

Programación Orientada a Objetos

Dr. Freddy Alberto Paz Espinoza

fpaz@pucp.edu.pe



Agenda



Definición de POO - Orígenes



Conceptos Básicos

Paradigma

Objeto

Clase

Identificadores



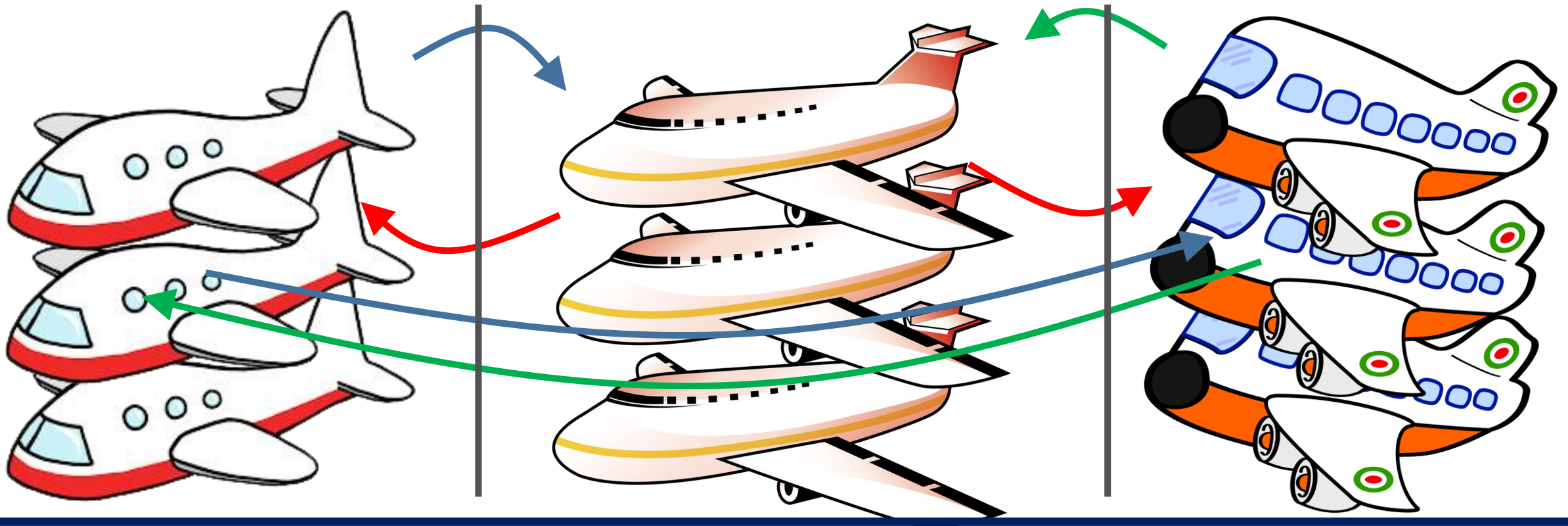
Programación Orientada a Objetos

- Es un **paradigma de programación** basado en el concepto de **objetos**.
- Las características principales de lo que se denomina POO surgen en **1960** y aunque algunos autores difieren en sus orígenes, tienen su inicio en **Simula 67**, un lenguaje de programación diseñado en el centro de cómputo de Oslo¹.

