TITLE

ONE SHOT TO JAVASCRIPT DOM MANIPULATION

COURSE INSTRUCTOR

ABDULLAH AL NOMAN PRINCE

SOFTWARE ENGINEER

SUBJECT MATTER EXPERT & SUPPORT INSTRUCTOR, OSTAD

TOPICS WILL BE COVERD

FUNDAMENTAL TOPICS OF JAVASCRIPT

BASICS OF DOM MANIPULATION

ONE PROJECT

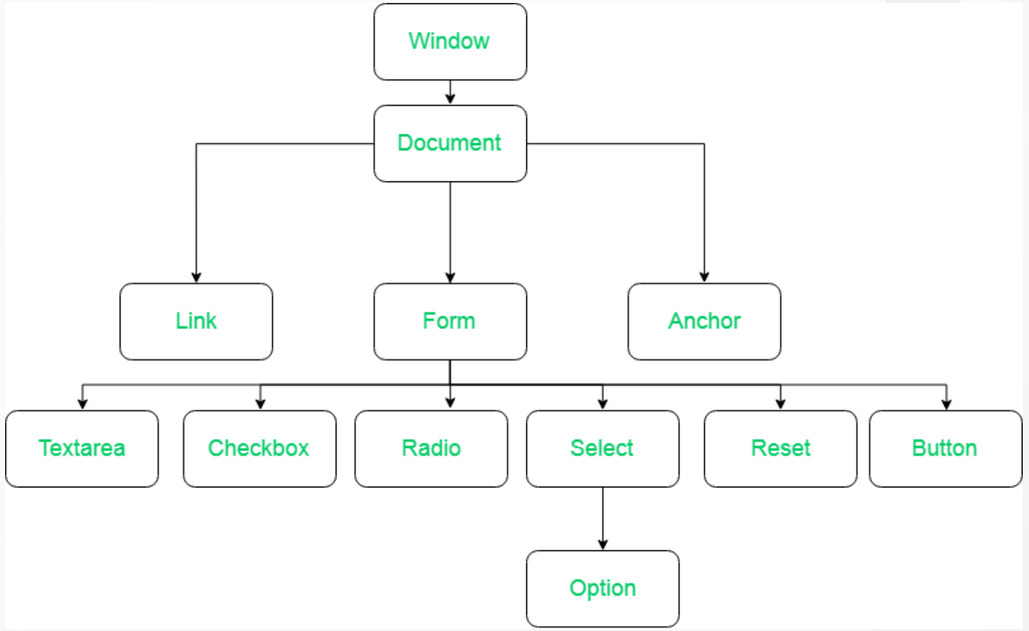
WHAT IS DOM?

DOM refers to Document Object Model

HTML page is the Document

HTML elements are the Object

These elements are represented by a tree model.



IMPORTANT DOM PROPERTIES AND METHODS

console.log(document)

console.dir(document)

console.log(document.title)

console.log(document.domain)

