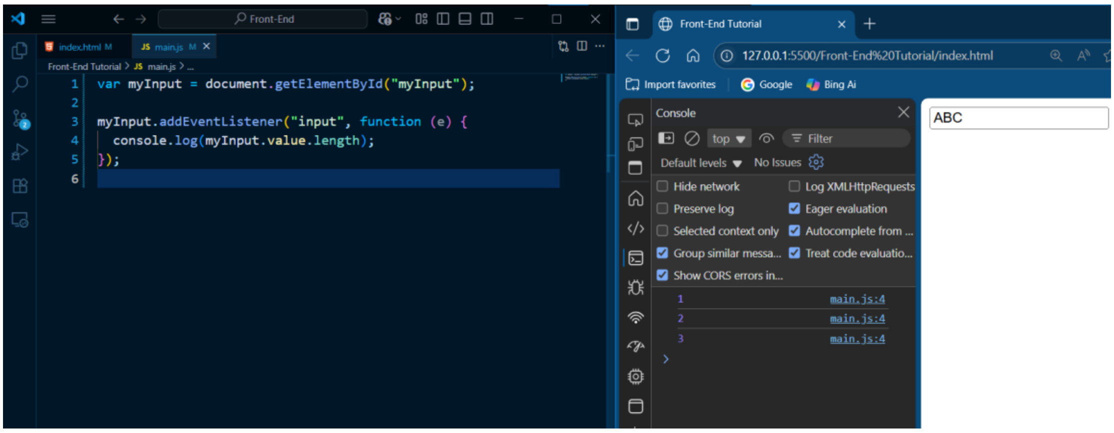
abdelrahman@gmail.com

Option A

Submit

☐ Hide network
☐ Log XMLHttpRequests
☐ Preserve log
☑ Eager evaluation
☐ Selected context only
☑ Autocomplete from ...
☑ Group similar messa...
☑ Treat code evaluatio...
☑ Show CORS errors in...

23. Add input event on the text input that logs its length whenever it changes.



Front-End Tutorial > main.js > ...

```
var myInput = document.getElementById("myInput");

myInput.addEventListener("input", function (e) {
  console.log(myInput.value.length);
});
```

127.0.0.1:5500/Front-End%20Tutorial/index.html

ABC

Console

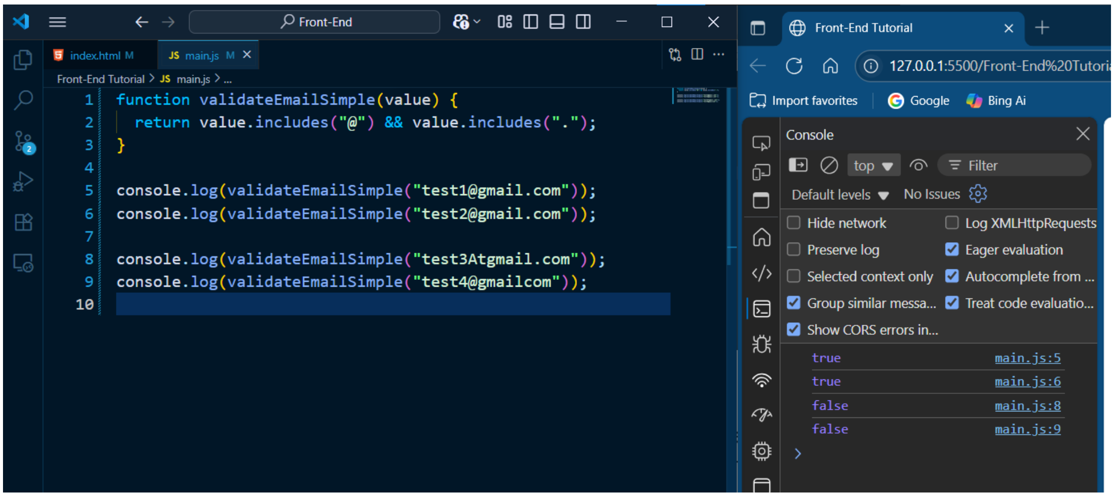
top ▾ Filter

Default levels ▾ No Issues

☐ Hide network ☐ Log XMLHttpRequests
☐ Preserve log ☑ Eager evaluation
☐ Selected context only ☑ Autocomplete from ...
☑ Group similar messa... ☑ Treat code evaluatio...
☑ Show CORS errors in...

```
1                    main.js:4
2                    main.js:4
3                    main.js:4
```

24. Write validateEmailSimple(value) returning true if value contains both '@' and '.'; add 2 passing / 2 failing examples (comments).