¹Ole Johan Dahl y Kristen Nygaard

Programación Orientada a Objetos

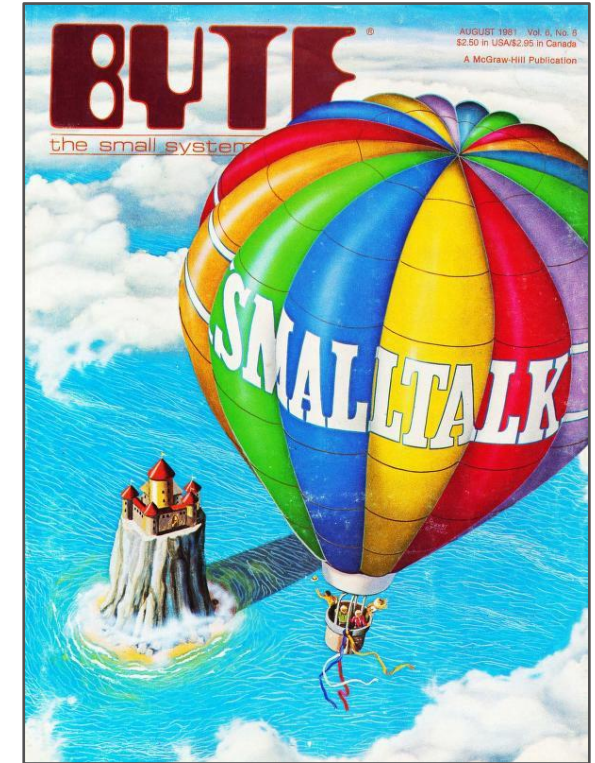
'Ole Johan Dahl y Kristen Nygaard



Programación Orientada a Objetos

Programación Orientada a Objetos

- Posteriormente, en **Agosto de 1981**, se publica en la revista Byte, la descripción del lenguaje de programación **Smalltalk**, el cual refinó algunos de los conceptos originados en el lenguaje **Simula**.
- POO fue una de las propuestas de solución para la “**crisis del software**”.



Programación Orientada a Objetos

- Si bien las técnicas de POO facilitan la creación de complejos sistemas de software por medio de mejores mecanismos de abstracción, **no son la panacea universal de solución.**



Programación Orientada a Objetos

- Programar una computadora sigue siendo una de las tareas más difíciles realizadas por un ser humano.

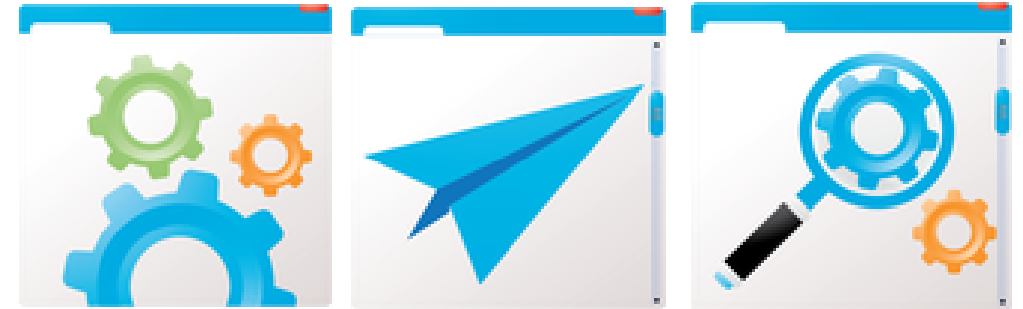
 ***Talento***

 ***Creatividad***

 ***Lógica***

 ***Habilidad para construir y utilizar abstracciones***

 ***Experiencia***



Paradigma

- La palabra **paradigma** irrumpió en el vocabulario moderno a través del influyente libro “**The Structure of Scientific Revolutions**” del historiador de ciencia Thomas Samuel Kuhn (1962).

