

Tarea 7 De Javascript

- Problema 1:



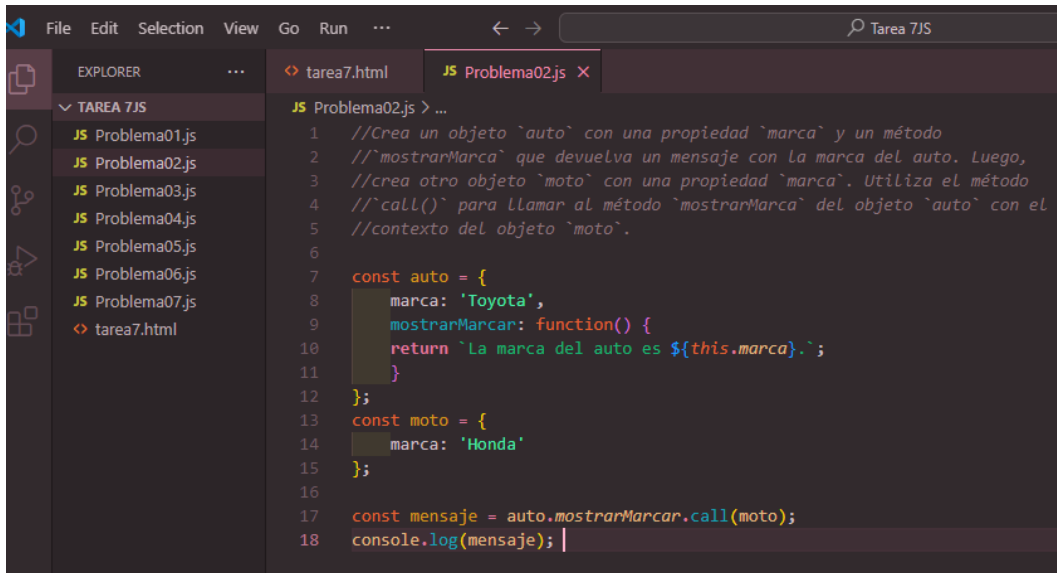
```
File Edit Selection View Go Run ...
< tarea7.html x JS Problema01.js x
TAREA 7JS
JS Problema01.js
JS Problema02.js
JS Problema03.js
JS Problema04.js
JS Problema05.js
JS Problema06.js
JS Problema07.js
tarea7.html

JS Problema01.js > ...
1 //Escribe una función llamada `saludar` que tome un parámetro
2 //`nombre` y devuelva un saludo personalizado. Luego, crea un objeto
3 //`persona` con una propiedad `nombre`. Utiliza el método `call()` para
4 //llamar a la función `saludar` con el contexto del objeto `persona`.
5 //Si lo deseas apoyate en la siguiente estructura://
6
7 function saludar(nombre) {
8   return `Hola, ${this.nombre}!`;
9 }
10
11 const persona = {
12   nombre: 'Juan'
13 };
14
15 const saludo = saludar.call(persona, persona.nombre);
16 console.log(saludo);
```

La función saludar permite saludar a la propiedad "nombre" y después el método "return" permite que el saludo sea con la propiedad nombre.

Por eso es que se debe crear un objeto, en este caso se nombra el objeto "persona" y su propiedad es "Juan", después de esto se hace una variable que sea "saludo" y se usa el método "call()" para llamar a las propiedades que se van a usar y aparecer en consola.

