

PROGRAMACIÓN ORIENTADA A OBJETOS - 22951

Universidad Industrial de Santander
Escuela de Ingeniería de Sistemas e Informática
Programa de Ingeniería de Sistemas



#LaUISqueQueremos





Universidad
Industrial de
Santander

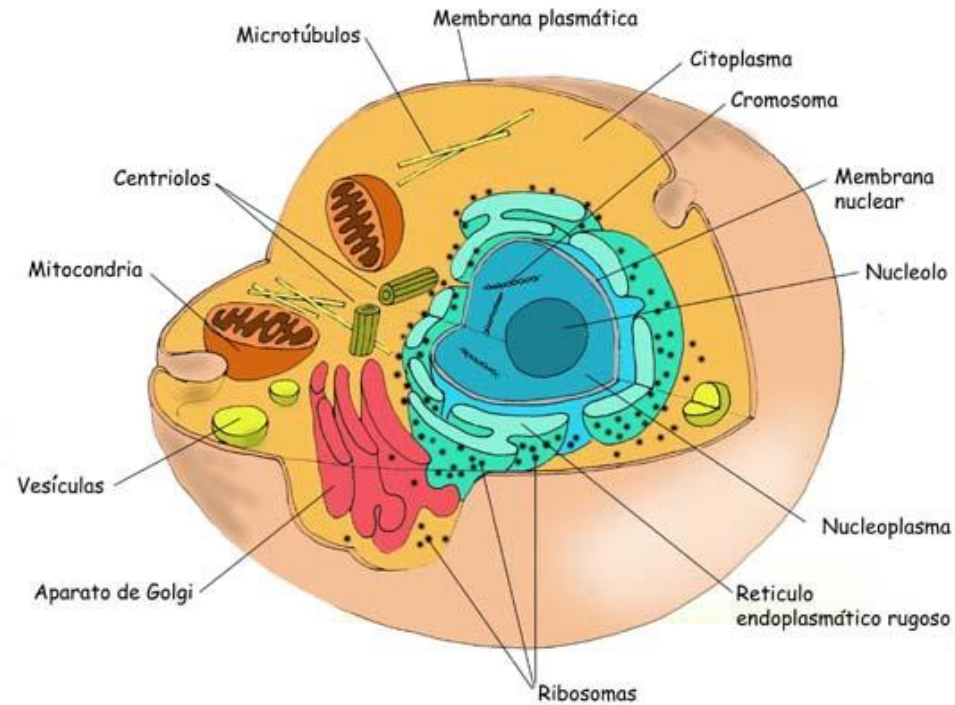
Encapsulamiento



Encapsulación



Universidad
Industrial de
Santander



Encapsular. Proteger, esconder, separar.



Universidad
Industrial de
Santander

ESCONDER LA IMPLEMENTACIÓN





```
public class Cuadrado {  
  
    public double lado;  
  
    public double area() {  
        return lado * lado;  
    }  
  
}
```

```
public class Cuadrado {  
  
    public double lado;  
  
    public double area() {  
        return Math.pow(lado, 2);  
    }  
  
}
```

```
public class Principal {  
    public static void main(String[] args) {  
        Cuadrado c1 = new Cuadrado();  
        c1.lado = 20.0;  
        System.out.println(c1.area());  
    }  
}
```

Encapsulación

- El paradigma OO pretende:
 - Separar las cosas que cambian de las que permanecen iguales.
- Importante para las librerías:
 - El cliente (desarrollador) confía en que su código siga funcionando cuando una nueva versión es liberada.
 - El desarrollador de la librería debe tener la libertad de hacer los cambios necesarios y mejoras.
- Una opción es usar una convención:
 - Podría funcionar para los métodos.
 - ¿Que sucede con los atributos?.



Ejemplo Intuitivo



Cliente



API

- Ajustar volumen
- Cambiar canal
- Encender/Apagar

El cliente necesita
saber cómo usar el
API



Implementación

- Tubos catódicos
- 20" y 22 Kg
- Sony Triniton

La implementación
necesita saber cual API
implementar

El cliente y la implementación se ponen de acuerdo en el API



Universidad
Industrial de
Santander



Ejemplo Intuitivo



Cliente



API

- Ajustar volumen
- Cambiar canal
- Encender/Apagar

El cliente necesita
saber cómo usar el
API



Implementación

- LED – Pantalla HD
- 40" y 12 Kg
- Samsung Smart TV

La implementación
necesita saber cual API
implementar



Universidad
Industrial de
Santander



Encapsulación

- Una propiedad fundamental de la POO es esconder la implementación de las clases.
- Esta propiedad esta presente en objetos de uso diario:
 - Utilizar un automóvil no requiere conocimientos avanzados de mecánica.
 - Utilizar un TV no requiere conocimientos de electrónica.
- Un método que calcule la raíz cuadrada:
 - Estimación.
 - Método de babilonia.
 - Cálculo digito por digito.





