DIAGRAMAS UML

https://www.plantuml.com/plantuml/uml/SyfFKj2rKt3CoKnELR1Io4ZDoSa700003

ESCENARIO 01

**FACTORY**

@startuml

interface IVehiculo {

+getPlaca(): String

+getMarca(): String

+getAnio(): int

+getValorBase(): double

+calcularMatricula(): double

}

abstract class Vehiculo implements IVehiculo {

-placa: String

-marca: String

-anio: int

-valorBase: double

}

class Auto extends Vehiculo {

+calcularMatricula(): double

}

class Camioneta extends Vehiculo {

+calcularMatricula(): double

}

class Camion extends Vehiculo {

+calcularMatricula(): double

}

class VehiculoFactory {

+crearVehiculo(tipo: String, placa: String, marca: String, anio: int, valorBase: double): Vehiculo

}

Vehiculo <|-- Auto

Vehiculo <|-- Camioneta

Vehiculo <|-- Camion

VehiculoFactory --> Vehiculo

@enduml

**BUILDER**

@startuml

class Vehiculo {

-placa: String

-marca: String

-anio: int

-valorBase: double

-color: String

-cilindraje: double

-numEjes: int

-capacidadCarga: double

+calcularMatricula(): double

}

class VehiculoBuilder {

-placa: String

-marca: String

-anio: int

-valorBase: double

-color: String

-cilindraje: double

-numEjes: int

-capacidadCarga: double

+setPlaca(p: String): VehiculoBuilder

+setMarca(m: String): VehiculoBuilder

+setAnio(a: int): VehiculoBuilder

+setValorBase(v: double): VehiculoBuilder

+setColor(c: String): VehiculoBuilder

+setCilindraje(ci: double): VehiculoBuilder

+setNumEjes(n: int): VehiculoBuilder

+setCapacidadCarga(c: double): VehiculoBuilder

+build(): Vehiculo

}

VehiculoBuilder --> Vehiculo

@enduml

**ESCENARIO 2**

**FACTORY**

@startuml

interface IActivoFijo {

+getCodigo(): String

+getNombre(): String

+getPrecio(): double

}

abstract class ActivoFijo implements IActivoFijo {

-codigo: String

-nombre: String

-precio: double

+getCodigo()

+getNombre()

+getPrecio()

}

class Computador extends ActivoFijo

class Mesa extends ActivoFijo

class Auto extends ActivoFijo

class Silla extends ActivoFijo

class ActivoFijoFactory {

+crearActivo(tipo: String, codigo: String, nombre: String, precio: double): ActivoFijo

}

ActivoFijo <|-- Computador

ActivoFijo <|-- Mesa

ActivoFijo <|-- Auto

ActivoFijo <|-- Silla

ActivoFijoFactory --> ActivoFijo

@enduml

**BUILDER**

@startuml

class ActivoFijo {

-codigo: String

-nombre: String

-precio: double

-color: String

-material: String

-fechaCompra: Date

-vidaUtil: int

-proveedor: String

}

class ActivoFijoBuilder {

-codigo: String

-nombre: String

-precio: double

-color: String

-material: String

-fechaCompra: Date

-vidaUtil: int

-proveedor: String

+setCodigo(codigo: String): ActivoFijoBuilder

+setNombre(nombre: String): ActivoFijoBuilder

+setPrecio(precio: double): ActivoFijoBuilder

+setColor(color: String): ActivoFijoBuilder

+setMaterial(material: String): ActivoFijoBuilder

+setFechaCompra(fecha: Date): ActivoFijoBuilder

+setVidaUtil(vida: int): ActivoFijoBuilder

+setProveedor(proveedor: String): ActivoFijoBuilder

+build(): ActivoFijo

}

ActivoFijoBuilder --> ActivoFijo

@enduml