



**USMP**

FACULTAD DE  
INGENIERÍA Y ARQUITECTURA

Semestre Académico 2022- I



**ASIGNATURA**

# Algoritmo y Estructura de Datos I

---

Docente: Juan Alberto Puerta Arce

**Escuela Profesional de  
Ingeniería de Computación y Sistemas**

# Semana 1

- Concepto de la Teoría Orientada a Objetos (TOO). Definición de Clase y Objeto. Características de las Clases. Diagrama de Clase.
- Componentes de una clase, atributos y métodos. Correlación con el lenguaje de programación.

# Algoritmo

Conjunto de instrucciones ejecutadas de manera lógica (tiene un orden específico) y finita, que permite resolver un problema.



# Representación de un Algoritmo

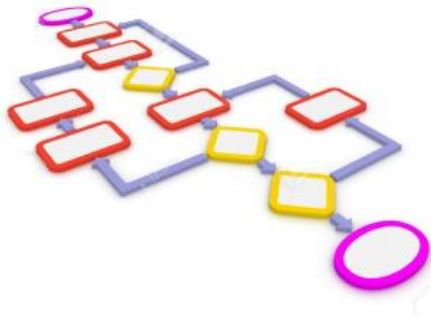


Diagrama de Flujo

```

procedimiento Visval (linein: array [1 .. n][2] de reales)
  lineout[1]=linein[1];      El primer punto pasa a la línea de salida
  npin=n;                    Inicialización n° puntos línea de entrada
  npout=1;                    Inicialización n° puntos línea de salida
  mientras (npin>2)
    hallar punto con area efectiva minima, amin;
    eliminar este punto;
    area[punto]=amin;        Se da un valor de área efectiva a cada punto
  fin de mientras
  desde (i=2;i≤npin-1;i++)
    si (area[i]≥tolerancia)
      npout++;
      lineout[npout]=linein[i];
    fin de si
  fin de desde
  npout++;
  lineout[npout]=linein[npin]; El último punto pasa a la línea de salida
fin del procedimiento
  
```

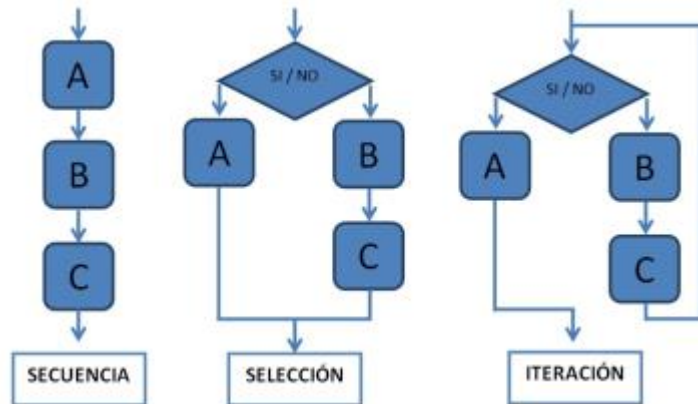
Seudocódigo



Lenguaje de Programación

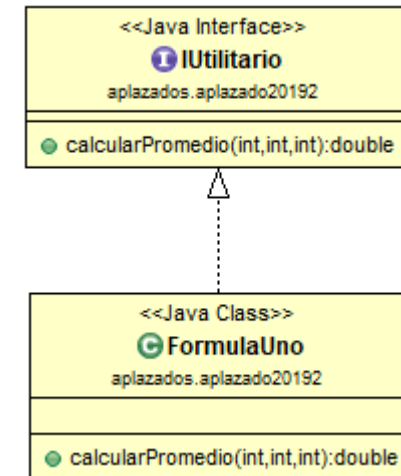
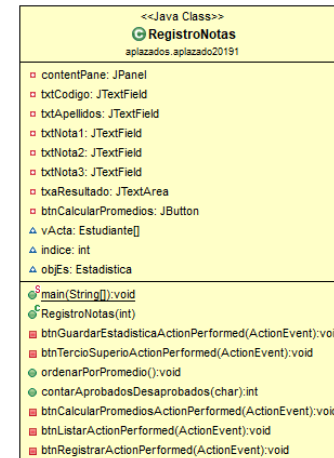
# Enfoques de programación

## Programación Estructurada



- Más fácil de leer.
- Más ordenado.