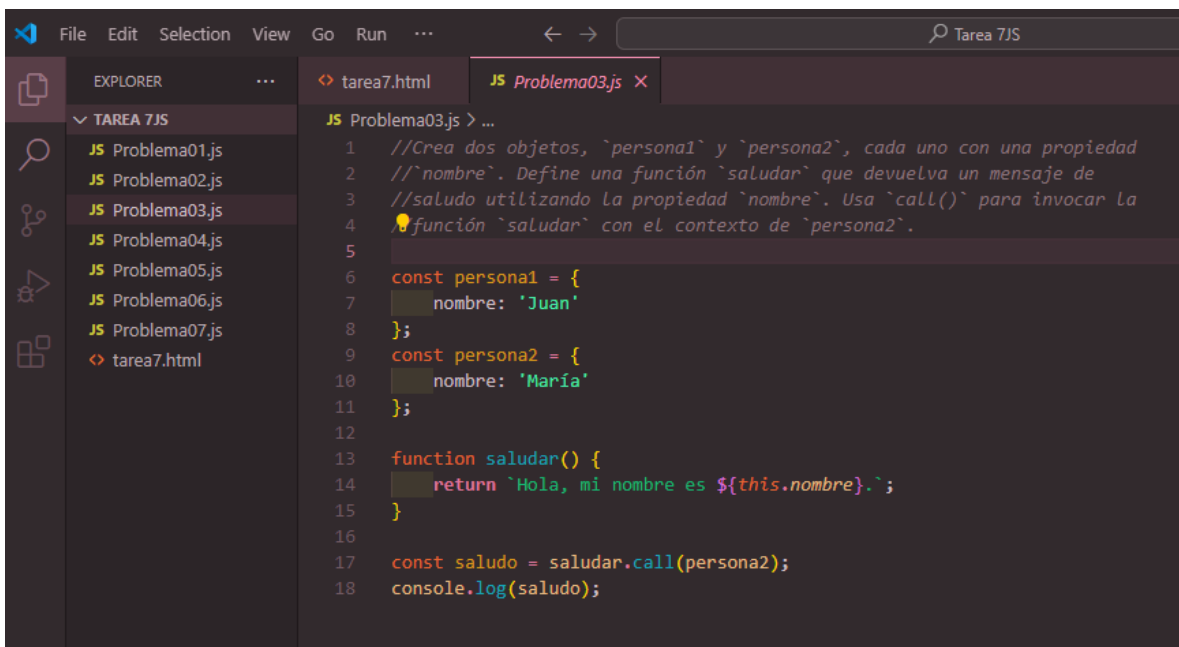
- Problema 2:



```
1 //Crea un objeto `auto` con una propiedad `marca` y un método
2 //`mostrarMarca` que devuelva un mensaje con la marca del auto. Luego,
3 //crea otro objeto `moto` con una propiedad `marca`. Utiliza el método
4 //`call()` para llamar al método `mostrarMarca` del objeto `auto` con el
5 //contexto del objeto `moto`.
6
7 const auto = {
8   marca: 'Toyota',
9   mostrarMarca: function() {
10     return `La marca del auto es ${this.marca}.`;
11   }
12 };
13 const moto = {
14   marca: 'Honda'
15 };
16
17 const mensaje = auto.mostrarMarca.call(moto);
18 console.log(mensaje);
```

CREAMOS LOS OBJETOS "AUTO" Y "MOTO" CON LA PROPIEDAD "MARCA", APARTE LA FUNCIÓN "MOSTRARMARCA" PARA PODER ENSEÑAR EN CONSOLA EL MENSAJE QUE QUEREMOS DAR LLAMANDO A LA FUNCIÓN Y A LA PROPIEDAD DEL OBJETO (EN ESTE CASO ES MARCA)