Front-End Tutorial > main.js > ...

```
function validateEmailSimple(value) {
  return value.includes("@") && value.includes(".");
}

console.log(validateEmailSimple("test1@gmail.com"));
console.log(validateEmailSimple("test2@gmail.com"));

console.log(validateEmailSimple("test3Atgmail.com"));
console.log(validateEmailSimple("test4@gmailcom"));
```

127.0.0.1:5500/Front-End%20Tutori

Console

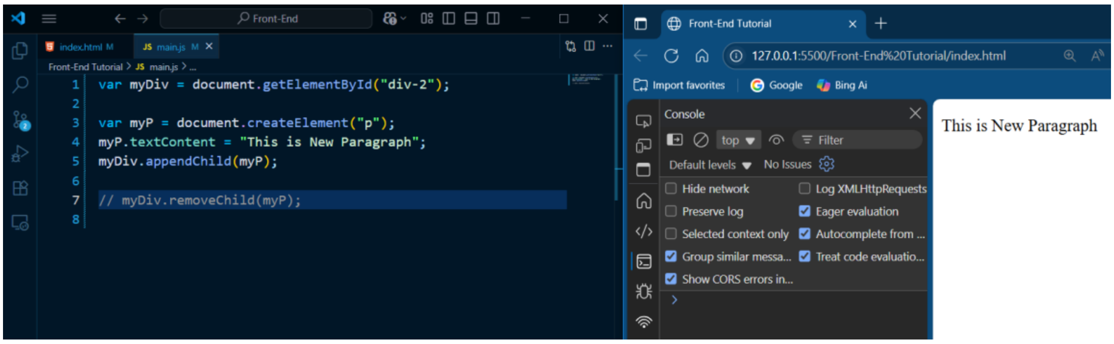
top ▾ Filter

Default levels ▾ No Issues

☐ Hide network ☐ Log XMLHttpRequests
☐ Preserve log ☑ Eager evaluation
☐ Selected context only ☑ Autocomplete from ...
☑ Group similar messa... ☑ Treat code evaluatio...
☑ Show CORS errors in...

```
true                 main.js:5
true                 main.js:6
false                main.js:8
false                main.js:9
```

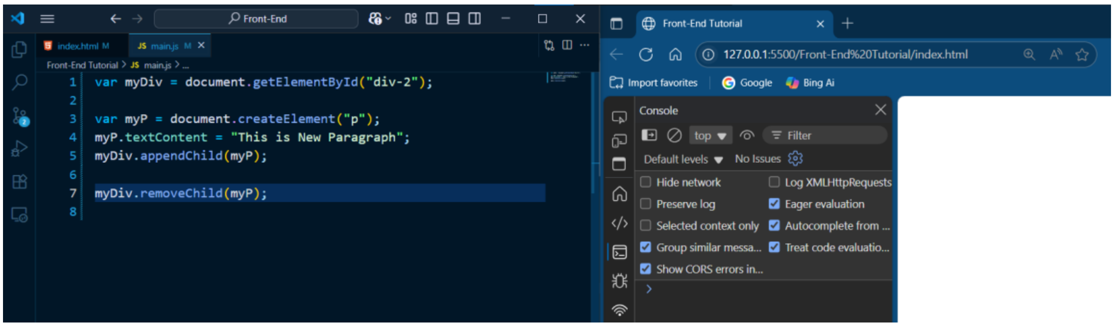
25. Create an element, append it to #div-2, then remove it using parent.removeChild(child).



```
var myDiv = document.getElementById("div-2");

var myP = document.createElement("p");
myP.textContent = "This is New Paragraph";
myDiv.appendChild(myP);

// myDiv.removeChild(myP);
```

☐ Hide network ☐ Log XMLHttpRequests
☐ Preserve log ☑ Eager evaluation
☐ Selected context only ☑ Autocomplete from ...
☑ Group similar messa... ☑ Treat code evaluatio...
☑ Show CORS errors in...

