

Archivos

```
> node_modules
  app.js
  automovil.js
  camion.js
  moto.js
  package-lock.json
  package.json
  vehiculo.js
```

app.js

```
1  const express=require('express');
2  const app=express();
3  const puerto=3000;
4
5  const Vehiculo=require('./vehiculo');
6  const Moto=require('./moto');
7  const Automovil=require('./automovil');
8  const Camion=require('./camion');
9
10 app.get('/vehiculo/:marca/:modelo/:capacidad',(req,res)=>{
11   const marca=req.params.marca;
12   const modelo=req.params.modelo;
13   const capacidad=req.params.capacidad;
14   const vehiculo1=new Vehiculo(marca,modelo,capacidad);
15   res.send(
16     {
17       marca: vehiculo1.marca,
18       modelo: vehiculo1.modelo,
19       capacidad: vehiculo1.capacidad,
20     }
21   );
22   console.log(vehiculo1.marca);
23   console.log(vehiculo1.modelo);
24   console.log(vehiculo1.capacidad);
25 });
26 app.get('/vehiculo/moto/:marca/:modelo/:capacidad/:potencia',(req,res)=>{
27   const marca=req.params.marca;
28   const modelo=req.params.modelo;
29   const capacidad=req.params.capacidad;
30   const potencia=req.params.potencia;
31   const moto1=new Moto(marca,modelo,capacidad,potencia);
32   res.send(moto1.registrar());
33 });
34 app.get('/vehiculo/automovil/:marca/:modelo/:capacidad/:tipo',(req,res)=>{
35   const marca=req.params.marca;
36   const modelo=req.params.modelo;
37   const capacidad=req.params.capacidad;
38   const tipo=req.params.tipo;
39   const automovil1=new Automovil(marca,modelo,capacidad,tipo);
40   res.send(automovil1.registrar());
41 });
42 app.get('/vehiculo/camion/:marca/:modelo/:capacidad/:carga',(req,res)=>{
43   const marca=req.params.marca;
44   const modelo=req.params.modelo;
45   const capacidad=req.params.capacidad;
46   const carga=req.params.carga;
47   const camion1=new Camion(marca,modelo,capacidad,carga);
48   res.send({marca:camion1.marca});
49 });
50
51 app.listen(puerto,()=>{
52   console.log(`Servidor corriendo en http://localhost:${puerto}`);
53 });
```

vehiculo.js

```
1 class Vehiculo{
2   #marca
3   #modelo
4   #capacidad
5   constructor(marca,modelo,capacidad){
6     this.#marca=marca;
7     this.#modelo=modelo;
8     this.#capacidad=capacidad;
9   }
10  registrar(){
11    | return 'Registrar: Error de metodo, Este metodo cumple con poliforfismo, por lo tanto lo implementan las clases hijas';
12  }
13  consultar(){
14    | return 'Consultar: Error de metodo, Este metodo cumple con poliforfismo, por lo tanto lo implementan las clases hijas';
15  }
16  administrar(){
17    | return 'Administrar: Error de metodo, Este metodo cumple con poliforfismo, por lo tanto lo implementan las clases hijas';
18  }
19  get marca(){
20    | return this.#marca;
21  }
22  set marca(nuevaMarca){
23    | this.#marca=nuevaMarca;
24  }
25  get modelo(){
26    | return this.#modelo;
27  }
28  set modelo(nuevoModelo){
29    | this.#marca=nuevoModelo;
30  }
31  get capacidad(){
32    | return this.#capacidad;
33  }
34  set capacidad(nuevaCapacidad){
35    | this.#marca=nuevaCapacidad;
36  }
37 }
38 module.exports=Vehiculo;
```

moto.js

```
1 const Vehiculo=require("../vehiculo");
2
3 class Moto extends Vehiculo{
4   constructor(marca,modelo,capacidad,potencia){
5     | super(marca,modelo,capacidad);
6     | this.potencia=potencia;
7   }
8   registrar(){
9     | return 'La moto modelo ${this.modelo} y potencia ${this.potencia} ya ha sido registrada';
10  }
11  consultar(){
12    | return 'La moto de marca ${this.marca} y modelo ${this.modelo} se encuentra en mantenimiento';
13  }
14  administrar(){
15    | return 'La moto de marca ${this.marca} y capacidad ${this.capacidad} pasajeros, tiene una ruta ya planificada';
16  }
17 }
18 module.exports=Moto;
```

automovil.js

```
1 const Vehiculo=require("../vehiculo");
2
3 class Automovil extends Vehiculo{
4   constructor(marca,modelo,capacidad,tipo){
5     | super(marca,modelo,capacidad);
6     | this.tipo=tipo;
7   }
8   registrar(){
9     | return 'El automovil modelo ${this.modelo} y tipo ${this.tipo} ya ha sido registrado';
10  }
11  consultar(){
12    | return 'El automovil de marca ${this.marca} y modelo ${this.modelo} se encuentra activo';
13  }
14  administrar(){
15    | return 'El automovil de marca ${this.marca} y capacidad ${this.capacidad} pasajeros, tiene una ruta ya planificada';
16  }
17 }
18 module.exports=Automovil;
```

camion.js

```
1  const Vehiculo=require('./vehiculo');
2
3  class Camion extends Vehiculo{
4      constructor(marca,modelo,capacidad,carga){
5          |   super(marca,modelo,capacidad);
6          |   this.carga=carga;
7          |
8          registrar(){
9          |   return `El camión modelo ${this.modelo} y capacidad ${this.capacidad} aun no ha sido registrado`;
10         |
11         consultar(){
12         |   return `El camión de marca ${this.marca} y modelo ${this.modelo} se encuentra inactivo`;
13         |
14         administrar(){
15         |   return `El camión de marca ${this.marca} y carga ${this.carga} toneladas, tiene una ruta ya planificada`;
16         |   }
17         }
18     module.exports=Camion;
```