

TALLERES ORACLE

INSTRUCTOR : ING. FORTUNATO VELIZ

WWW.TALLERESORACLE.COM

Programación Orientada a Objetos



Clases y Objetos

Ing. Fortunato Veliz
superveliz@gmail.com

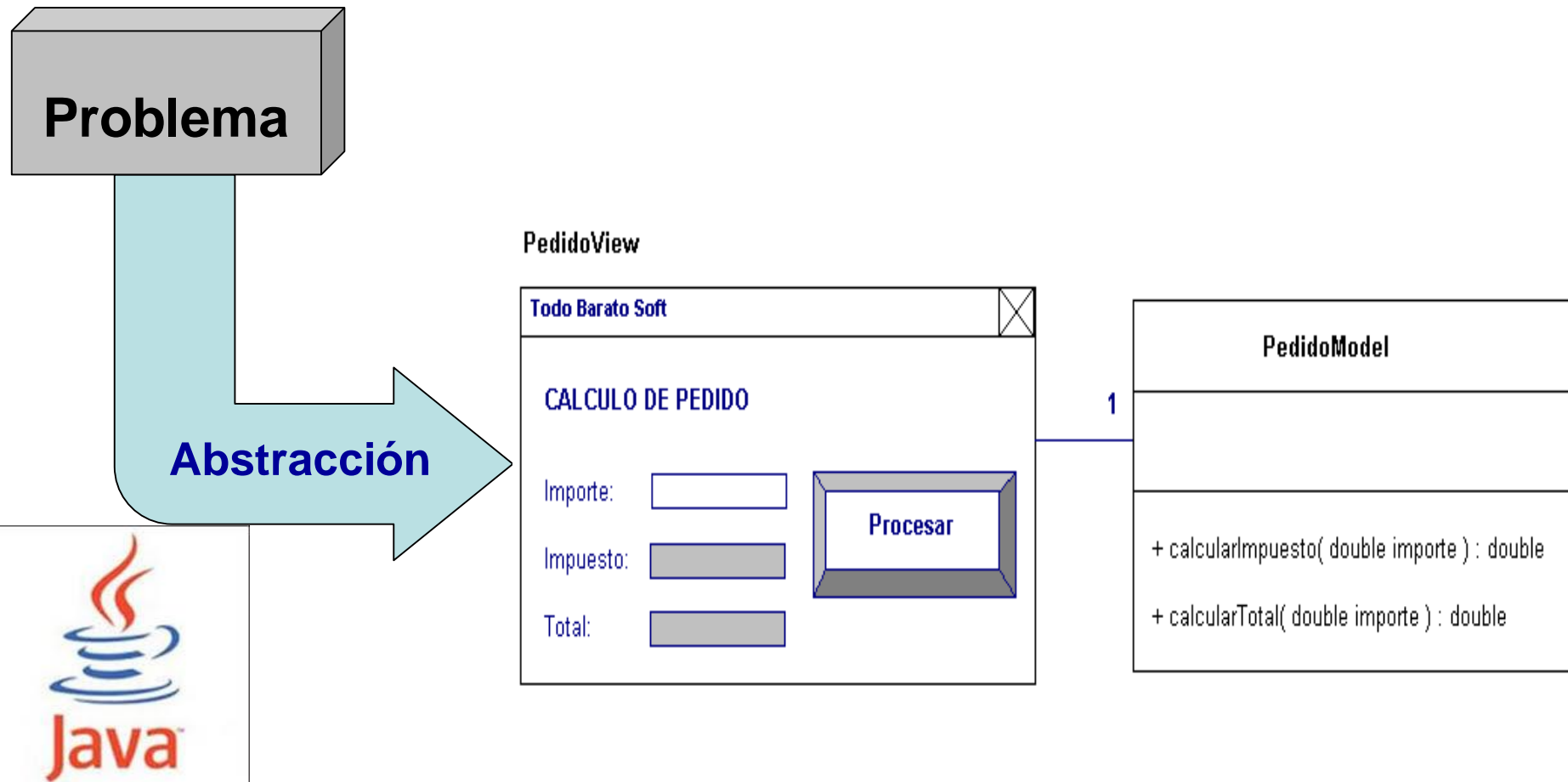
ÍNDICE

- Objetivo
- Abstracción
- Definición de clase y objeto
- Implementación de clases
- Creación y uso de objetos
- Proyectos propuestos



