

CLASES EN JAVA

La clase es la unidad fundamental de programación en Java.

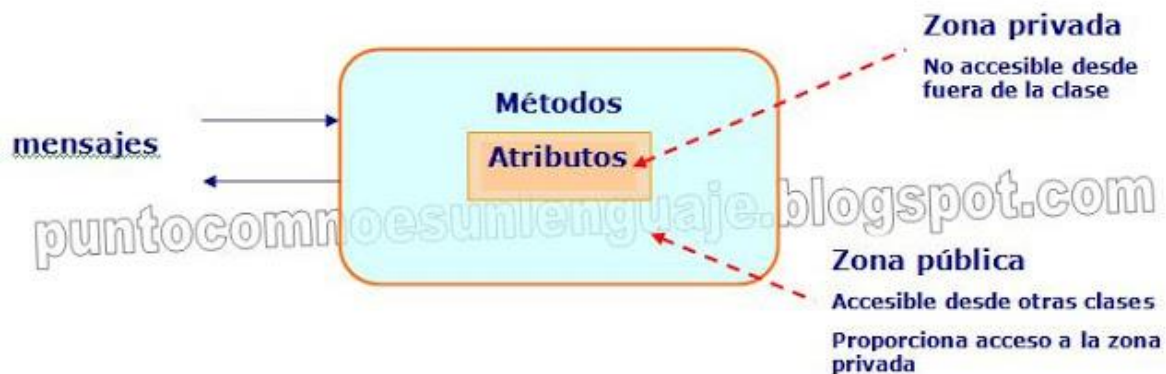
Un programa Java Orientado a Objetos está formado por un conjunto de clases. A partir de esas clases se crearán objetos que interactuarán entre ellos enviándose mensajes para resolver el problema.

Una clase representa al conjunto de objetos que comparten una estructura y un comportamiento comunes.

Puede considerarse como una *plantilla* o *prototipo* de objetos: define los atributos que componen ese tipo de objetos y los métodos que pueden emplearse para trabajar con esos objetos.

Las clases incluyen por tanto atributos y métodos. Los atributos definen el estado de cada objeto de esa clase y los métodos su comportamiento.

Los atributos debemos considerarlos como la zona más interna, oculta a los usuarios del objeto. El acceso a esta zona se realizará a través de los métodos.



La sintaxis general para definir una clase en Java es:

```
[modificadorDeAcceso] class NombreClase [extends NombreSuperClase]
[implements NombreInterface1, NombreInterface2, ... ] {

    //atributos de la clase (0 ó más atributos)
    [modificadorDeAcceso] tipo nombreAtributo;

    //métodos de la clase (0 ó más métodos)
    [modificadorDeAcceso] tipoDevuelto nombreMetodo([lista
parámetros])
    [throws listaExcepciones]{
        // instrucciones del método
    }
}
```

```

        [return valor;]
    }
}

```

Todo lo que aparece entre corchetes es opcional, por lo tanto la definición mínima de una clase en Java es:

```

class NombreClase{
}

```

Como hemos visto antes, el concepto de clase incluye la idea de ocultación de datos, que básicamente consiste en que no se puede acceder a los datos directamente (zona privada), sino que hay que hacerlo a través de los métodos de la clase.

De esta forma se consiguen dos objetivos importantes:

- Que el usuario no tenga acceso directo a la estructura interna de la clase, para no poder generar código basado en la estructura de los datos.
- Si en un momento dado alteramos la estructura de la clase todo el código del usuario no tendrá que ser retocado.

El **modificador de acceso** se utiliza para definir el nivel de ocultación o visibilidad de los miembros de la clase (atributos y métodos) y de la propia clase.

Los modificadores de acceso **ordenados de menor a mayor visibilidad** son:

MODIFICADOR DE ACCESO	EFFECTO	APLICABLE A
private	Restringe la visibilidad al interior de la clase . Un atributo o método definido como private solo puede ser usado en el interior de su propia clase.	Atributos Métodos
<Sin modificador>	Cuando no se especifica un modificador, el elemento adquiere el <i>acceso por defecto o friendly</i> . También se le conoce como acceso de package (paquete). Solo puede ser usado por las clases dentro de su mismo paquete .	Clases Atributos Métodos
protected	Se emplea en la herencia. El elemento puede ser utilizado por cualquier clase dentro de su paquete y por cualquier subclase independientemente del paquete donde se encuentre.	Atributos Métodos
public	Es el nivel máximo de visibilidad. El elemento es visible desde cualquier clase.	Clases Atributos Métodos

(Anonimo, 2013)

<http://puntocomnoesunlenguaje.blogspot.com.co/2012/07/clases-y-objetos-en-java.html>