This is New Paragraph



```
var myDiv = document.getElementById("div-2");

var myP = document.createElement("p");
myP.textContent = "This is New Paragraph";
myDiv.appendChild(myP);

myDiv.removeChild(myP);
```

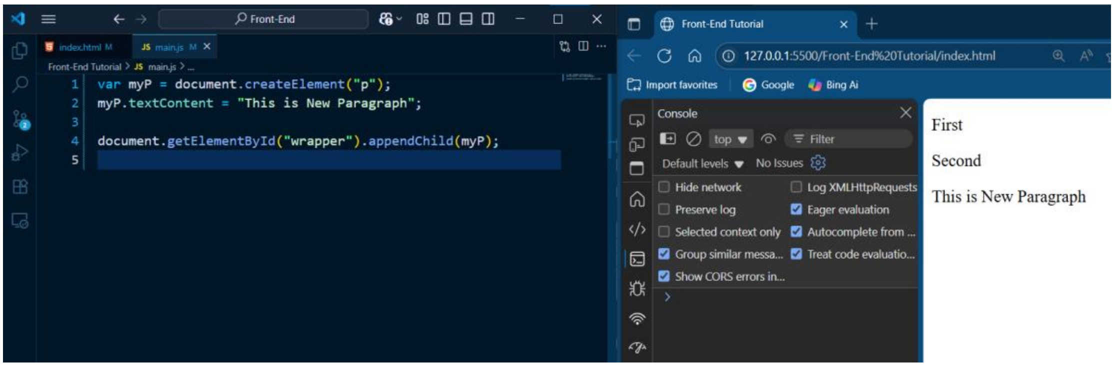
☐ Hide network ☐ Log XMLHttpRequests
☐ Preserve log ☑ Eager evaluation
☐ Selected context only ☑ Autocomplete from ...
☑ Group similar messa... ☑ Treat code evaluatio...
☑ Show CORS errors in...

25. Create an element, append it to <div id="wrapper">

<p id="first">First</p>

<p id="second">Second</p>

</div>



```
var myP = document.createElement("p");
myP.textContent = "This is New Paragraph";

document.getElementById("wrapper").appendChild(myP);
```

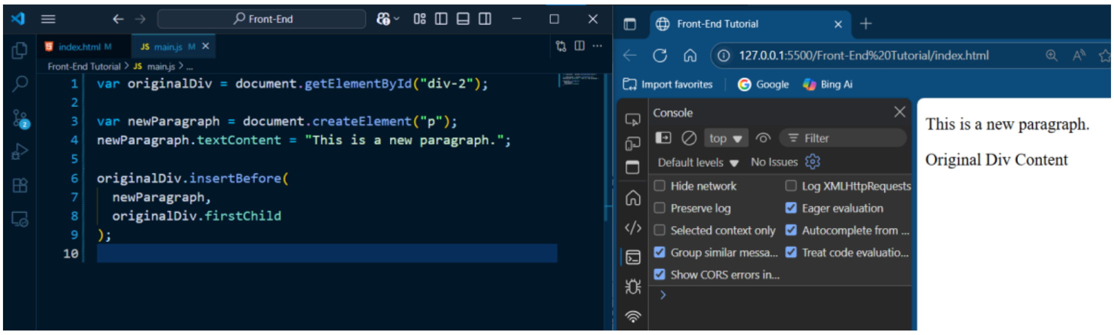
☐ Hide network ☐ Log XMLHttpRequests
☐ Preserve log ☑ Eager evaluation
☐ Selected context only ☑ Autocomplete from ...
☑ Group similar messa... ☑ Treat code evaluatio...
☑ Show CORS errors in...

First

Second

This is New Paragraph

26. Clone #div-2 , set clone id="div-2-clone", and insert it after original using insertAfter.



```
var originalDiv = document.getElementById("div-2");

var newParagraph = document.createElement("p");
newParagraph.textContent = "This is a new paragraph.";

originalDiv.insertBefore(
  newParagraph,
  originalDiv.firstChild
);
```

