



Enterprise Java Developer



Programación Orientada a Objetos

Clases y Objetos

Eric Gustavo Coronel Castillo
gcoronelc@gmail.com
gcoronelc.blogspot.com

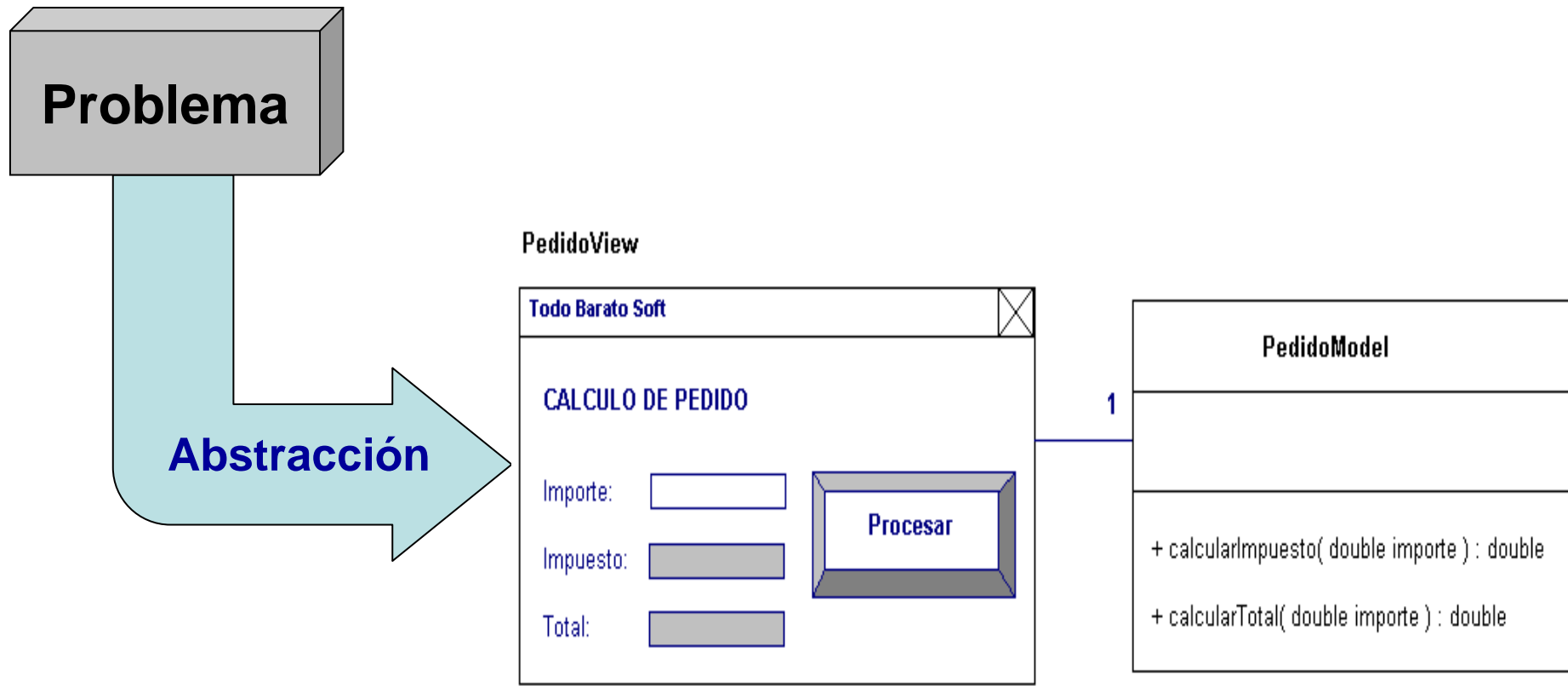
Ricardo Walter Marcelo Villalobos
ricardomarcelo@hotmail.com

ÍNDICE

- Objetivo
- Abstracción
- Definición de clase y objeto
- Implementación de clases
- Creación y uso de objetos
- Proyecto ejemplo
- Bibliografía

OBJETIVO

Entender los conceptos de Clase y Objeto, y su aplicación en la solución de problemas sencillos.