## Programación Orientada a Objetos (usando como base la Teoría Orientada a Objetos)



- Reutilización de código
- Soluciones mas manejables

# Teoría Orientada a Objetos (TOO)

- El mundo que nos rodea está lleno de elementos, animados o inanimados, concretos o abstractos, los cuales poseen propiedades y pueden tener una acción asociada.
- La TOO representa (abstrae) elementos existentes en el mundo que nos rodea, en términos de objetos los cuales poseen atributos y métodos, y son únicos.

## Mundo real

Propiedades  
Acciones/Operaciones



## En la TOO

Atributos  
Métodos

# Abstracción



Representar en  
términos de  
objetos

Nombre del objeto

Atributos

Métodos

Representación  
gráfica de un  
objeto



# Ejemplo de objeto



## Propiedades

Color: negro

Numero de llantas: 4

Numero de asientos: 4

Marca: bmw

Modelo: 6xi

Velocidad máxima: 220

Numero de cambios: 6

## Operaciones

Encender

Avanzar

Retroceder

Encender las luces

Activar alarma

# objeto



# Ejemplo de objeto



TOO

**objeto**

xfg312 : Auto

color: negro  
numllan: 4  
numasi: 4  
marca: bmw  
modelo: 6xi  
velmax: 220  
numcam: 6

encender() : int  
avanzar(char) : void  
retroceder() : void  
encender\_luces() : void  
activar\_alarma(int): char



# Clase

Cuando se desea  
representar un grupo de  
objetos con atributos y  
métodos idénticos, se  
usa una clase.

# Ejemplo de Clase

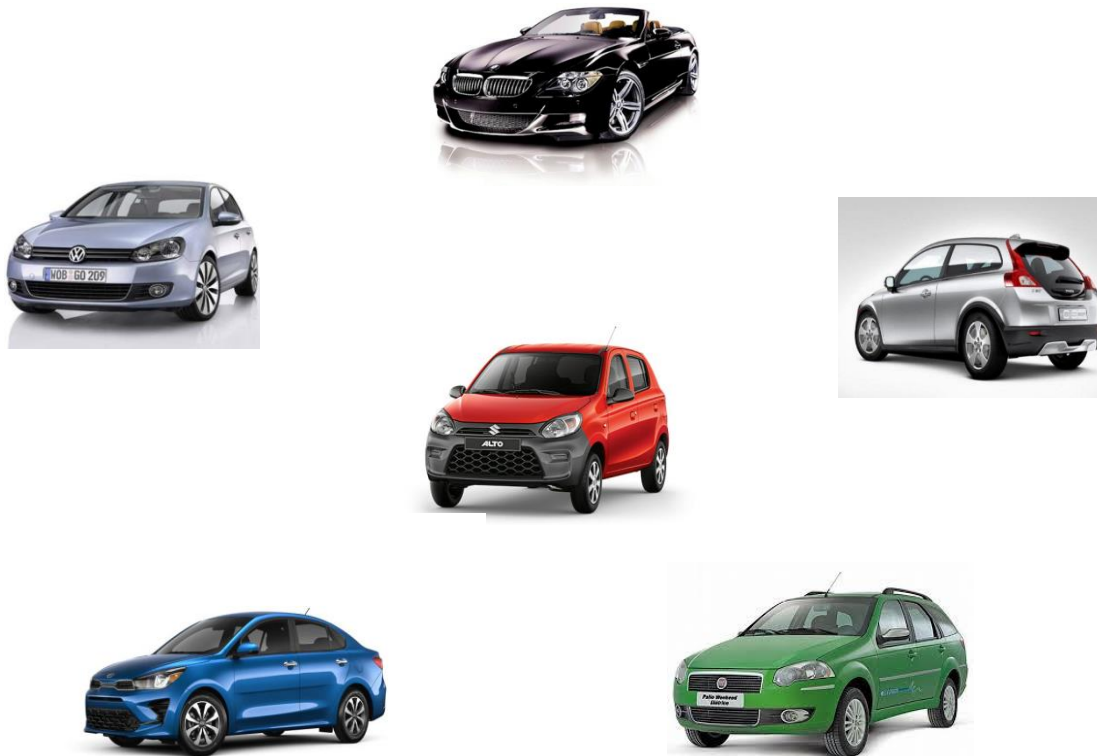


Nombre de la  
Clase

atributos

métodos

# Ejemplo de Clase



## Auto

color: string  
numllan: int  
numasi: int  
marca: string  
modelo: string  
velmax: int  
numcam: int

encender() : int  
avanzar(char) : void  
retroceder() : void  
encender\_luces() : void  
activar\_alarma(int): char

# Correlación con el lenguaje de programación

TOO → POO

A yellow sticky note with a red pushpin is located on the left side of the slide. The note has the text "Recordar!!" written on it in a handwritten style.

