

Fonaments de la programació

Programació Orientada a Objectes (2)
Classes. El constructor

`{.js}`
JavaScript



UNIÓ EUROPEA
Fons Europeu de
Desenvolupament Regional

El constructor d'una classe

El constructor d'una classe és un mètode especial que s'encarrega de rebre paràmetres personalitzats quan estem utilitzant l'operador **new**.

1. Ha de tenir el nom exacte "**constructor**"
2. S'executa automàticament quan es crea un objecte nou
3. S'utilitza per inicialitzar les propietats de l'objecte
4. Si no definiu un mètode constructor, JavaScript afegirà un mètode constructor buit.

```
class InstrumentCorda {  
  // métodos de clase  
  constructor(tipus, marca) {  
    this.tipus = tipus;  
    this.marca = marca;  
  }  
}
```



```
let guitar = new InstrumentCorda("Guitar", "Yamaha")
```

La paraula clau **this**

La paraula clau **this** fa referència directa als atributs de l'objecte.
Mirem aquest exemple:

```
class InstrumentCorda {  
  // métodos de clase  
  constructor(tipus, marca) {  
    this.tipus = tipus;  
    this.marca = marca;  
  }  
  tocarNota(nota) {  
    return this.tipus " + està tocant la nota " + nota;  
  }  
  tocarAcord(acord) {  
    return this.tipus " + està tocant l'acord " + acord;  
  }  
}
```

Amb la paraula **this**,
diferenciem
l'**atribut** de l'objecte
del **paràmetre**
del constructor.

Getters i Setters

Els mètodes **Getters** i **Setters** són mètodes d'accés a les propietats d'un objecte.

Els Mètodes **GETTERS**

Serveixen per recuperar el valor d'un atribut

```
getMarca() {  
    return this.marca;  
}
```

Els Mètodes **SETTERS**

Serveixen per donar valor a un atribut

```
setMarca("marca") {  
    this.marca = marca;  
}
```

```
let guitar = new InstrumentCorda("Guitar", "Yamaha");  
alert(guitar.getMarca());
```

```
guitar.setMarca("Fender");  
alert(guitar.getMarca());
```

Esta página dice

Yamaha

Aceptar

Esta página dice

Fender

Aceptar

El mètode toString()

El mètode `toString` és un mètode que s'hereta de la classe `Object`. Si aquest mètode no se sobreescriu i personalitza, retorna el tipus d'objecte. En sobreescriure el mètode `toString`, podem obtenir ràpidament els valors dels atributs d'un objecte, només referenciant-lo.

El codi següent crea una funció anomenada `instrumentCordaToString` per sobreescriure el mètode predeterminat `toString`. Aquesta funció genera una cadena que conté el tipus i la marca de l'objecte, en la forma " propietat = valor;".

```
InstrumentCorda.prototype.toString = function instrumentCordaToString() {  
    let retorno = "Instrument: " + this.tipus + ", marca " + this.marca;  
    return retorno;  
}
```

Ara podem imprimir els valors d'un objecte simplement fent-hi referència:

```
let guitar = new InstrumentCorda("Guitar", "Tatay");  
alert(guitar);
```

Esta página dice
Instrument: Guitar, marca Tatay

Aceptar

Codi complet

```
<html>
  <body>
    <h2>JavaScript POO2</h2>
    <p>Utilitzant la classe InstrumentCorda</p>
    <script>
      class InstrumentCorda {
        constructor(tipus, marca) {
          this.tipus = tipus;
          this.marca = marca;
        }
        getMarca() {
          return this.marca;
        }

        setMarca(valor) {
          this.marca = valor;
        }
      }

      InstrumentCorda.prototype.toString = function instrumentCordaToString() {
        let retorno = "Instrument: " + this.tipus + ", marca " + this.marca;
        return retorno;
      }

      let guitar = new InstrumentCorda("Guitar", "Yamaha");
      alert(guitar);
    </script>
  </body>
</html>
```