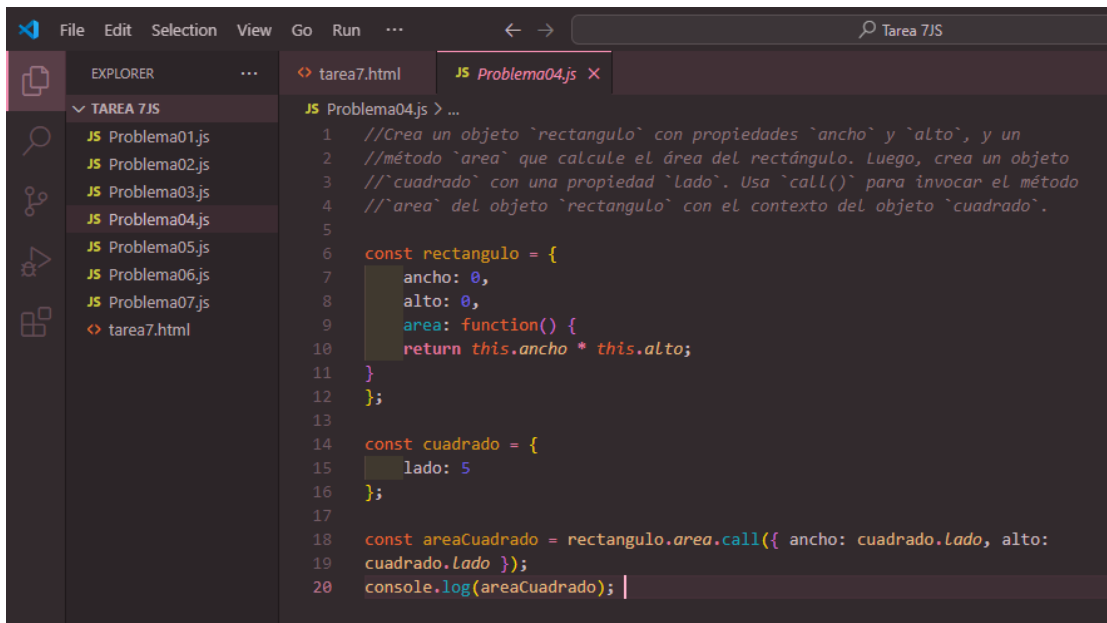
- PROBLEMA 3:



```
1 //Crea dos objetos, `persona1` y `persona2`, cada uno con una propiedad
2 //`nombre`. Define una función `saludar` que devuelva un mensaje de
3 //saludo utilizando la propiedad `nombre`. Usa `call()` para invocar la
4 //función `saludar` con el contexto de `persona2`.
5
6 const persona1 = {
7   nombre: 'Juan'
8 };
9 const persona2 = {
10   nombre: 'María'
11 };
12
13 function saludar() {
14   return `Hola, mi nombre es ${this.nombre}.`;
15 }
16
17 const saludo = saludar.call(persona2);
18 console.log(saludo);
```

CREAMOS DOS VARIABLES CON OBJETOS QUE TIENEN LA PROPIEDAD DE "NOMBRE" CADA UNO DE ELLOS, DESPUÉS LA FUNCIÓN DE "SALUDAR" JUNTO CON EL MENSAJE DEL OBJETO QUE SE ESTÁ SELECCIONANDO PARA QUE APAREZCA EN CONSOLA DEL NAVEGADOR AUTOMÁTICAMENTE LA VARIABLE "SALUDO".

