



Anexo

Diseño de clases. Se debe utilizar como definición para la resolución de los ejercicios.

Tener en cuenta que en estos pseudo diagramas de clases, se utiliza la siguiente notación:

“+/-”: Indica el modificador de acceso del atributo o método:

- Privado

+ Publico

“: tipo”: Indica el tipo de dato del atributo cuando se indica a continuación de uno, o el tipo de dato que devuelve un método cuando esta a continuación de este ultimo.

Atributo en MAYUSCULA: Indica que dicho atributo es una constante.

Atributo o método en *cursiva*: Indica que el atributo o método es de tipo static.

Persona

Persona
<i>Atributos</i>
- nombre: String - genero: String - dni: long - peso: double - altura: double - edad: int - viva: boolean
<i>Constructores</i>
+Persona(nombre: String, genero: String, dni: long, peso: double, altura: double)
<i>Metodos</i>
+toString(): String +alimentar(kilos: double): void +crecer(metros: double): void +cumplirAnos(): void +morir(): void +pesar(): double



Ascensor

Ascensor
<i>Atributos</i>
-PISO_MAXIMO: int -PISO_MINIMO: int -PESO_MAXIMO: double -CAPACIDAD_MAXIMA: int - puertaAbierta: boolean - sobrecarga: boolean - <i>pisoActual</i> : long - pesoActual: double - capacidadActual: int - ocupantes: Persona[]
<i>Constructores</i>
+Ascensor(capacidadMaxima: int, pesoMaximo: double)
<i>Metodos</i>
+ingresar(elQueSube: Persona): void +salir(elQueBaja: Persona): void +abrirPuerta(): void +cerrarPuerta(): void +subir(): void +bajar(): void +getPisoActual(): int +irAlPiso(pisoDeseado:int): void

Cuenta

Cuenta
<i>Atributos</i>
- titular: String - saldo: double
<i>Constructores</i>
+Cuenta(titular: String) +Cuenta(titular: String, saldoInicial: double)
<i>Metodos</i>
+getTitular(): String +setTitular(titular: String): void +getSaldo(): double +toString(): String +depositar(cantidad: double): boolean +retirar(cantidad: double): boolean



Coche

Coche	
Atributos	
- CERO_KM: int	
- ANO_ACTUAL: int	
- <i>cantidadCoches</i> : static int	
- marca: String	
- modelo: String	
- kilometros: int	
- ano: int	
- precio: double	
Constructores	
+Coche(marca: String, modelo: String, precio: double)	
+Coche(marca: String, modelo: String, kilometros: int, ano: int, precio: double)	
Metodos	
+getCantidadCoches(): static int	
+setKilometros(kilometros: int): void	
+setPrecio(precio: int): void	
+getPrecio(): double	+getMarca() : String
+calcularAntiguedad(): int	+getModelo() :String
+getKilometros(): int	

Calculadora

Calculadora	
Atributos	
- operador1: double	
- operador2: double	
Constructores	
+Calculadora()	
+Calculadora(operador1: double, operador2: double)	
Metodos	
+sumar(): double	
+sumar(operador1: double, operador2: double): double	
+retar(): double	
+restar(operador1: double, operador2: double): double	
+multiplicar(): double	
+multiplicar(operador1: double, operador2: double): double	
+dividir(): double	
+dividir(numerador: double, divisor: double): double	



Usuario

Usuario
<i>Atributos</i>
- usuario: String - contraseña: String - nombre: String - apellido: String - dni: int - edad: int
<i>Constructores</i>
+Usuario(usuario: String, contraseña: String, nombre: String , apellido: String , dni: int, edad: int)
<i>Metodos</i>
+getUsuario(): String +getContraseña(): String +setContraseña(contraseña: String): void +setEdad(edad: int): void

Nota

Nota
<i>Atributos</i>
- valor: float
<i>Constructores</i>
+Nota(valorInicial: float)
<i>Metodos</i>
+getValor(): float +aprobado(): boolean +recuperar(nuevoValor: float): void
+promocionado() : boolean

Cubo

Cubo
<i>Atributos</i>
- longitudLado: float
<i>Constructores</i>
+Cubo(lado: float)
<i>Metodos</i>
+getLongitudLado(): float +setLongitudLado(longitudLado: float): void +getSuperficieCara(): float +getVolumen(): float



Cerradura

Cerradura
<i>Atributos</i>
- claveDeApertura: int - cantidadDeFallosConsecutivosQueLaBloquean: int
<i>Constructores</i>
+Cerradura(claveDeApertura: int, cantidadDeFallosConsecutivosQueLaBloquean: int)
<i>Metodos</i>
+abrir(clave: int): boolean +cerrar(): void +estaAbierta(): boolean +fueBloqueada(): boolean

Sensor

Sensor
<i>Atributos</i>
- estado: boolean
<i>Constructores</i>
+Sensor()
<i>Metodos</i>
+activado(): boolean +activar(): void +desactivar(): void

Alarma

Alarma
<i>Atributos</i>
- sensorDeMovimiento: Sensor - sensorDeContacto: Sensor - sensorDeSonido: Sensor - encendida: boolean - codigoDeSeguridad: int
<i>Constructores</i>
+Alarma(codigoDeSeguridad: int)
<i>Metodos</i>
+encender(): void +apagar(): void +activada(): boolean



Temperatura

Temperatura
<i>Atributos</i>
+ CELSIUS: byte + FARENHEIT: byte + KELVIN: byte - valor: double
<i>Constructores</i>
+Temperatura(valor: double)
<i>Metodos</i>
+setValor(valor: double) +setValor(valor: double, unidad: byte) +getValor(): double +convertir(unidad: byte): double

Distancia

Distancia
<i>Atributos</i>
+ DECAMETRO: byte + HECTOMETRO: byte + KILOMETRO: byte - valor: double
<i>Constructores</i>
+Recorrido(valor: double)
<i>Metodos</i>
+setValor(valor: double) +setValor(valor: double, unidad: byte) +getValor(): double +convertir(unidad: byte): double

Tornillo

Tornillo
<i>Atributos</i>
- TIPO_DE_CABEZA: char - LONGITUD: int - CANTIDAD_DE_ROSCA: int - posicionActual: int
<i>Constructores</i>
+ Tornillo(tipoDeCabeza: char, longitud: int, cantidadDeRosas: int)
<i>Metodos</i>
+ getLongitud(): int + getTipoCabeza(): char + girar(sentido: char): boolean



Tarugo

Tarugo
<i>Atributos</i>
- LONGITUD: int
<i>Constructores</i>
+ Tarugo(longitud: int)
<i>Metodos</i>
+getLongitud(): int

Destornillador

Destornillador
<i>Atributos</i>
- TIPO_DE_CABEZA: char
<i>Constructores</i>
+ Destornillador(tipoDeCabeza: char)
<i>Metodos</i>
+desatornillar(tornillo: Tornillo): void
+atornillar(tornillo: Tornillo): void
+desatornillar(tornillo: Tornillo, tarugo: Tarugo): void
+atornillar(tornillo: Tornillo, tarugo: Tarugo): void

Tambor

Tambor
<i>Atributos</i>
- POSICION_MINIMA: int
- POSICION_MAXIMA: int
- posicion: int
<i>Constructores</i>
<i>Metodos</i>
+getPosicion(): int
+girar(): void



Tragamonedas

Tragamonedas
<i>Atributos</i>
-tambor1: Tambor
-tambor2: Tambor
-tambor3: Tambor
<i>Constructores</i>
+Tragamonedas()
<i>Metodos</i>
+getPosicion(): int
+girar(): void