



Clases en Java

Facultad de Ciencias

Diana Isabel Ramírez García

Introducción a Ciencias de la Computación 2021-1

¿Cómo está definida una clase?

De contener:

- Estructura ----> Atributos
- Comportamiento ----> Métodos

Para defenir a todos los objetos dentro de ella

¿Cómo definimos una clase en Java?

- **Encabezado:**
 - Se conforma por la visibilidad de la clase "public" o "package" y nombre de la clase
- **Cuerpo:**
 - Se conforma por los atributos y métodos

Visibilidad de Atributos y Métodos

Encapsulamiento: Organizamos los elementos de un objeto de manera que estos pueden ser accedidos o son privados.

La **visibilidad** especifica a cuáles atributos o métodos se puede tener acceso desde el cliente que usará el objeto.

Visibilidad de Atributos y Métodos

Palabras reservadas visibilidad:

- public: Puede ser manipulado fuera de la clase
- private: Solo puede ser manipulado en su misma clase
- protected

Calificadores:

- static: Define al atributo como compartido por todos los objetos de la clase.
- final: Define al atributo como constante.

Visibilidad de Atributos y Métodos

	public	private
Atributo	Viola la encapsulación	Se tiene encapsulación
Método	Proporciona servicio a clientes	Auxiliar a otros métodos en la clase

Tabla 4.1 Visibilidad en la clase.

Estructura de Atributos y Métodos

Atributos:

- Tipo
- Identificador
- Opcionalmente un valor inicial.

Métodos:

- Tipo de valor que devuelve
- Nombre
- Parámetros (tipo nombre)

Alcance

- El área dentro del programa en la cual se puede tener acceso a un dato, este es delimitado en java por medio de llaves {}
- Las variables definidas en un bloque son conocidas como **variables locales**, es decir **la variable como su valor** se pierde fuera del bloque. Mientras que las variables de instancia (atributos) su alcance es toda la clase.

¿Qué puede utilizar un método?

- Variables de instancia
- Parámetros y variables locales.
- Datos obtenidos por medio de consola
- Atributos públicos o estáticos de otra clase.

Métodos Constructores

- Su objetivo es asignar un valor inicial a los atributos de tal manera que el objeto tenga un estado válido.
- Tiene el mismo nombre de la clase
- No tiene especificado un valor de retorno.
- Se manda a llamar con el operador *new*

Métodos Constructores

- **Tipos:**

- **Por omisión:** Para definir un estado por default
- **A partir de valores:** Recibe párametros para definir el estado de sus atributos.
- **Copia:** Recibe un objeto de la misma clase y copia su estado.

Métodos Modificadores

Métodos para modificar el valor de atributos privados.

Métodos de Acceso

Métodos para conocer el valor de atributos privados.

Métodos Calculadores

Métodos diseñados para implementar cualquier comportamiento deseado del objeto.

Método Main

```
public static void main(String [] pps) {  
.... //Algoritmo  
}
```

- Es un método público

Método Main

```
public static void main(String [] pps) {  
.... //Algoritmo  
}
```

- Está relacionado con la clase y no con los objetos. Al ser un **método estático** no requiere de un objeto para llamarlo.

Método Main

```
public static void main(String [] pps) {  
.... //Algoritmo  
}
```

- No devuelve ningún valor

Método Main

```
public static void main(String [] pps) {  
.... //Algoritmo  
}
```

- Nombre del método, este puede ser diferente pero por convención se utiliza **main**

Método Main

```
public static void main(String [] pps) {  
.... //Algoritmo  
}
```

- Recibe un parámetro pps, el nombre también puede variar.

Método Equals

Como se vió en la clase String este método es necesario ya el método equals hace una comparación de estado entre los objetos, mientras que "==" compara por referencia.

Método toString

Método para convertir un objeto en una cadena de caracteres.

Bibliografía

- López, A.. (2014). Introducción al desarrollo de programas con Java. México: Las prensas de Ciencias.