OBJETIVO

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



Qué es la orientación por objeto

- El término "orientación por objeto" significa que organizaremos el software como una colección de objetos discretos que incorporan tanto estructuras de datos como procedimientos.
- Esto contrasta con la programación convencional, en la cual las estructura de datos y el comportamiento están solo aproximadamente conectados

Qué es el desarrollo Orientado por Objetos

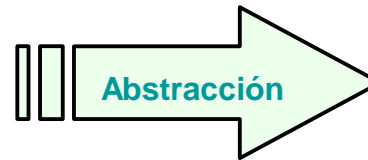
- El desarrollo orientado por objetos es una nueva forma de pensar acerca del software basado sobre abstracciones que existen en el mundo real.
- En este contexto, el desarrollo es referido a la primera parte del ciclo de vida del software: análisis, diseño e implantación.

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



Persona
+ Nombre : String
+ Edad : Integer
+ Profesion : String
+ Caminar()
+ Correr()
+ Cantar() : String

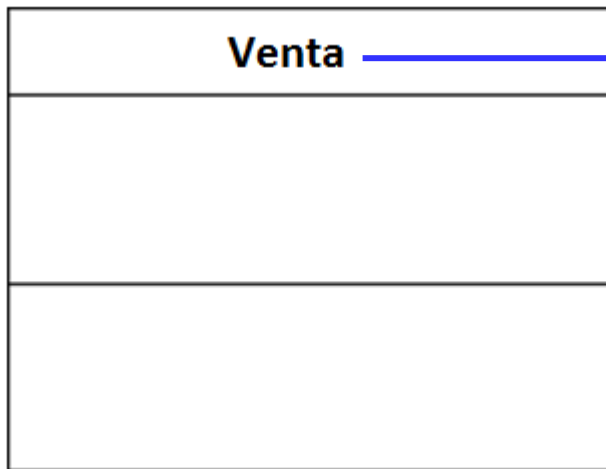
Animal
+ Raza : String
+ Genero : String
+ Comer()

Transporte
+ Tipo : String
+ Marca : String
+ Año : Integer
+ Encender() : Boolean
+ Acelerar(Velocidad : Integer)

DEFINICIÓN DE CLASE Y OBJETO

CLASE

- Las clases son plantillas que agrupan comportamiento (métodos) y estados (atributos) de los futuros objetos.
- 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



objPersona1 : Persona

+ Nombre : Jennifer
+ Edad : 34
+ Profesion : Cantante

+ Caminar()
+ Correr()
+ Cantar():String



objPersona2 : Persona

+ Nombre : Zidane
+ Edad : 40
+ Profesion : Futbolista

+ Caminar()
+ Correr()
+ Cantar():String

Instancia

Instancia

Notación UML de Clase

Persona

+ Nombre : String
+ Edad : Integer
+ Profesion : String

+ Caminar()
+ Correr()
+ Cantar() : String



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 una propiedad de una entidad.
- 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 la clase que pueden ser solicitados por otras clases.
- 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 PROPUESTO

PROYECTO 01

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.



PROYECTO PROPUESTO

PROYECTO 02

El Director Académico del Instituto EduTec ha solicitado al Departamento de Sistemas un programa en Java que permita a sus profesores calcular el promedio de un alumno.

Se saben que son cuatro practicas y que se promedian las tres mejores. También se tiene un examen parcial y un examen final.

Los pesos son los siguientes:

- Promedio de Practicas (PP) : 30%
- Examen Parcial (EP): 30%
- Examen Final (EF): 40%

Demos