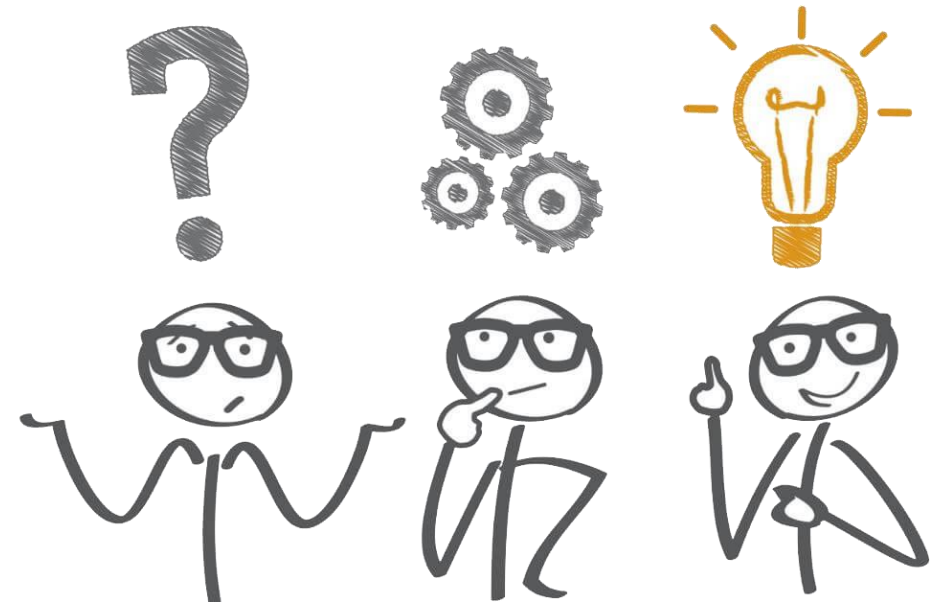
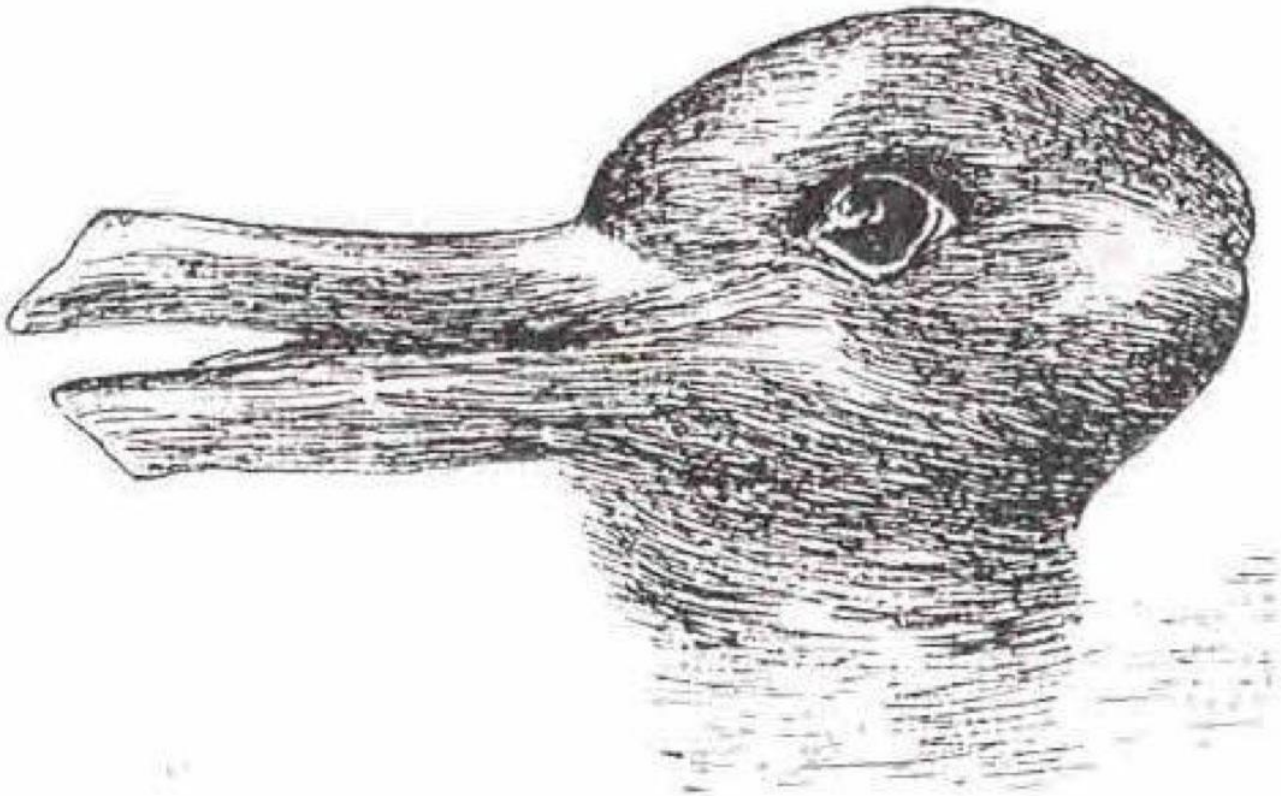
Un modelo para describir un conjunto de teorías, estándares y métodos que agrupados representan una forma de organizar el conocimiento, esto es, una forma de ver el mundo.