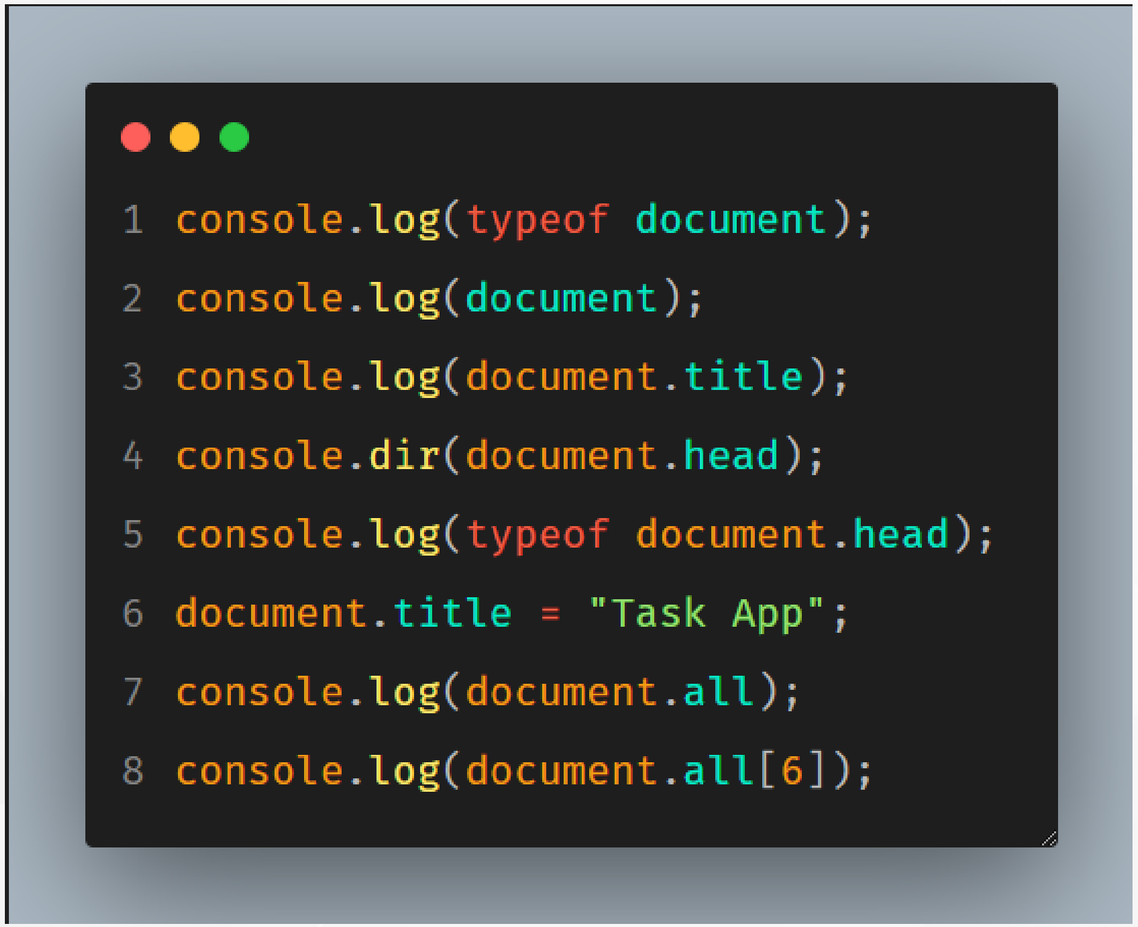
console.log(document.head)

console.log(document.all) returns HTML Collection

HTML Collecetion is a special type of object not array.

Selecting elements by index is not proper way.

IMPORTANT DOM PROPERTIES AND METHODS



```
1 console.log(typeof document);
2 console.log(document);
3 console.log(document.title);
4 console.dir(document.head);
5 console.log(typeof document.head);
6 document.title = "Task App";
7 console.log(document.all);
8 console.log(document.all[6]);
```

IMPORTANT DOM PROPERTIES AND METHODS

GET ELEMENT BY ID

```
const heading = document.getElementById('heading')
heading.innerText = "JavaScript")        Browser representation
heading.textContent = "JavaScript")      Return only Strings presents on code
heading.innerHTML = "JavaScript")        Returns everything inside an element

heading.style.color = "red")
heading.style.fontSize= "50px")
```

IMPORTANT DOM PROPERTIES AND METHODS

GET ELEMENT BY ID

```
1 console.log(document.getElementById("header"));
2 let header = document.getElementById("header");
3 header.innerText = "JavaScript";
4 header.textContent = "PHP";
5 header.innerHTML = "PHP";
6
7 header.style.color = "red";
8 header.style.fontSize = "40px";
```

IMPORTANT DOM PROPERTIES AND METHODS

GET ELEMENTS BY ClassName

const headings = document.getElementsByClassName('heading') HTML Collection

```
console.log(document.getElementsByClassName("item"));
let itemUl = document.getElementById("items");
let items = itemUl.getElementsByClassName("item");
console.log(items);

for (let i = 0; i < items.length; i++) {
  console.log(items[i]);
}

