

Programación Orientada a Objetos



Unidad 1 – Desarrollo en Capas Sesión 1- Clases y Objetos

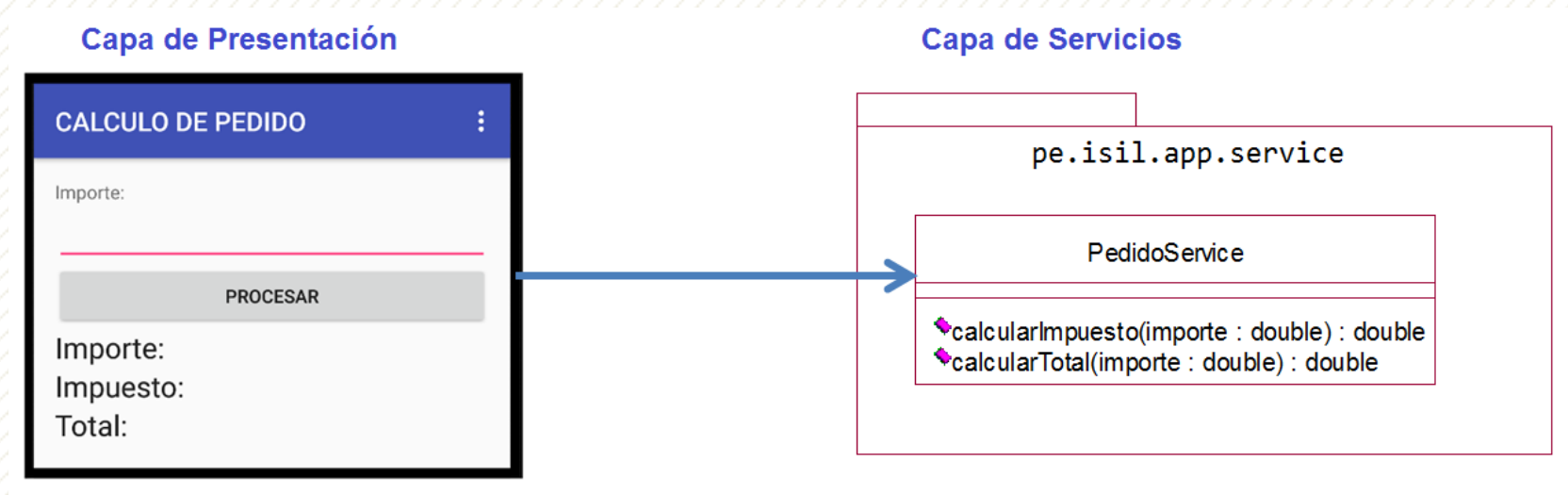
- **Arquitectura en Capas**
- **Clases y Objetos**
- **Implementación de Clases**
- **Creación y Uso de Objetos**



1. Entender la arquitectura en capas de una aplicación.
2. Aplicar la programación en capas para resolver problemas sencillos.

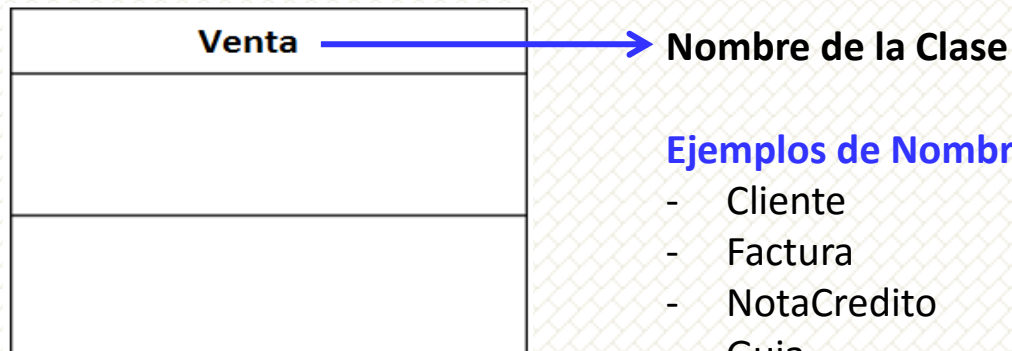


La arquitectura basada en capas se enfoca en la distribución de roles y responsabilidades de forma jerárquica proveyendo una forma muy efectiva de separación de responsabilidades. El rol indica el modo y tipo de interacción con otras capas, y la responsabilidad indica la funcionalidad que está siendo desarrollada.



CLASE

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

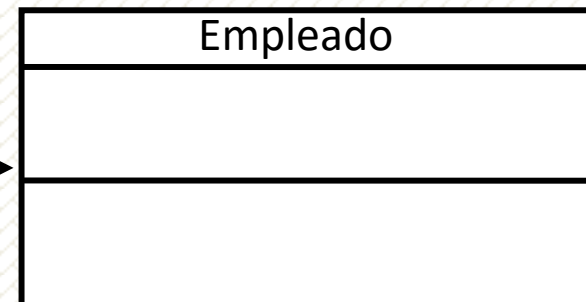


Ejemplos de Nombres de Clase

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

OBJETO

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



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**.

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;

}
```


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( ... ) {
        ...
    }

    ...
}
```

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. |

OPERADOR NEW

```
NombreClase variable = new NombreClase();
```

ó

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

ACCESO A LOS MÉTODOS

```
variable.nombreMétodo ( ... )
```