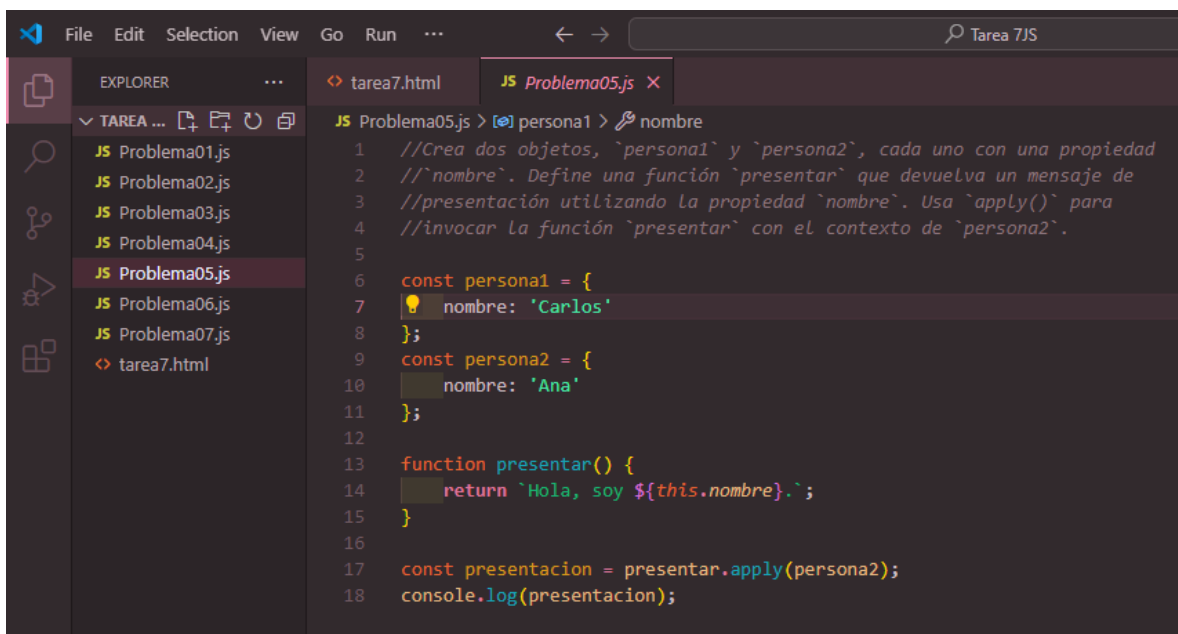
- Problema 4:



```
1 //Crea un objeto `rectangulo` con propiedades `ancho` y `alto`, y un
2 //método `area` que calcule el área del rectángulo. Luego, crea un objeto
3 //`cuadrado` con una propiedad `lado`. Usa `call()` para invocar el método
4 //`area` del objeto `rectangulo` con el contexto del objeto `cuadrado`.
5
6 const rectangulo = {
7   ancho: 0,
8   alto: 0,
9   area: function() {
10     return this.ancho * this.alto;
11   }
12 };
13
14 const cuadrado = {
15   lado: 5
16 };
17
18 const areaCuadrado = rectangulo.area.call({ ancho: cuadrado.lado, alto:
19   cuadrado.lado });
20 console.log(areaCuadrado);
```

EN ESTE CÓDIGO HAY DOS VARIABLES QUE SON "RECTÁNGULO" Y "CUADRADO" QUE TIENEN DIFERENTES PROPIEDADES (ANCHO, ALTO Y LADO), DESPUÉS ESTÁ LA FUNCIÓN QUE ES MEDIR EL ÁREA DEL CUADRADO Y SE USA LA FUNCIÓN CALL PARA PODER LLAMAR A LA PROPIEDAD Y AL OBJETO O LA VARIABLE.

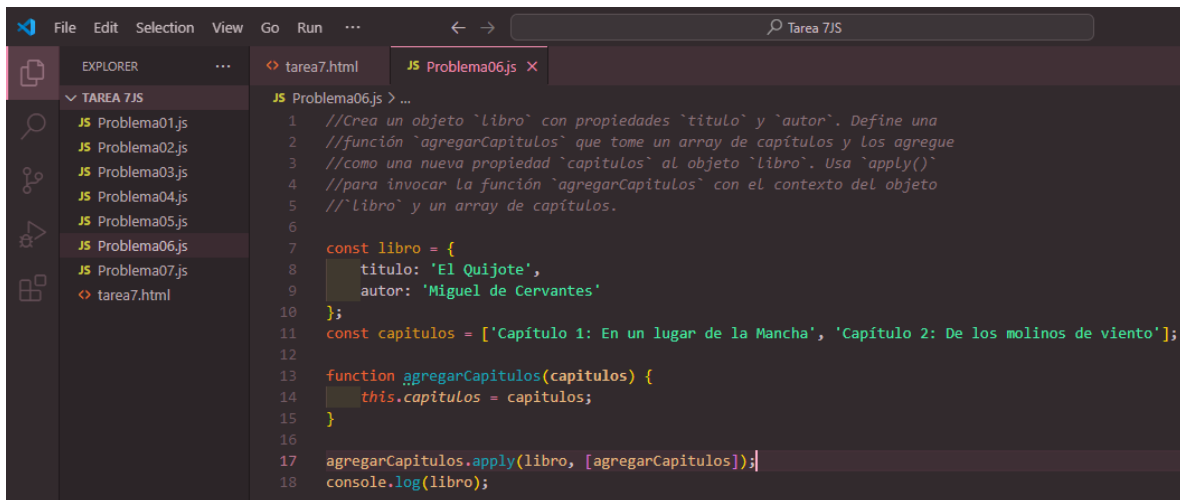
- Problema 5:



```
1 //Crea dos objetos, `persona1` y `persona2`, cada uno con una propiedad
2 //`nombre`. Define una función `presentar` que devuelva un mensaje de
3 //presentación utilizando la propiedad `nombre`. Usa `apply()` para
4 //invocar la función `presentar` con el contexto de `persona2`.
5
6 const persona1 = {
7   nombre: 'Carlos'
8 };
9 const persona2 = {
10   nombre: 'Ana'
11 };
12
13 function presentar() {
14   return `Hola, soy ${this.nombre}.`;
15 }
16
17 const presentacion = presentar.apply(persona2);
18 console.log(presentacion);
```

