

UNIVERSIDAD NACIONAL DE INGENIERIA



INTRODUCCION A PROGRAMACION

TITULO: Clase práctica de herencia y encapsulamiento

INTEGRANTE: *Zahid Enrique Esquivel Lainer*

AÑO: 1er año

GRUPO: 1M7-S

CARNET: 2024-1693U

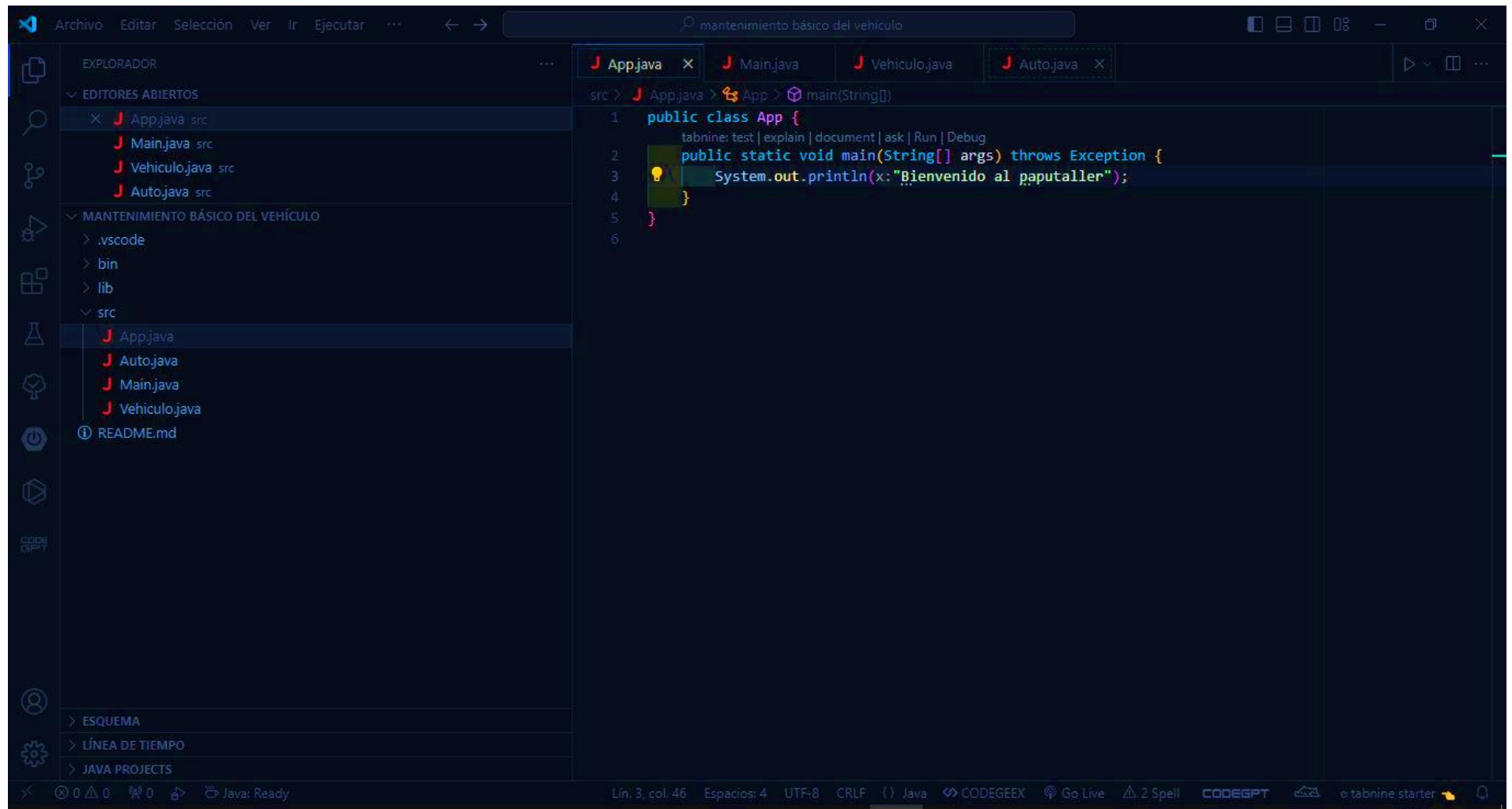
DOCENTE: Ing. Christopher Larios

FECHA DE ENTREGA: 25/5/2024

Agrega un método realizar Mantenimiento en la clase Vehículo que simule un mantenimiento básico del vehículo. Este método será sobrescrito en cada clase derivada para realizar tareas de mantenimiento específicas. En cada clase derivada (Auto, Camión), implementa el método realizar Mantenimiento con un mensaje que describa las acciones específicas de mantenimiento para ese tipo de vehículo. En la clase Auto, agrega un método llamado calcularCostoMantenimiento que utilice programación estructurada para calcular el costo de mantenimiento del auto. Detalles del Método:

- El método debe aceptar como parámetros el costo por kilómetro y el número de kilómetros recorridos.
- Debe calcular el costo total del mantenimiento y devolverlo.
- costo Total= costoPorKm* kilometrosRecorridos;

Mi código, es un código básico, que lo que hice es guiarme en la clase practica y hacerlos un poco parecidos...