Interfaz de clase

- Cada clase en POO provee su interfaz.
- La interfaz de la clase es la parte visible de la clase.
- Para definir la visibilidad existen un conjunto de reglas:
 - Cada clase especifica el acceso a sus miembros (atributos y métodos).
 - El usuario de la clase solo puede acceder a los miembros si la especificación lo permite.
- Para definir la visibilidad en Java se usan las palabras claves:
 - public, private o ninguno (acceso de paquete).





Universidad
Industrial de
Santander

PROTEGER LOS ATRIBUTOS



Atributos Accedidos Libremente



Universidad
Industrial de
Santander

```
public class Persona {  
  
    public String nombre;  
    public int edad;  
  
}
```

```
public class ClientePersona1 {  
  
    public static void main(String[] args) {  
        Persona per = new Persona();  
        // Es esta una edad normal?  
        per.edad = 2425;  
        // Puede un nombre tener 54 caracteres?  
        per.nombre = "Pedro Pablo Perico Pico Perez Paez Panchoba";  
    }  
}
```



Ejemplo

Persona
-nombre: String -edad: int
+Persona() +setNombre(String nNombre) +getNombre(): String +setEdad(int nEdad) +getEdad(): int -validar(): boolean

- El cliente ve el API de la clase *Persona* con:
 - Un constructor
 - 4 métodos *getNombre()*, *setNombre()*, *getApellido()* y *setApellido()*



```
public class Persona {  
  
    private String nombre;  
    private int edad;  
  
    public int getEdad() {  
        return edad;  
    }  
  
    public void setEdad(int ed) {  
        if (edad < 120)  
            edad = ed;  
    }  
  
    public String getNombre() {  
        return nombre;  
    }  
  
    public void setNombre(String nNombre) {  
        boolean valido = validar(nNombre);  
        if (valido) {  
            nombre = nNombre;  
        }  
    }  
  
    private boolean validar(String texto) {  
        return true;  
    }  
  
}
```



Universidad
Industrial de
Santander




```
public class Persona {  
  
    private String nombre;  
    private int edad;  
  
    public int getEdad() { return edad; }  
  
    public void setEdad(int ed) {  
        if (edad < 120)  
            edad = ed;  
    }  
  
    public String getNombre() { return nombre; }  
  
    public void setNombre(String nNombre) {  
        boolean valido = validar(nNombre);  
        if (valido) {  
            nombre = nNombre;  
        }  
    }  
  
    private boolean validar(String texto) {  
        if (texto.length() < 50)  
            return true;  
        else  
            return false;  
    }  
}
```



Universidad
Industrial de
Santander



Usando la clase - OK

```
public class ClientePersona2 {  
  
    public static void main(String[] args) {  
        Persona per = new Persona();  
        per.setNombre("Juan");  
        per.setEdad(21);  
        String nombrePer = per.getNombre();  
        int edadPer = per.getEdad();  
        System.out.println("Nombre : " + nombrePer);  
        System.out.println("Edad : " + edadPer);  
    }  
}
```



Universidad
Industrial de
Santander



Usando la clase – NO OK

```
public class ClientePersona3 {  
  
    public void metodoSimple() {  
        Persona per = new Persona();  
        per.setNombre("Juan");  
        per.setEdad(21);  
        per.validar("Juan");  
        String nombrePer = per.getNombre();  
        int edadPer = per.edad;  
        System.out.println("Nombre : " + nombrePer);  
        System.out.println("Edad : " + edadPer);  
    }  
}
```

Description

Errors (2 items)

The field `Persona.edad` is not visible

The method `validar(String)` from the type `Persona` is not visible



Universidad
Industrial de
Santander



Reglas Visibilidad Java



Universidad
Industrial de
Santander

Modificador	Clase	Paquete	Resto
private	Si	No	No
public	Si	Si	Si
sin modificador	Si	Si	No





Universidad
Industrial de
Santander



#OrgulloEISI

iGracias!

#LaUISqueQueremos