Recordar!!

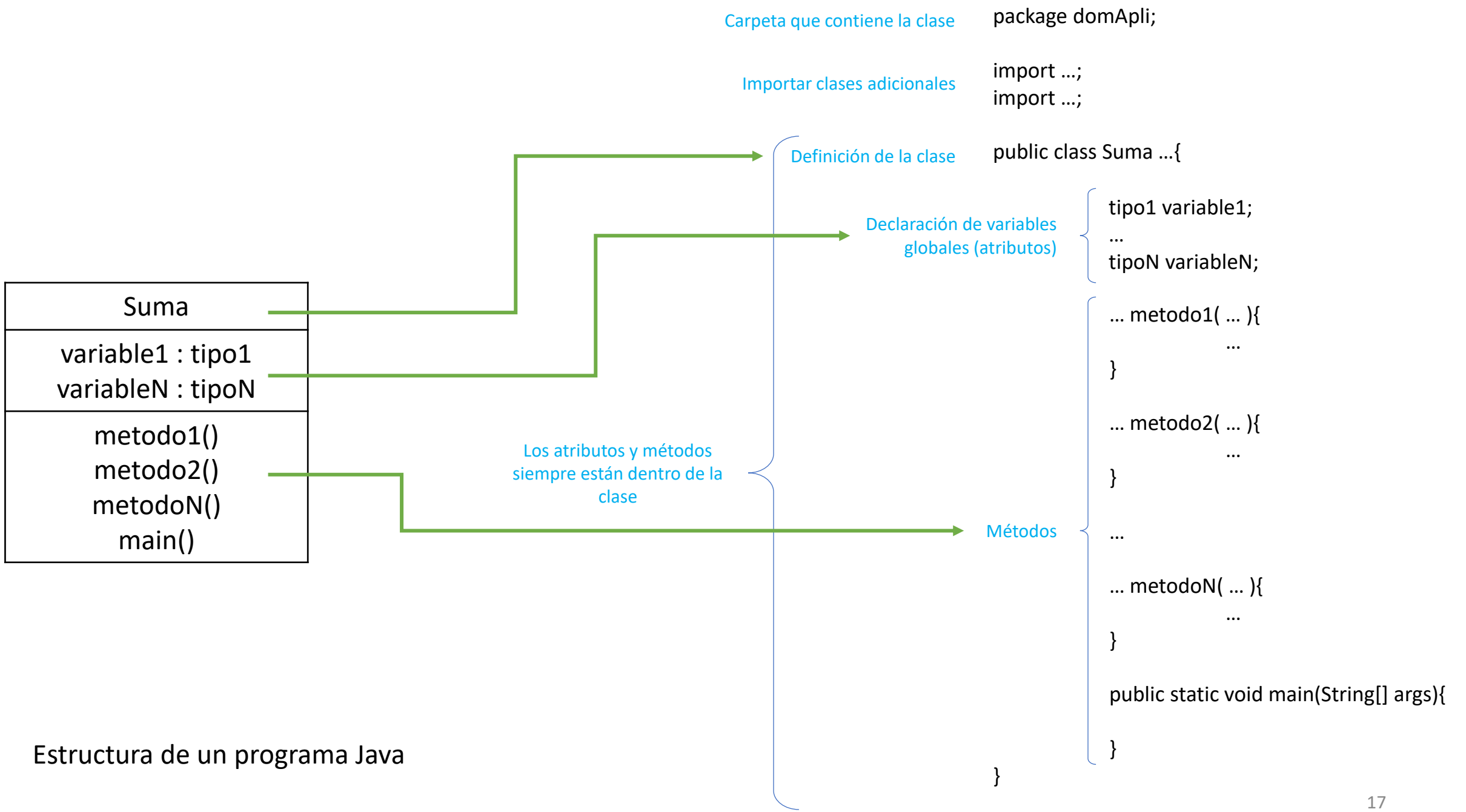
Una **CLASE** se  
“traduce” en la  
estructura de  
un **PROGRAMA**



# Clase - Programa

TOO	POO
Nombre de la Clase	Nombre del programa
Atributos	Variables globales
Métodos	Subprogramas

Nombre de la Clase
atributos
métodos



# Estructura de un programa C#

El método main  
está en la  
clase Program!!