127.0.0.1:5500/Front-End%20Tutorial/index.html

Console

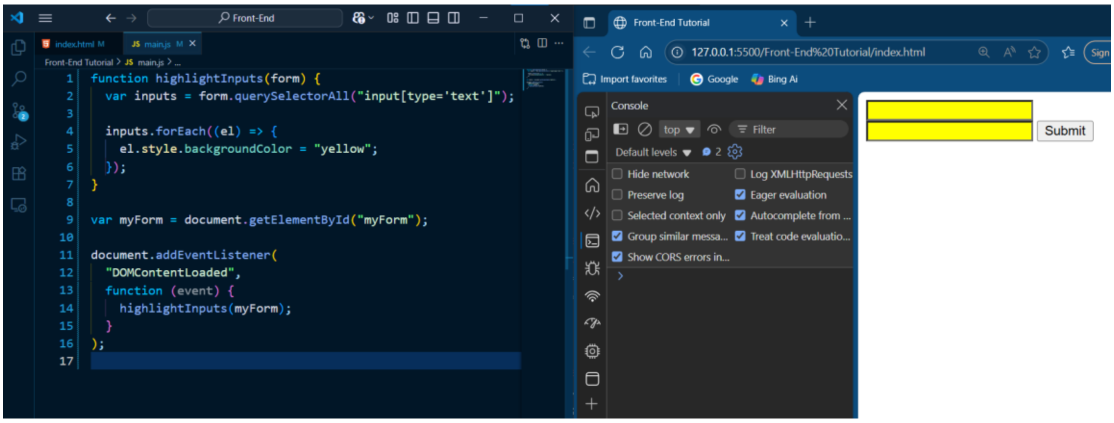
Default levels ▾ No Issues

☐ Hide network ☐ Log XMLHttpRequests
☐ Preserve log ☑ Eager evaluation
☐ Selected context only ☑ Autocomplete from ...
☑ Group similar messa... ☑ Treat code evaluatio...
☑ Show CORS errors in...

This is a new paragraph.

Original Div Content

27. Highlight all text & email inputs (green border) in a function highlightInputs(form) and call it on DOMContentLoaded.



```
function highlightInputs(form) {
  var inputs = form.querySelectorAll("input[type='text']");

  inputs.forEach((el) => {
    el.style.backgroundColor = "yellow";
  });
}

var myForm = document.getElementById("myForm");

document.addEventListener(
  "DOMContentLoaded",
  function (event) {
    highlightInputs(myForm);
  }
);
```

127.0.0.1:5500/Front-End%20Tutorial/index.html

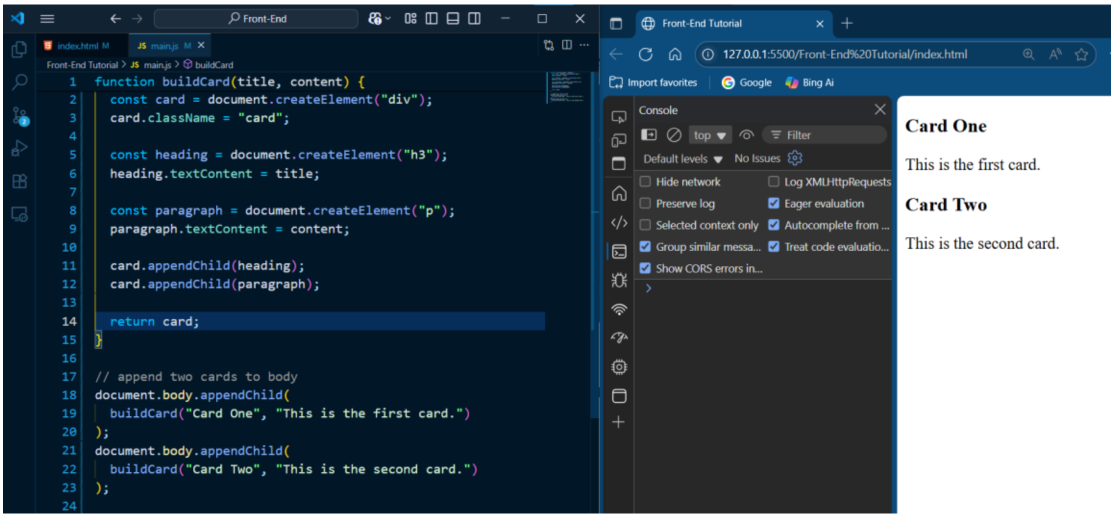
Console

Default levels ▾ 2

☐ Hide network ☐ Log XMLHttpRequests
☐ Preserve log ☑ Eager evaluation
☐ Selected context only ☑ Autocomplete from ...
☑ Group similar messa... ☑ Treat code evaluatio...
☑ Show CORS errors in...

