

Examen Javascript

IES Luis Simarro, 2 DAW, Curs 2020-2021

1 (1 punt) Observa aquest codi:

```
function makeWorker() {  
  let name = "Pedro";  
  return function() {  
    console.log(name);  
  };  
}  
let name = "Juan";  
let work = makeWorker();  
work();
```

Quin nom ix per la consola? Explica perquè.

Pedro.

La variable name és declarada dins i fora de la funció, però la funció que retorna accedirà a la de dins perquè és una closure.

2 (1 punt) Observa aquest codi:

```
let phrase = "Hola";  
if (true) {  
  let user = "Juan";  
  function sayHi() {  
    alert(`${phrase}, ${user}`);  
  }  
}  
sayHi();
```

Què va a passar? Explica perquè.

Encara que user siga declarat dins del if, és accessible a la funció perquè es genera una closure. Si vulguem accedir a user sense sayhi() falla. La declaració de funcions no té àmbit de bloc, per tant, és accesible fora del if.

3 (4 punts) Amb aquest codi:

```
function teclat(){  
  this.tecles = `  
1 2 3 4 5 6 7 8 9 0 - =  
Q W E R T Y U I O P [ ] \  
A S D F G H J K L ; '  
Z X C V B N M , . / `;  
`  
}
```

- a) Utilitza'l com una funció constructora d'objectes teclat i fes que tinga una funció en el prototype per a dibuixar eixe teclat dins d'un div amb id="teclat". No cal que tinga molt d'estil.

```
function teclat() {
    this.tecles = `1 2 3 4 5 6 7 8 9 0 - =
Q W E R T Y U I O P [ ] \
A S D F G H J K L ; '
Z X C V B N M , . / `;
}

teclat.prototype.dibuixar = function() {
    let tecles = this.tecles.split(' ');
    let contenidor = document.querySelector('#teclat');
    let fila = document.createElement('div');
    for (let tecla of tecles) {
        let t = document.createElement('div');
        t.style.display='inline-block';
        let contingut = `<span style="display:block; width:50px;
height:50px; border: 1px solid #000;">${tecla}</span>`;
        t.addEventListener('click', ()=> { console.log(tecla)});
        t.innerHTML = contingut;
        fila.appendChild(t);
        if (['=', '\\', '"'].indexOf(tecla) > -1) {
            contenidor.appendChild(fila);
            fila = document.createElement('div');
        }
    }
};

(() =>{
    "use strict"; // Prova a descomentar
    document.addEventListener("DOMContentLoaded", function () {
        let t = new teclat();
        t.dibuixar();
    });
})();
```

- b) Cada vegada que es presiona una tecla, ix per la consola la tecla pulsada.
c) Transforma, en un altre codi, aquesta funció constructora en una class.

```

class teclat {
  constructor() {
    this.tecles = `1 2 3 4 5 6 7 8 9 0 - =
Q W E R T Y U I O P [ ] \
A S D F G H J K L ; '
Z X C V B N M , . /`;
  }

  dibujar() {
    let tecles = this.tecles.split(' ');
    let contenedor = document.querySelector('#teclat');
    let fila = document.createElement('div');
    for (let tecla of tecles) {
      let t = document.createElement('div');
      t.style.display = 'inline-block';
      let contingut = `${tecla}</span>`;
      t.addEventListener('click', () => { console.log(tecla); });
      t.innerHTML = contingut;
      fila.appendChild(t);
      if (['=', '\\', '"'].indexOf(tecla) > -1) {
        contenedor.appendChild(fila);
        fila = document.createElement('div');
      }
    }
  }
}

(() => {
  "use strict"; // Prova a descomentar
  document.addEventListener("DOMContentLoaded", function () {
    let t = new teclat();
    t.dibujar();
  });
})();

```

4 (2 punts) Observa aquest codi:

```
((() => {
  "use strict";
  class Sensor {
    constructor() {
      this.activo = false;
      this.element = document.createElement('div');
      this.element.innerHTML = `<span>${this.activo}</span>`;
      this.element.addEventListener('click', function () {
        this.activo = !this.activo;
        this.element.innerHTML = `<span>${this.activo}</span>`;
      })
    }
  }

  document.addEventListener("DOMContentLoaded", function () {
    let sensor = new Sensor();
    document.querySelector('#sensores').appendChild(sensor.element);
  });
})();
```

No pot funcionar. Explica perquè i troba dos solucions.

Dins de la funció callback del click, `this` és l'element `span` i no el sensor. Es pot solucionar amb una funció fletxa en compte de la funció actual o amb `that`. També es pot afegir el sensor com a atribut de l'element.

5 (2 punts) Tenim aquest array:

```
a = [1,2,3,4,5,6,7,8,9,10,11,12,13,14,15];
```

Amb les funcions `map()`, `filter()` i `reduce()` has d'aconseguir:

- Treure un array amb cada número multiplicat per 2.**
- Treure un array dels números que són divisibles entre 3.**
- Treure la suma dels números divisibles entre 3.**
- Treure un array amb cada número multiplicat per la suma dels números imparells.**

```
a = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15];

let per2 = a.map((n) => n * 2);
console.log(per2);

let divisibles = a.filter((n) => n % 3 == 0)
console.log(divisibles);
```

```
let sumaDivisibles = a.filter((n) => n % 3 == 0).reduce((p, n) => p +  
n)  
console.log(sumaDivisibles);  
  
let sumaImparells = a.map((x) => (  
  x * a.filter((n) => n % 2 == 1).reduce((p, n) => p + n)  
  )  
)  
console.log(sumaImparells);
```