PROGRAMACIÓN MODULAR

Es un paradigma de programación que consiste en dividir un problema complejo en otros más pequeños y sencillos llamados módulos o métodos.

Se hace un análisis top-down.

Un módulo es un subprograma que resuelve un problema sencillo y que el conjunto de estos crea un programa.

**ENTRADA**

* Parámetros

**↓**

**PROCESO**

* Función

**↓**

**SALIDA**

* Valor de retorno

VENTAJAS

* **Mejora la legibilidad y manejabilidad**
* **Favorece la localización de errores**
* **Mejora del mantenimiento**
* **Permite la reutilización**
* **Mayor abstracción en la solución**

MÉTODOS EN JAVA

Son bloques de código con un nombre y que opcionalmente pueden:

* **Recibir parámetros**
* **Tener instrucciones**
* **Devolver un valor**

CABECERA

[modificadores] <tipo\_devuelto> nombre ([parámetros])

MODIFICADORES

* **void, public, private, protected**
* **static**
* **native**
* **abstract**
* **final**
* **synchronized**

TIPO DEVUELTO

* **Básico**
* **Clases**
* **void**

CUERPO

Son las instrucciones que puede realizar y que terminan su ejecución cuando llegan a un cierre de llave o una sentencia **return** con o sin parámetros.

SENTENCIA RETURN

Se tiene que devolver el tipo de dato especificado en el método. Puede haber 0 (void) o varias sentencias return (no recomendable).

Devuelven un único valor, a menos que se devuelva un objeto con varios campos.

TIPOS DE MÉTODOS

SEGÚN SU ALCANCE

* **De instancia**
* **Estáticos**

PASO DE PARÁMETROS

POR VALOR

Aquellos métodos que reciban como parámetros tipos de datos básicos (int, float, short, etc) creando así una copia de la variable pasada como parámetro. Los cambios realizados solo afectan a la variable del método.

POR REFERENCIA

Métodos que reciban como parámetros **objetos**. NO se crea un nuevo objeto, sino una nueva referencia a la dirección de memoria del objeto pasado como parámetro. Los cambios efectuados dentro del método también tienen efecto fuera de este.