Paradigma

- Paradigma de Programación

Un modelo de programación utilizado, el cual está descrito y definido por un conjunto de teorías, estándares y métodos que en conjunto, representan una propuesta de solución por software hacia una problemática determinada.

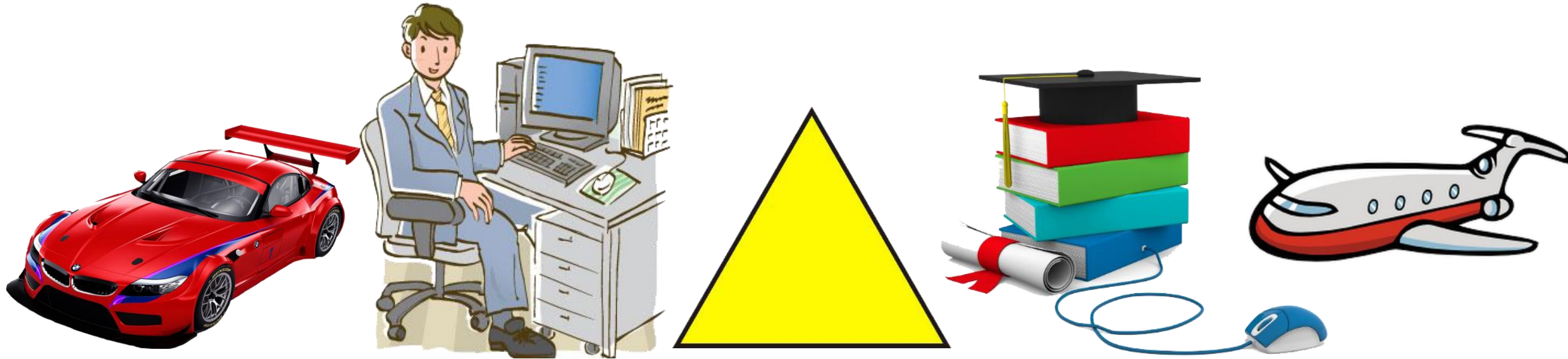
Paradigma



Jastrow, J. (1899). The mind's eye. Popular Science Monthly, 54, 299-312.

Objeto

- Es la representación de un concepto del mundo real (tangible o intangible) dentro de nuestro sistema de software.



Composición de un Objeto

- **Estado.**- Está representado por un conjunto de características del objeto.
- **Comportamiento.**- Está representado por un conjunto de **métodos** u **operaciones** que realiza el objeto.

Estado de un Objeto



Marca: Toyota

Modelo: Yaris

Color: Negro

Año: 2016

Tipo: Automático

Consumo del Motor: 1.5 litros

Comportamiento de un Objeto



Prender el Motor

Apagar el Motor

Acelerar

Frenar

Girar a la Derecha (grados)

Girar a la Izquierda (grados)

Cambiar Velocidad (velocidad)

ACTIVIDAD N° 1




- Defina un **objeto** a partir de algún concepto de mundo real. Describa su estado y comportamiento. Coméntelo en clase.

Mensaje

Un programa se conceptualiza como un conjunto de objetos que se comunican e interactúan entre sí.

Los objetos se comunican unos a otros enviando y recibiendo **mensajes**. 

Es la petición de un objeto a otro para solicitar la ejecución de alguno de sus métodos o para obtener el valor de una de sus características. 

Mensaje

Si un objeto desea que otro objeto realice alguna acción entonces le envía un mensaje. Este mensaje podría contener información adicional en forma de parámetros.

Cuando un objeto recibe un mensaje ejecutará un método u operación.