CREAMOS DOS OBJETOS QUE TIENEN LA MISMA PROPIEDAD "NOMBRE" Y LA FUNCIÓN QUE ES PRESENTAR CADA UNO DE ESOS OBJETOS CON LA FUNCIÓN "APPLY()" Y PARA ESO HAY UNA VARIACIÓN QUE ES LA PRESENTACIÓN; LA VARIABLE DEL MENSAJE QUE VA A APARECER EN CONSOLA.

- PROBLEMA 6:



```
1 //Crea un objeto `libro` con propiedades `titulo` y `autor`. Define una
2 //función `agregarCapitulos` que tome un array de capitulos y los agregue
3 //como una nueva propiedad `capitulos` al objeto `libro`. Usa `apply()`
4 //para invocar la función `agregarCapitulos` con el contexto del objeto
5 //`libro` y un array de capitulos.
6
7 const libro = {
8   titulo: 'El Quijote',
9   autor: 'Miguel de Cervantes'
10 };
11 const capitulos = ['Capítulo 1: En un lugar de la Mancha', 'Capítulo 2: De los molinos de viento'];
12
13 function agregarCapitulos(capitulos) {
14   this.capitulos = capitulos;
15 }
16
17 agregarCapitulos.apply(libro, [agregarCapitulos]);
18 console.log(libro);
```

PARA EL PROBLEMA 6 DE ESTE EJERCICIO CREAMOS UN OBJETO LIBRO QUE TIENE LAS PROPIEDAD "TÍTULO" Y "AUTOR", CREAMOS TAMBIÉN EL ARRAY DE CAPÍTULOS Y UNA FUNCIÓN QUE AGREGUE CAPÍTULOS LLAMANDO A LAS VARIABLES DECLARADAS Y AL FINAL APARECE EL NOMBRE DEL LIBRO DESPUÉS DE HABER LLAMADO LA FUNCIÓN CON "APPLY()",

- PROBLEMA 7:



```
1 //Crea un objeto `cuentaBancaria` con propiedades `titular` y `saldo`.
2 //Define una función `actualizarSaldo` que tome un monto y lo sume al
3 //saldo actual de la cuenta. Usa `apply()` para invocar la función
4 //`actualizarSaldo` con el contexto del objeto `cuentaBancaria` y un array
5 //que contenga el monto a agregar.
6
7 const cuentaBancaria = {
8   titular: 'Juan Pérez',
9   saldo: 1000
10 };
11 const monto = 500;
12
13 function actualizarSaldo(monto) {
14   this.saldo += monto;
15 }
16 actualizarSaldo.apply(cuentaBancaria, [monto]);
17 console.log(cuentaBancaria);
18
```

CREAMOS UN OBJETO QUE ES UNA CUENTA BANCARIA Y LA PROPIEDAD QUE TIENE ES "TITULAR" Y "SALDO",
LA FUNCIÓN DONDE SE ACTUALIZA EL SALDO DEL MONTO, CON THIS LO QUE SE HACE ES LLAMAR A LA
PROPIEDAD DECLARADA ANTERIORMENTE DEL MONTO QUE SON 500 EN ESTE CASO, Y LA FUNCIÓN DE
ACTUALIZAR SALDO JUNTO CON APPLY VA A MOSTRAR EN CONSOLA LA INFORMACIÓN DE LA CUENTA
BANCARIA.