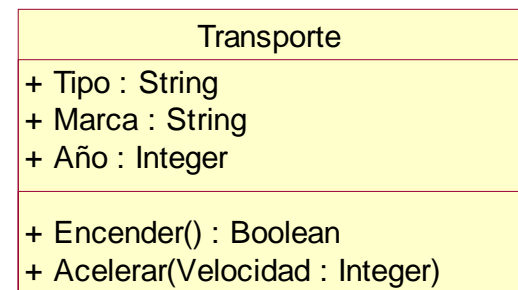
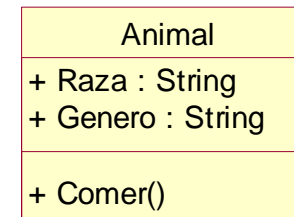
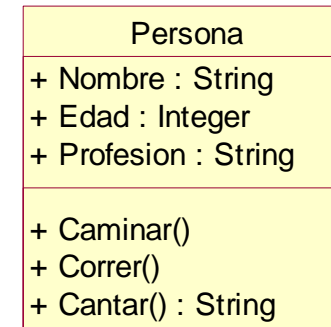
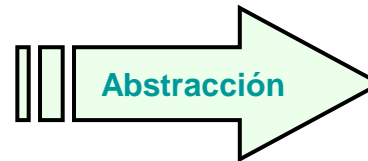


ABSTRACCIÓN

Consiste en capturar, percibir y clasificar las características (datos-atributos) y comportamientos (operaciones) necesarias (relevantes) del mundo real (proceso a sistematizar) para dar solución al problema.



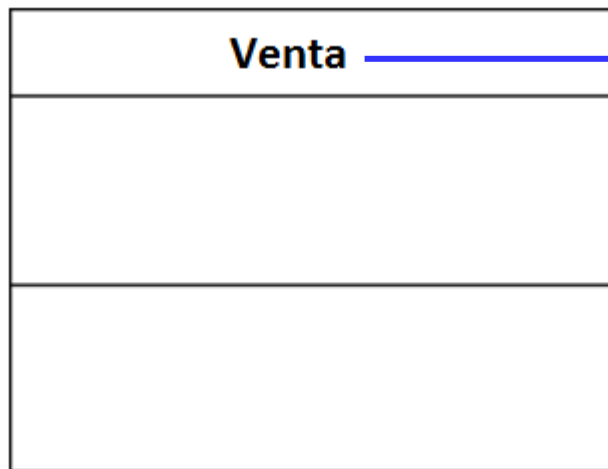
Notación UML



DEFINICIÓN DE CLASE Y OBJETO

CLASE

- Una clase define un tipo de objeto en particular.
- Por ejemplo, la clase Empleado define a todos los trabajadores de una empresa.



→ Nombre de la Clase

Ejemplos de Nombres de Clase

- Cliente
- Factura
- NotaCredito
- Guia
- Pedido
- Matricula
- CuentaMaestra

DEFINICIÓN DE CLASE Y OBJETO

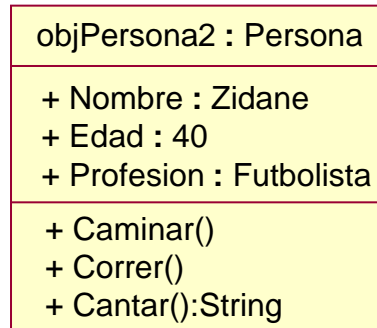
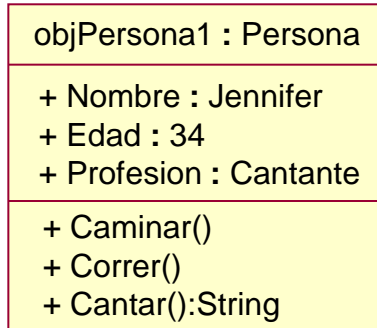
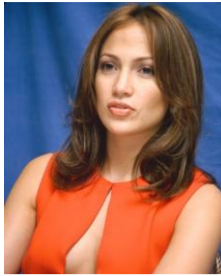
OBJETO

- Un objeto es una instancia de una clase.
- Por ejemplo, cada trabajador de una empresa es una instancia de la clase **Empleado**.

