

# ACTIVITATS BOM, DOM i EVENTS

## 1. Selecciona l'element button de la pàgina.

En aquest escenari, el codi existent afegeix un eventListener per a un event de click en una variable buttonElem. Completa el codi JS necessari per assignar l'element botó a la variable buttonElem. Crea un .html i un .js per resoldre i provar aquesta activitat.

HTML:

...

```
<button type="button" id="button">OFF</button>
```

...

JS:

```
const buttonElem =  
  
buttonElem.addEventListener('click', () => {  
  const oldText = buttonElem.innerText;  
  return buttonElem.innerText = oldText === "ON" ? "OFF" : "ON";  
});
```

## 2. Consulta d'elements descendents

El codi existent espera les variables 'buttonElem' i 'inputElem' per representar el botó i elements d'entrada. Assigna els elements respectius a les variables.

En aquest cas, els dos elements no tenen identificadors únics - com per exemple un id. En comptes d'això són descendents directes d'un element div amb id wrapper. Crea un .html i un .js per resoldre i provar aquesta activitat.

HTML:

...

```
<div id="wrapper">  
  <input type="text" value="OFF" readonly/>  
  <button type="button">Click Me</button>  
</div>
```

...

JS:

```
const buttonElem =  
const inputElem =  
buttonElem.addEventListener('click', () => {  
  const oldText = inputElem.value;  
  return inputElem.value = oldText === "ON" ? "OFF" : "ON";  
});
```

### 3. Selecció múltiple d'elements

Assigna els elements de la llista a la variable 'listItems' per utilitzar un mètode de selector apropiat.

Crea un .html i un .js per resoldre i provar aquesta activitat, verifica que tots els elements mostren el valor 'ON'.

HTML:

```
...  
  
<ul id="list">  
<li>OFF</li>  
<li>OFF</li>  
<li>OFF</li>  
<li>OFF</li>  
<li>OFF</li>  
<li>OFF</li>  
<li>OFF</li>  
</ul>  
  
...
```

JS:

```
const listItems =  
  
const handleHover = (event) => {  
  return event.target.innerText = 'ON';  
};  
if(listItems.length > 1) {  
  listItems.forEach(item => item.addEventListener('mouseover', handleHover));  
}
```

### 4. Elimina un element del DOM

En l'HTML s'ha definit un cercle verd que queda tapat per el cercle roig.

Completa la funció removeRedCircle per treure el cercle amb id red des del DOM.

Crea un .html i un .js per resoldre i provar aquesta activitat.

HTML:

```
...  
  
<div id="green"/>  
<div id="red"/>  
<button type="button" id="button">Click Me</button>  
  
...
```

JS:

```
const button = document.querySelector('#button');  
  
const removeRedCircle = () => {  
  
};  
  
button.addEventListener('click', removeRedCircle);
```

## 5. Obtenció del nom complet amb inputs

Completa el codi JavaScript per interactuar amb els elements d'HTML definits.

Cerquem el nom complet d'un usuari. Amb el click del botó, combina els noms dels primers dos camps d'entrada. Insereix el nom complet en el tercer camp d'entrada.

Crea un .html i un .js per resoldre i provar aquesta activitat.

HTML:

```
...  
<input type="text" id="firstName" placeholder="Max" value="Max"/>  
<input type="text" id="lastName" placeholder="Musterman" value="Musterman"/>  
<input type="text" id="fullName" placeholder="full name" readonly/>  
<button type="button" id="button">Verify Code</button>  
...
```

JS:

```
const button = document.getElementById('button');  
button.addEventListener('click' , () => {  
  // Completa el codi  
});
```