Estructura de un Mensaje

- 1.** Objeto que recibe el mensaje
- 2.** Método que se debe ejecutar como respuesta
- 3.** Parámetros necesarios del método

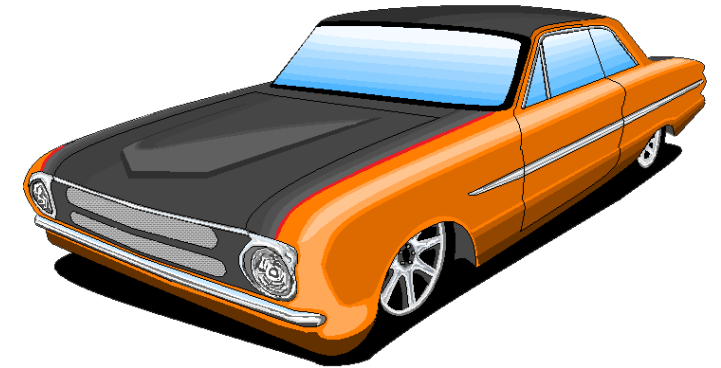
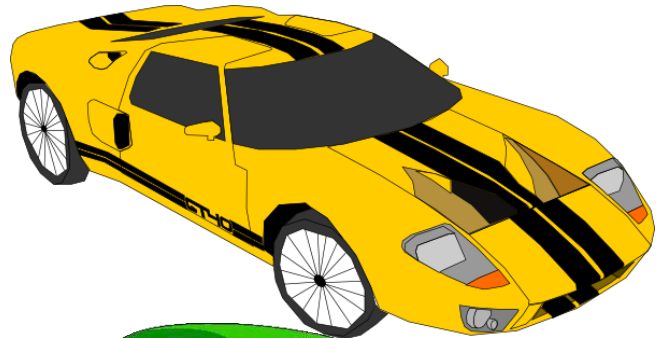
Sintaxis

`<Variable_Objeto>.<Nombre_Método>([<Lista de Parámetros>])`

`miAuto.cambiarVelocidad(100);`

Clase

Representan un tipo particular de objetos con características y comportamiento similar.



OBJETOS DE LA CLASE
AUTO

La tendencia natural del individuo es de clasificar los objetos según sus características comunes.

Una clase es un nivel de abstracción alto que permite describir un conjunto de características y comportamientos comunes de los objetos que representa.

Clase



La clase es una plantilla que describe las características y métodos comunes de un conjunto de objetos.

Un objeto es una instancia de una clase. Cada objeto es miembro de una clase específica y posee el conjunto de características y métodos especificados en la misma.

Clase

La clase está definida por:

📌 El conjunto de características que tienen los objetos de la clase.

📌 Los métodos que pueden ejecutar los objetos de la clase y cómo lo hacen.

Los nombres o identificadores de las clases deben colocarse en singular.

Representación de una Clase en JAVA

```
class <identificador> extends <Clase> {  
    //Propiedades  
    <Modo de Acceso> <Tipo de dato> <Nombre del Atributo>;  
    ...  
    //Métodos  
    Para funciones:  
    <Modo de Acceso> <Tipo de dato> <Nombre> (Lista Parámetros){}  
    Para procedimientos:  
    <Modo de Acceso> void <Nombre> (Lista Parámetros){}  
}
```


Identificador

- Los identificadores deben comenzar con una letra, un carácter de moneda (\$) o un carácter de conexión como el guion bajo (_). Los identificadores no puede comenzar con un número.
- Después del primer caracter, los identificadores pueden contener cualquier combinación de letras, caracteres de moneda, caracteres de conexión o números.
- No existe un límite para el número de caracteres que un identificador puede contener.
- No se pueden usar las palabras reservadas como un identificador.
- Los identificadores son *case-sensitive*.

Bibliografía

- D.J. Barnes y M. Kölling, Programación orientada a objetos con Java. Pearson Educación, 2007
- T. Budd, An introduction to Object-Oriented Programming (Third Edition). Pearson Education, 2001
- E. Gamma, R. Helm, R. Johnson, J. Vlissides, Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley, 1994
- B. Stroustrup, The C++ Programming Language (Third Edition) Addison-Wesley, 1997
- Agustín Froufe. Java 2. Manual de usuario y tutorial. Ed. Ra-Ma
- J. Sánchez, G. Huecas, B. Fernández y P. Moreno, Iniciación y referencia: Java 2. Osborne McGraw-Hill, 2001.
- B. Meyer, Object-Oriented Software Construction (Second Edition). Prentice Hall, 1997.