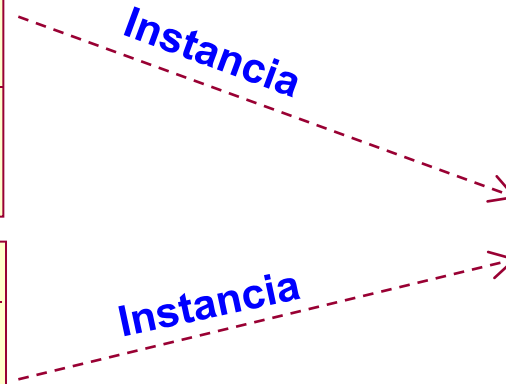
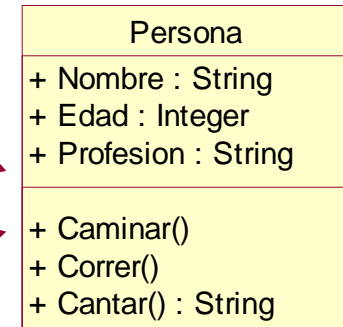


DEFINICIÓN DE CLASE Y OBJETO

Notación UML de OBJETO



Notación UML de Clase



IMPLEMENTACIÓN DE CLASES

SINTAXIS

```
public class NombreClase {  
  
    // Definición de variables  
  
    // Definición de métodos  
  
}
```

El nombre del archivo debe tener el mismo nombre de la clase.

Por ejemplo, si la clase se llama **Producto** el nombre del archivo que contiene a la clase se debe llamar **Producto.java**.

IMPLEMENTACIÓN DE CLASES

ATRIBUTOS

- Representa un dato del objeto.
- Cada atributo de un objeto tiene un valor que pertenece a un dominio de valores determinado.
- En Java se implementan creando variables a nivel de clase.

Venta
<ul style="list-style-type: none">- id : Integer- fecha: Date- cliente: String- importe: Double...

```
public class Venta {  
  
    // Variables que implementación de atributos  
    private Integer id;  
    private Date fecha;  
    private String cliente;  
    private Double importe;  
  
}
```

IMPLEMENTACIÓN DE CLASES

OPERACIONES

- Son servicios proporcionado por un objeto que pueden ser solicitados por otros objetos.
- Determinan el comportamiento del objeto.
- La implementación en Java se realiza mediante métodos,

Venta
<ul style="list-style-type: none">- id : Integer- fecha: Date- cliente: String- importe: Double...
<ul style="list-style-type: none">+ buscar() : boolean+ insertar() : void+ modificar() : void+ eliminar() : void...

```
public class Venta {  
  
    // Implementación de atributos  
    private Integer id;  
    ...  
  
    // Implementación de operaciones  
    public boolean buscar( . . . ) {  
        ...  
        ...  
    }  
  
    ...  
  
}
```

IMPLEMENTACIÓN DE CLASES

DEFINICIÓN DE MÉTODOS

```
public <tipo> nombreMétodo ( [ parámetros ] ) {  
  
    // Implementación  
  
    [ return valorRetorno; ]  
}
```

- <tipo>** Determina el tipo de dato que retorna el método, si no retorna ningún valor se utiliza **void**.
- return** Esta sentencia finaliza la ejecución del método, se acompaña de un valor cuando el método debe retornar un resultado.

CREACIÓN Y USO DE OBJETOS

OPERADOR NEW

NombreClase variable = new NombreClase();

ó

NombreClase variable = null;
variable = new NombreClase();

ACCESO A LOS MÉTODOS

variable.nombreMétodo (...)

PROYECTO EJEMPLO

La empresa "Todo Barato" necesita facilitar la elaboración de los pedidos que realizan sus empleados a sus proveedores, el problema radica al momento de calcular el impuesto.

La empresa ha solicitado a su departamento de sistemas elaborar un programa en Java que permita ingresar el importe del pedido, y calcule el impuesto y el total que se debe pagar al proveedor.

Bibliografía