```

IMPORTANT DOM PROPERTIES AND METHODS

GET ELEMENTS BY TagName

const heading = document.getElementsByClassName('h1') HTML Collection

```
1 console.log(document.getElementsByTagName("h1"));
```

IMPORTANT DOM PROPERTIES AND METHODS

QuerySelector

It only selects the first matched element

```
const heading = document.querySelector('#heading')
const heading = document.querySelector('.headingTitle')
const heading = document.querySelector('h1')
```

Takes parameter like CSS selectors

```
const heading = document.querySelector('.item:last-child')
```

IMPORTANT DOM PROPERTIES AND METHODS

QuerySelector

```
1 console.log(document.querySelector(".item"));
2 console.log(document.querySelector(".item:last-child"));
3 let lastChild = document.querySelector(".item:last-child");
4 lastChild.style.color = "red";
5 console.log(document.querySelector(".item:nth-child(2)"));
6 console.log(document.querySelector(".item:nth-child(5)"));
```

IMPORTANT DOM PROPERTIES AND METHODS

QuerySelectorAll NodeList

It returns all matches element

```
const heading = document.querySelectorAll('.headingTitle')
```

IMPORTANT DOM PROPERTIES AND METHODS

QuerySelectorAll NodeList

It returns all matches element

```
const heading = document.querySelectorAll('.headingTitle')
```

IMPORTANT DOM PROPERTIES AND METHODS

QuerySelectorAll

NodeList

```
1  console.log(document.querySelectorAll(".item"));
2  console.log(document.querySelectorAll(".item:last-child"));
3  let lastChilds = document.querySelectorAll(".item:last-child");
4  for (let lastChild of lastChilds) {
5    lastChild.style.color = "blue";
6  }
7
8  // ul→li
9  let ul = document.querySelector("#items").querySelector(".item:last-child");
10 console.log(ul);
```

IMPORTANT DOM PROPERTIES AND METHODS

```
const parent = document.querySelector("#items");
const children = parent.children;
console.log(children);

//Descendant
const grandParent = document.querySelector(".todo-list");
const parent = grandParent.children;
const children = parent[1].children;
console.log(children);

//Accessing from GP to Child directly
const grandParent = document.querySelector(".todo-list");
const children = grandParent.querySelectorAll(".item");
console.log(children);

// Ancistor
const child = document.querySelector("#items");
const p = child.parentElement;
const p = child.closest(".todo-list");
console.log(p);
```

IMPORTANT DOM PROPERTIES AND METHODS

Creating an element

```
1 const divELement = document.createElement("div");
2 console.log(divELement);
3 divELement.className = "box";
4 divELement.setAttribute("id", "box");
5 console.log(divELement);
6 const container = document.querySelector(".todo-list");
7 console.log(container);
8 const h2 = document.querySelector("h2");
9 container.insertBefore(divELement, h2);
```

IMPORTANT DOM PROPERTIES AND METHODS

Creating an element

```
1  //Insert Last
2  const container = document.querySelector(".todo-list");
3  const divELement = document.createElement("div");
4  container.appendChild(divELement);
5  container.append("hello");
6  // Only takes html element
7  container.appendChild("hello");
8
9  container.append(divELement, "HI", "Bye");
10 //Only adds first element
11 container.appendChild(divELement, "HI", "Bye");
```

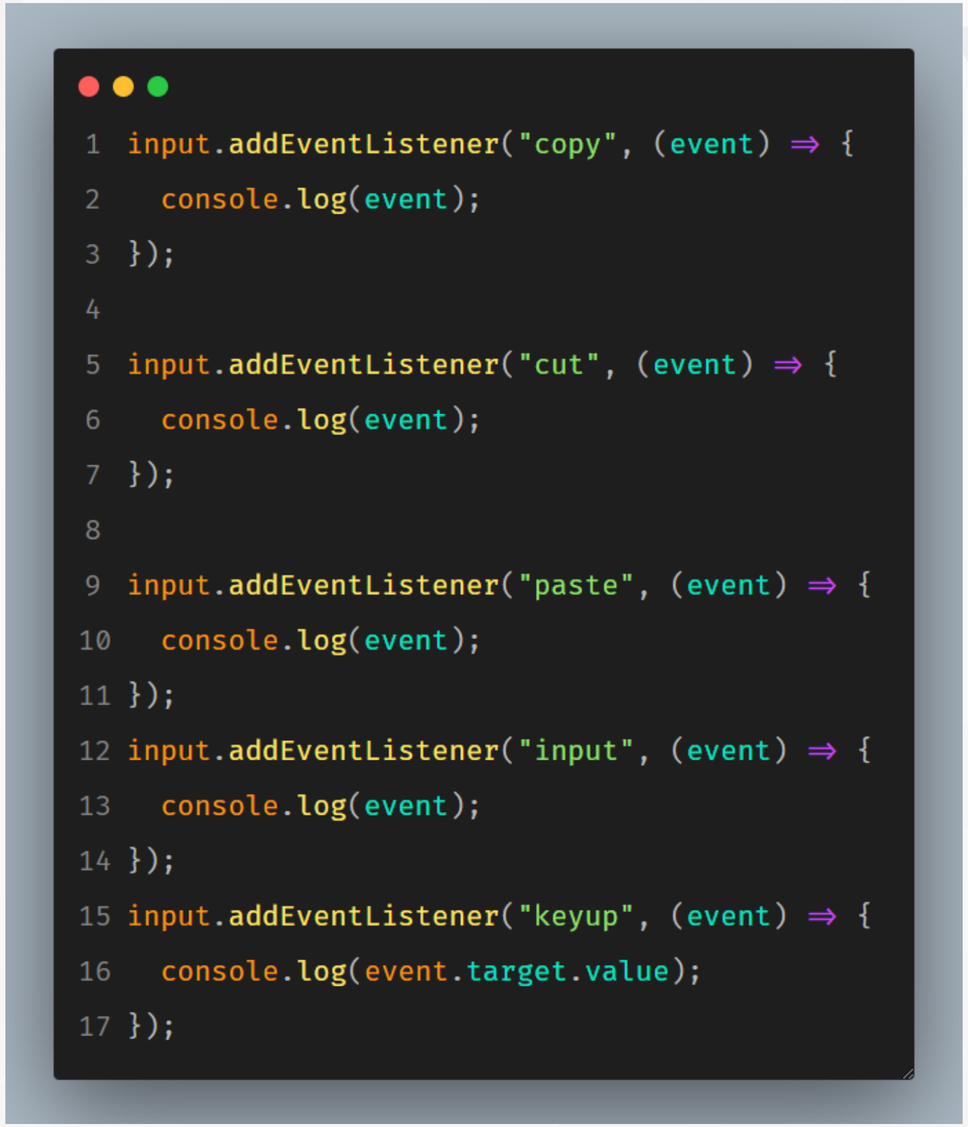
IMPORTANT DOM PROPERTIES AND METHODS

EventListeners

```
1  const input = document.getElementById("new-task");
2  console.log(input);
3  input.addEventListener("keyup", (event) => {
4    console.log(event);
5  });
6  input.addEventListener("keydown", (event) => {
7    console.log(event);
8  });
9
10 input.addEventListener("focus", (event) => {
11   console.log(event);
12 });
13
14 input.addEventListener("blur", (event) => {
15   console.log(event);
16 });
```

IMPORTANT DOM PROPERTIES AND METHODS

EventListeners



```
input.addEventListener("copy", (event) => {
  console.log(event);
});

input.addEventListener("cut", (event) => {
  console.log(event);
});

input.addEventListener("paste", (event) => {
  console.log(event);
});
input.addEventListener("input", (event) => {
  console.log(event);
});
input.addEventListener("keyup", (event) => {
  console.log(event.target.value);
});
```

IMPORTANT DOM PROPERTIES AND METHODS

EventListeners

```
// Form
document.querySelector("form").addEventListener("submit", (event) => {
  event.preventDefault();
  console.log(input.value);
});
```

THANK YOU