Suma
variable1 : tipo1 variableN : tipoN
metodo1() metodo2() metodoN()

Importar clases adicionales

```
using ...;  
using ...;
```

Carpeta que contiene la clase

```
namespace domApli;
```

Definición de la clase

```
public partial class Suma ...{
```

Declaración de variables  
globales (atributos)

```
    tipo1 variable1;  
    ...  
    tipoN variableN;
```

Los atributos y  
métodos siempre están  
dentro de la clase

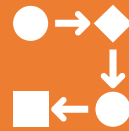
Métodos

```
    ... metodo1( ... ){  
        ...  
    }  
    ... metodo2( ... ){  
        ...  
    }  
    ...  
    ... metodoN( ... ){  
        ...  
    }
```

```
}
```

# Ejecución de un POO

# Secuencia de ejecución de un POO

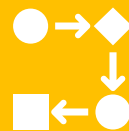


El primer método que se ejecuta es el método main.



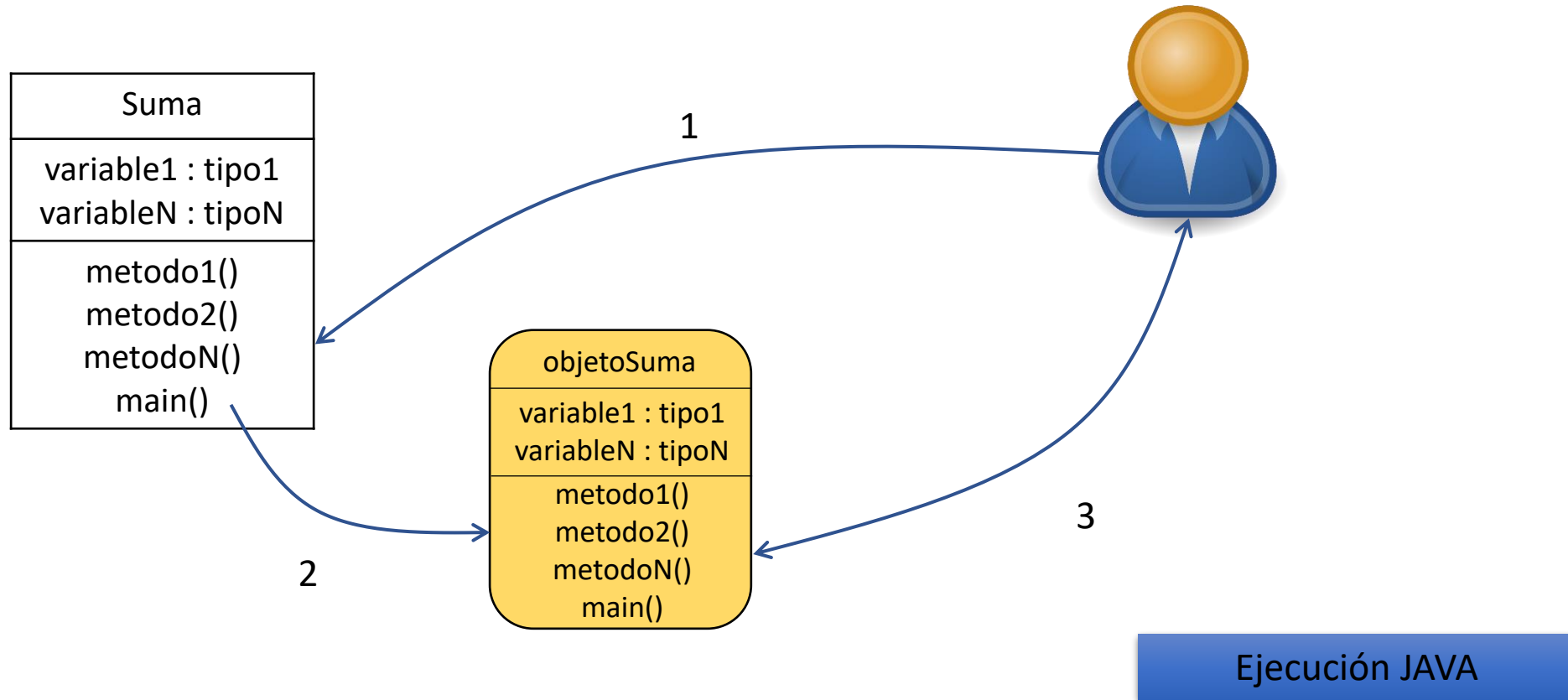
El método main:

- Crea un objeto de la clase
- Ejecuta el método constructor para iniciar los atributos

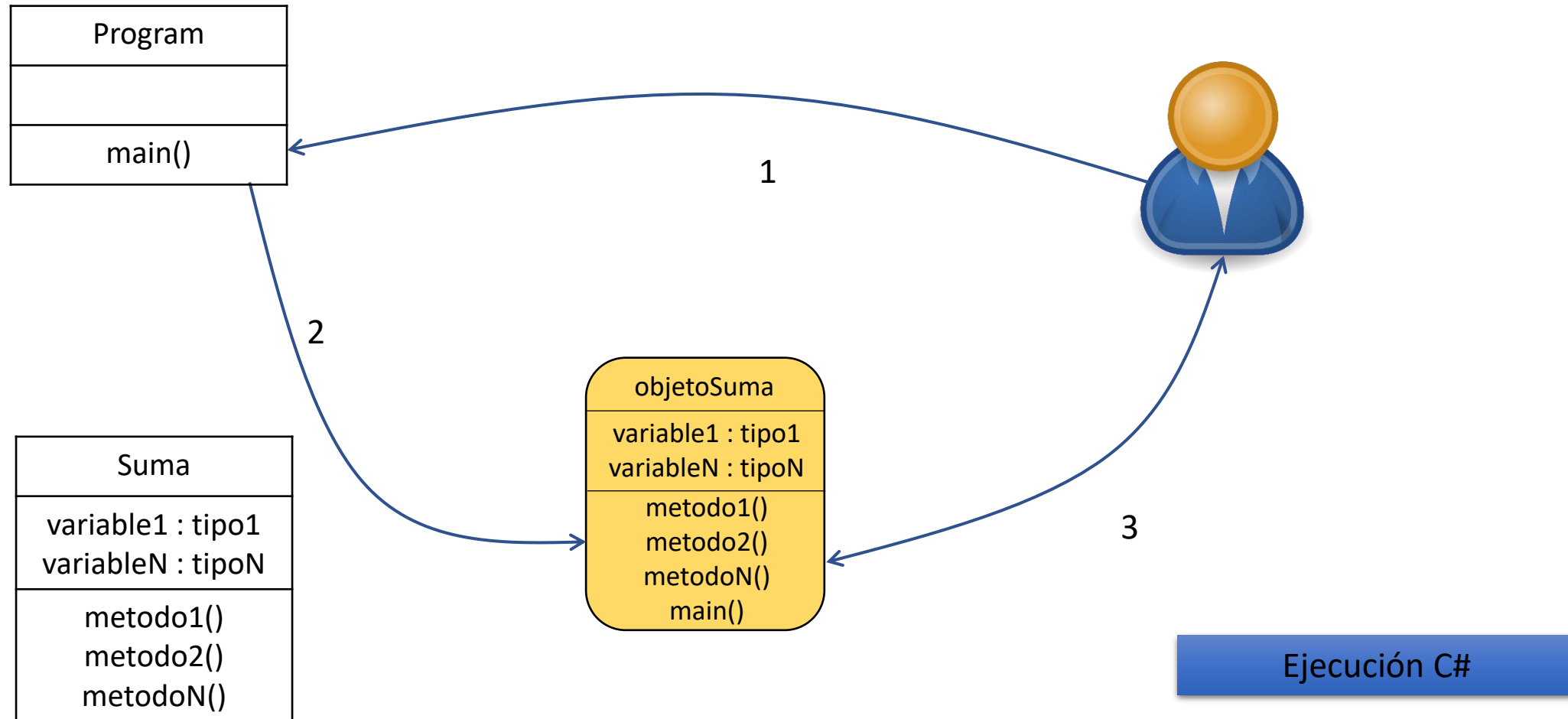


Se ejecutan los otros métodos.

