Ejercicio 1

Diagrama

Descripción generada automáticamente

class Libro{

constructor(titulo, autor, anioPublicacion){

this.\_titulo= titulo

this.\_autor= autor

this.\_publicacion= anioPublicacion

}

}

class Biblioteca{

constructor(){

this.listBooks=[]

}

agregarLibro(libro){

this.listBooks.push(libro)

}

buscarLibro(titulo){

return this.listBooks.find(libro => libro.\_titulo === titulo)

}

mostrar(){

this.listBooks.forEach(libro=>{

console.log(`Libro: ${libro.\_titulo} - Autor: ${libro.\_autor} - Año de su publicación: ${libro.\_publicacion}`)

})

}

}

const persona1= new Biblioteca();

persona1.agregarLibro(new Libro("El Quijote", "Cervantes", 1605))

persona1.agregarLibro(new Libro("Cien años de soledad", "García Márquez", 1967))

persona1.mostrar()

console.log(persona1.buscarLibro("Cien años de soledad"))

Ejercicio 2

Tabla

Descripción generada automáticamente

class Cuenta{

    constructor(numero, titular, saldo=0){

        this.\_numero= numero

        this.\_titular= titular

        this.\_saldo= saldo

    }

    depositar(cantidad){

        if(cantidad>0){

            this.\_saldo=+cantidad

            console.log(`Su depósito de $${cantidad} ha sido realizado. Saldo depositado: $${this.\_saldo}`)

        }else{

            console.log("El depósito debe ser mayor a 0")

        }

    }

    retirar(cantidad){

        if(cantidad <= this.\_saldo && cantidad >0){

            this.\_saldo-= cantidad

console.log(`Retiro de $${cantidad} realizado. Nuevo saldo: $${this.\_saldo}`)

        }else if(cantidad > this.\_saldo){

            console.log("Fondos insuficientes para su retiro")

        }else{

            console.log("El retiro debe ser mayor a 0")

        }

    }

    consultar(){

        console.log(`El saldo de la cuenta: ${this.\_numero} es de: $${this.\_saldo}`)

    }

}

const persona1 = new Cuenta("123456", "Karla Morales", 200)

persona1.retirar(150)

persona1.consultar()

persona1.depositar(1200)

Ejercicio 3

Diagrama, Tabla

Descripción generada automáticamente

class Empleado {

    constructor(nombre, puesto, salario) {

        this.\_nombre = nombre

        this.\_puesto = puesto

        this.\_salario = salario

    }

    calcular() {

        const salarioAnual = this.\_salario \* 12;

        console.log(`El salario anual de ${this.\_nombre} es: $${salarioAnual}`);

        return salarioAnual;

    }

    aumentar(porcentaje) {

        if (porcentaje > 0) {

            const aumento = this.\_salario \* (porcentaje / 100);

            this.\_salario += aumento;

            console.log(`El salario de ${this.\_nombre} ha sido aumentado en un ${porcentaje}%. Nuevo salario: $${this.\_salario}`);

        } else {

            console.log("El porcentaje de aumento debe ser mayor que 0.");

        }

    }

}

const empleado1 = new Empleado("Damaris López", "Chef", 2500);

empleado1.calcular()

empleado1.aumentar(20)

empleado1.calcular()

Ejercicio 4

Diagrama

Descripción generada automáticamente

class Producto {

    constructor(nombre, precio, stock) {

        this.\_nombre = nombre

        this.\_precio = precio

        this.\_stock = stock

    }

    agregar(cantidad) {

        if (cantidad > 0) {

            this.\_stock += cantidad

            console.log(`Se agregaron ${cantidad} unidades de ${this.\_nombre}. Stock actual: ${this.\_stock}`)

        } else {

            console.log("La cantidad a agregar debe ser mayor que 0.");

        }

    }

    vender(cantidad) {

        if (cantidad > 0 && cantidad <= this.\_stock) {

            this.\_stock -= cantidad;

            console.log(`Se vendieron ${cantidad} unidades de ${this.\_nombre}. Stock actual: ${this.\_stock}`)

        } else if (cantidad > this.\_stock) {

            console.log(`No hay suficiente stock de ${this.\_nombre}. Stock disponible: ${this.\_stock}`)

        } else {

            console.log("La cantidad a vender debe ser mayor que 0.");

        }

    }

    mostrar() {

        console.log(`Producto: ${this.\_nombre}  Precio: $${this.\_precio}  Stock: ${this.\_stock}`)

    }

}

const producto1 = new Producto("Laptop", 1200, 10);

producto1.mostrar()

producto1.agregar(5)

producto1.vender(3)

producto1.vender(15)

producto1.mostrar()

Ejercicio 5

Tabla

Descripción generada automáticamente

class Estudiante {

    constructor(nombre, edad) {

        this.\_nombre = nombre

        this.\_edad = edad

        this.\_calificaciones = []

    }

    agregar(calificacion) {

        if (calificacion >= 0 && calificacion <= 10) {

            this.\_calificaciones.push(calificacion)

            console.log(`Calificación ${calificacion} agregada a ${this.\_nombre}`)

        } else {

            console.log("La calificación debe estar entre 0 y 10")

        }

    }

    calcular() {

        if (this.\_calificaciones.length === 0) {

            console.log(`${this.\_nombre} no tiene calificaciones registradas`

  return 0

        }

        const suma = this.\_calificaciones.reduce((total, calificacion) => total + calificacion, 0)

        const promedio = suma / this.\_calificaciones.length

        console.log(`El promedio de ${this.\_nombre} es: ${promedio.toFixed(2)}`)

        return promedio

    }

    mostrar() {

        console.log(`Estudiante: ${this.\_nombre}  Edad: ${this.\_edad} Calificaciones: [${this.\_calificaciones.join(", ")}]`)

    }

}

const estudiante1 = new Estudiante("María Cea", 20)

estudiante1.agregar(8)

estudiante1.agregar(9)

estudiante1.agregar(7)

estudiante1.calcular()

estudiante1.mostrar()