Archivo Editar Selección Ver Ir ... mantenimiento básico del vehículo

EXPLORADOR

EDITORES ABIERTOS

- App.java src
- Main.java src
- Vehiculo.java src
- Auto.java src

MANTENIMIENTO BÁSICO DEL VEHÍCULO

- .vscode
- bin
- capturas
- lib
- src
 - App.java
 - Auto.java
 - Main.java
 - Vehiculo.java
- README.md

```
1 class Camion extends Vehiculo {
19
20     tabnine: test | explain | document | ask
21     public int getNivelDeAceiteTransmision() {
22         return nivelDeAceiteTransmision;
23     }
24
25     tabnine: test | explain | document | ask
26     public void setNivelDeAceiteTransmision(int nivelDeAceiteTransmision) {
27         this.nivelDeAceiteTransmision = nivelDeAceiteTransmision;
28     }
29
30     tabnine: test | explain | document | ask
31     @Override
32     public double realizarMantenimiento(double costoPorKilometro, int kilometrosRecorridos) {
33         double costoTotal = super.realizarMantenimiento(costoPorKilometro, kilometrosRecorridos);
34         System.out.println("Realizando mantenimiento específico del camión: Revisión de niveles de aceite del motor y transmisión.");
35         return costoTotal;
36     }
37
38     public class Main {
39         tabnine: test | explain | document | ask | Run | Debug
40         public static void main(String[] args) {
41             double costoPorKilometro = 0.1; // Ejemplo de costo por kilómetro
42             int kilometrosRecorridos = 200; // Ejemplo de kilómetros recorridos
43
44             Automovil auto = new Automovil(marca:"Toyota", modelo:"Corolla");
45             double costoTotalAuto = auto.realizarMantenimiento(costoPorKilometro, kilometrosRecorridos);
46             System.out.println("Costo total de mantenimiento del automóvil: " + costoTotalAuto);
47
48             Camion camion = new Camion(marca:"Volvo", modelo:"FH16");
49             double costoTotalCamion = camion.realizarMantenimiento(costoPorKilometro, kilometrosRecorridos);
50             System.out.println("Costo total de mantenimiento del camión: " + costoTotalCamion);
51
52             double costoMantenimientoAuto = auto.calcularCostoMantenimiento(costoPorKilometro, kilometrosRecorridos);
53             System.out.println("Costo de mantenimiento del automóvil: " + costoMantenimientoAuto);
54         }
55     }
56 }
```

ESQUEMA

LÍNEA DE TIEMPO

JAVA PROJECTS

Lin 39, col 37 Espacios: 4 UTF-8 LF Java CODEGEEK Go Live 72 Spell CODEGPT tabnine starter

Archivo Editar Selección Ver Ir ...

mantenimiento básico del vehículo

EXPLORADOR

EDITORES ABIERTOS

MANTENIMIENTO BÁSICO DEL VEHICULO

ESQUEMA

LÍNEA DE TIEMPO

JAVA PROJECTS

J App.java src

J Main.java src

J Vehiculo.java src

J Auto.java src

README.md

J App.java

J Auto.java

J Main.java

J Vehiculo.java

```
1 public class Vehiculo {
2     private String marca;
3     private String modelo;
4     private int kilometraje;
5
6     public Vehiculo(String marca, String modelo) {
7         this.marca = marca;
8         this.modelo = modelo;
9         this.kilometraje = 0;
10    }
11
12    tabnine: test | explain | document | ask
13    public String getMarca() {
14        return marca;
15    }
16
17    tabnine: test | explain | document | ask
18    public void setMarca(String marca) {
19        this.marca = marca;
20    }
21
22    tabnine: test | explain | document | ask
23    public String getModelo() {
24        return modelo;
25    }
26
27    tabnine: test | explain | document | ask
28    public void setModelo(String modelo) {
29        this.modelo = modelo;
30    }
31
32    tabnine: test | explain | document | ask
33    public int getKilometraje() {
34        return kilometraje;
35    }
36
37    tabnine: test | explain | document | ask
38    public void setKilometraje(int kilometraje) {
39        this.kilometraje = kilometraje;
40    }
41
42    tabnine: test | explain | document | ask
43    public double realizarMantenimiento(double costoPorKilometro, int kilometrosRecorridos) {
44        System.out.println("Realizando mantenimiento básico del vehículo.");
45        double costoTotal = costoPorKilometro * kilometrosRecorridos;
46        return costoTotal;
47    }
48 }
```

Lin. 40, col. 6 Espacios: 4 UTF-8 LF Java CODEGEEY Go Live 39 Spell CODEGPT e tabnine starter