Submit

28. Build function buildCard(title, content) returning <div class="card"> with an <h3> and <p>; append two cards to body.



```
function buildCard(title, content) {
  const card = document.createElement("div");
  card.className = "card";

  const heading = document.createElement("h3");
  heading.textContent = title;

  const paragraph = document.createElement("p");
  paragraph.textContent = content;

  card.appendChild(heading);
  card.appendChild(paragraph);

  return card;
}

// append two cards to body
document.body.appendChild(
  buildCard("Card One", "This is the first card.")
);
document.body.appendChild(
  buildCard("Card Two", "This is the second card.")
);
```

127.0.0.1:5500/Front-End%20Tutorial/index.html

Console

Default levels ▾ No Issues

☐ Hide network ☐ Log XMLHttpRequests
☐ Preserve log ☑ Eager evaluation
☐ Selected context only ☑ Autocomplete from ...
☑ Group similar messa... ☑ Treat code evaluatio...
☑ Show CORS errors in...

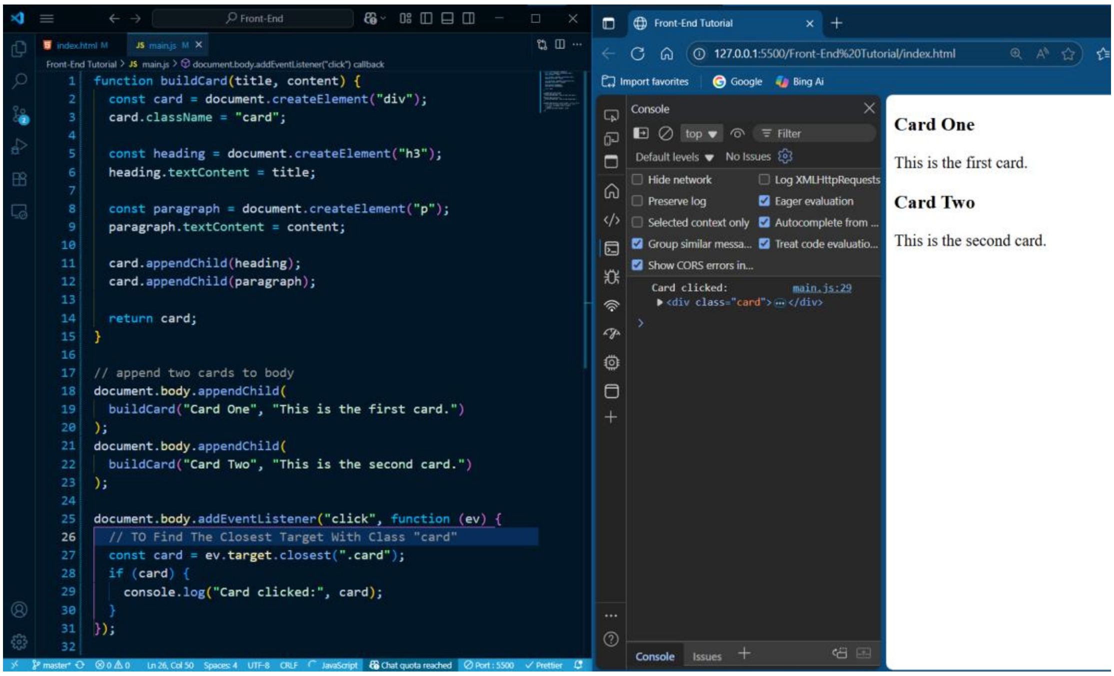
Card One

This is the first card.

Card Two

This is the second card.

29. Add delegated click listener on body logging when a .card is clicked.



```
function buildCard(title, content) {
  const card = document.createElement("div");
  card.className = "card";

  const heading = document.createElement("h3");
  heading.textContent = title;

  const paragraph = document.createElement("p");
  paragraph.textContent = content;

  card.appendChild(heading);
  card.appendChild(paragraph);

  return card;
}

// append two cards to body
document.body.appendChild(
  buildCard("Card One", "This is the first card.")
);
document.body.appendChild(
  buildCard("Card Two", "This is the second card.")
);

document.body.addEventListener("click", function (ev) {
  // TO Find The Closest Target With Class "card"
  const card = ev.target.closest(".card");
  if (card) {
    console.log("Card clicked:", card);
  }
});
```

```
Card clicked:                main.js:29
▶ <div class="card"> … </div>
```

Card One

This is the first card.

Card Two

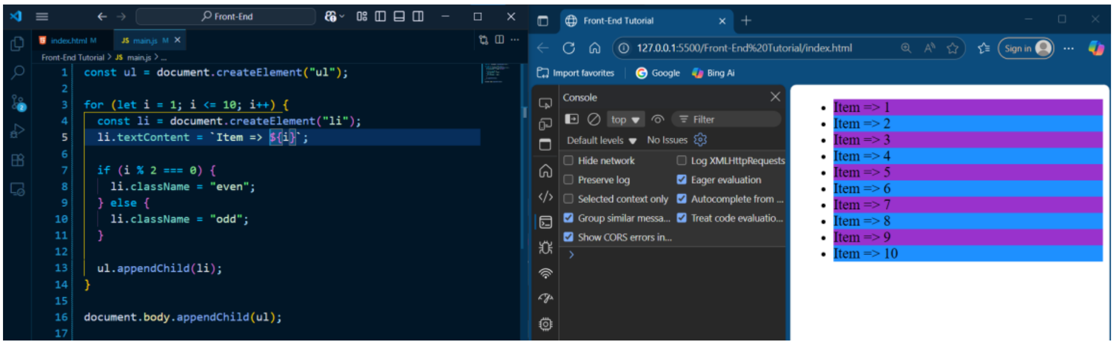
This is the second card.

30. Reflection (comment): Which two tasks were most challenging and why?

>> Task No.27: Searching For “DOMContentLoaded” Event

>> Task No.29: How To Get The Event Target Element

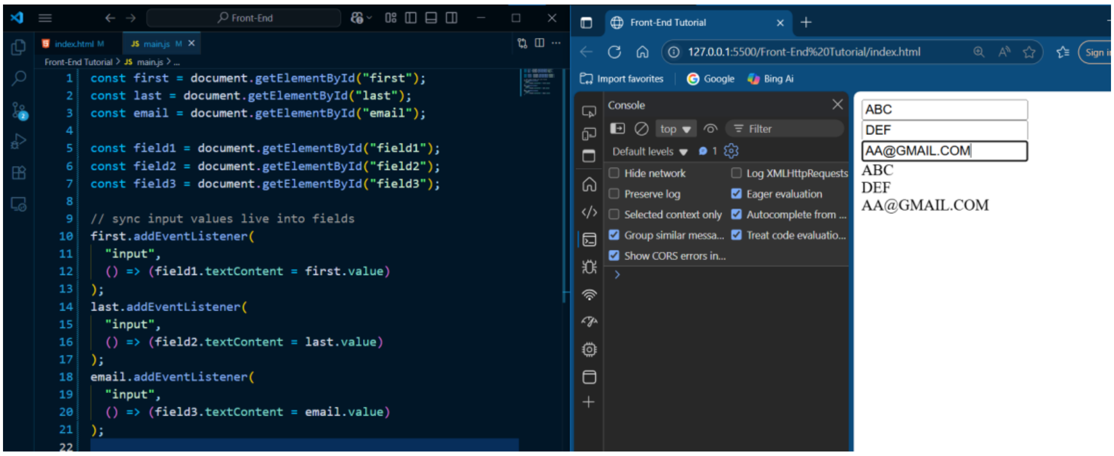
31. create and unordered list dynamically with at 10 items, the odd list items should have class "odd" and even items "even", -create the two classes in your CSS file-



```
const ul = document.createElement("ul");

for (let i = 1; i <= 10; i++) {
  const li = document.createElement("li");
  li.textContent = `Item => ${i}`;

  if (i % 2 === 0) {
    li.className = "even";
  } else {
    li.className = "odd";
  }

  ul.appendChild(li);
}

document.body.appendChild(ul);
```

- Item => 1
- Item => 2
- Item => 3
- Item => 4
- Item => 5
- Item => 6
- Item => 7
- Item => 8
- Item => 9
- Item => 10

32. Form Has 3 Inputs And Separated Div With 3 Fields => Inputs values in Form Appears in Div as child elements



```javascript
const first = document.getElementById("first");
const last = document.getElementById("last");
const email = document.getElementById("email");

const field1 = document.getElementById("field1");
const field2 = document.getElementById("field2");
const field3 = document.getElementById("field3");

// sync input values live into fields
first.addEventListener(
  "input",
  () => (field1.textContent = first.value)
);
last.addEventListener(
  "input",
  () => (field2.textContent = last.value)
);
email.addEventListener(
  "input",
  () => (field3.textContent = email.value)
);
```

127.0.0.1:5500/Front-End%20Tutorial/index.html

☐ Hide network ☐ Log XMLHttpRequests
☐ Preserve log ☑ Eager evaluation
☐ Selected context only ☑ Autocomplete from ...
☑ Group similar messa... ☑ Treat code evaluatio...
☑ Show CORS errors in...

ABC
DEF
AA@GMAIL.COM

ABC
DEF
AA@GMAIL.COM