# Por qué se crea un objeto al ejecutar un programa?

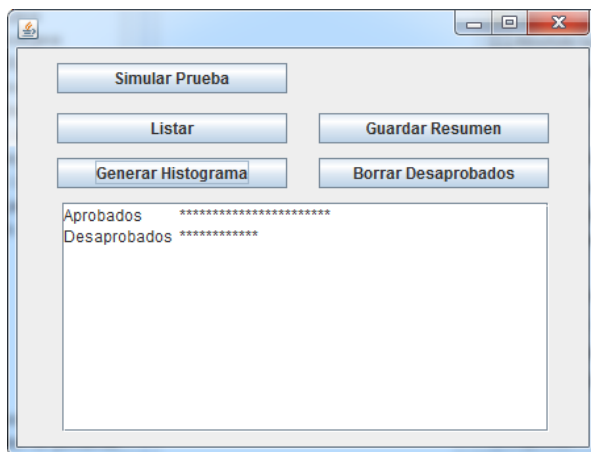


# Por qué se crea un objeto al ejecutar un programa.?





# Enunciados de ejercicios.



Diseño de Pantalla

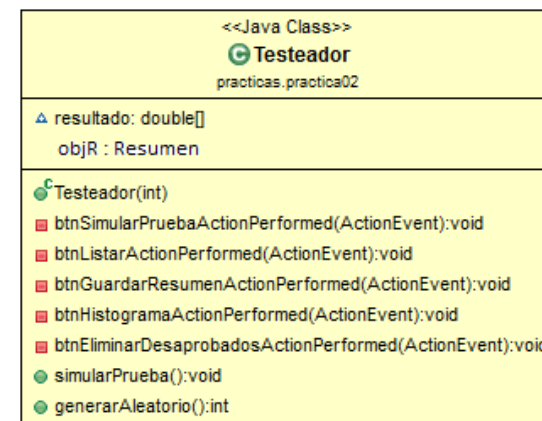
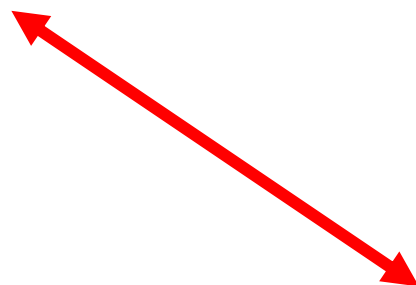
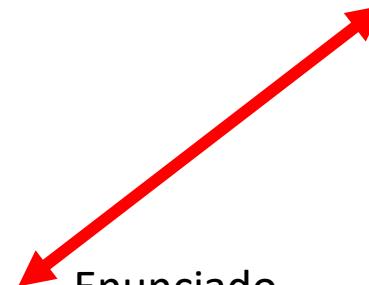


Diagrama de Clase

Componentes del  
Enunciado de un  
ejercicio

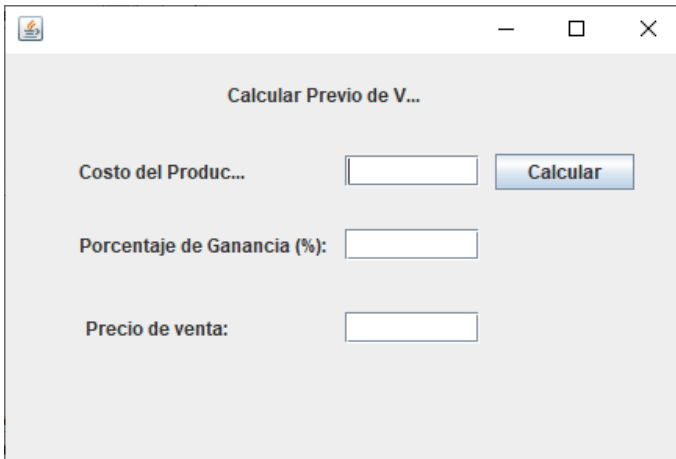


Enunciado



# Ejercicio

Implementa un programa que permita desarrollar la clase siguiente:











Calcular Previo de V...

Costo del Produc...

Porcentaje de Ganancia (%):

Precio de venta:

<<Java Class>>	
 <b>PrecioVenta</b>	
usmp.aedi.semana1	
	contentPane: JPanel
	txtCostoProducto: JTextField
	txtPorcentaje: JTextField
	txtPrecioVenta: JTextField
	<u>main(String[]):void</u>
	PrecioVenta()
	btnCalcularActionPerformed(ActionEvent):void

Crear un programa que permita calcular el precio de venta de un producto X. Se debe ingresar el costo de fabricación del producto y el porcentaje que se desea ganar al venderlo.



GRACIAS.



USMP

FACULTAD DE  
INGENIERÍA Y ARQUITECTURA

Escuela Profesional de  
Ingeniería de Computación y Sistemas

Acreditada por:

