

✓ element.classList

Restituisce un oggetto che rappresenta le classi CSS dell'elemento, con metodi utili come

`.add()` `.remove()` `.toggle()` `.contains()`

```
const titolo = document.querySelector("h1");
titolo.classList.add("evidenziato");
titolo.classList.remove("nascosto");
titolo.classList.toggle("attivo"); // se presente la toglie, se assente la aggiunge
```

✓ element.className

Permette di leggere o impostare direttamente tutte le classi dell'elemento come stringa.

```
const paragrafo = document.querySelector("p");
console.log(paragrafo.className); // es. "rosso grande"
paragrafo.className = "blu corsivo"; // sovrascrive tutte le classi
```

✓ element.innerHTML

Serve per leggere o modificare il contenuto HTML interno di un elemento.

```
const contenitore = document.getElementById("box");
contenitore.innerHTML = "<strong>Contenuto aggiornato!</strong>";
```

✓ element.style

Permette di leggere o modificare direttamente le proprietà CSS inline.

```
const box = document.querySelector("#box");
box.style.backgroundColor = "yellow";
box.style.fontSize = "18px";
```

✅ **element.addEventListener()**

Aggancia una funzione (callback) a un evento come click, mouseover, keydown, ecc.

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

```
bottone.addEventListener("click", function() {  
    alert("Hai cliccato il bottone!");  
});
```

=====

✅ **element.value**

Si usa con elementi `<input>`, `<textarea>`, o `<select>` per leggere o impostarne il valore.

```
const inputNome = document.getElementById("nomeUtente");  
console.log(inputNome.value); // stampa il valore inserito
```

```
inputNome.value = "Mario Rossi"; // imposta un valore predefinito
```