

UF 5.1

Métodos en Java

```
0010000000001010001101100000010010110001
110001011101000100011111111110100000100
0010100101100001101011101101011011001000
0110110000010101100100010000111000100111
1010011001011010011011010011110111101110
000110100110011001100110100011010
100100110100110011001010001110
10001001int main()0000101111
10101001{00001100011000
1111001100 printf("Hello World");0001100
00100000111 return 42;010101110110
000110100010001101000110100011010
1001001101111010111011110000001010001110
1000100100010101100100111011101000101111
1010100111001101010111000101010100011000
11110011000001101111110101001111110001100
00100000111111101010010010011010101110110
```



**Universidad
Europea de Madrid**

LAUREATE INTERNATIONAL UNIVERSITIES



INTRODUCCIÓN

¿Qué es un método?

Un método en Java es un conjunto de instrucciones definidas dentro de una clase, que realizan una determinada tarea y a las que podemos invocar mediante un nombre.

Ejemplos de métodos utilizados hasta ahora:

- `System.out.println()`
- `in.nextInt()`
- `Math.pow()`



INTRODUCCIÓN

¿Qué es un método?

Cuando se llama a un método, la ejecución del programa pasa al método y cuando éste acaba, la ejecución continúa a partir del punto donde se produjo la llamada.

Utilizando métodos:

- Podemos construir programas modulares.
- Se consigue la reutilización de código. En lugar de escribir el mismo código repetido cuando se necesite, por ejemplo para validar una fecha, se hace una llamada al método que lo realiza.

En Java un método siempre pertenece a una clase.

Todo programa java tiene un método llamado main. Este método es el punto de entrada al programa y también el punto de salida.



CASO DE PRUEBA

Definición

Realiza un programa en java que pidiendo por pantalla los números n y m realice el siguiente cálculo:

$$n! * m! / (n-m)!$$

Es un programa en el que sabiendo realizar el cálculo del factorial no reviste de dificultad, pero en el que tenemos que hacer un mismo fragmento de programa 3 veces.

Si dispusiésemos de un método factorial, con 3 llamadas al mismo método el programa se simplificaría mucho.



CASO DE PRUEBA

Código del main

```
public static void main(String[] args) {  
  
    Scanner in = new Scanner(System.in); int  
    n,m,total;  
    System.out.print("Introduce un valor para n: ");  
    n=in.nextInt();  
    System.out.print("Introduce un valor para m: ");  
    m=in.nextInt();  
    total=factorial(n)*factorial(m)/factorial(n-m);  
    System.out.print("El resultado final es: "+total);  
}
```



La estructura general de un método Java es la siguiente:

```
[especificadores] tipoDevuelto nombreMetodo([lista parámetros])  
[throws listaExcepciones]  
{  
    // instrucciones  
    [return valor;]  
}
```

Los elementos que aparecen entre corchetes son opcionales.



PRIMER EJEMPLO

```
public void mostrarFrase (){  
    System.out.println("Mi primer método en Java");  
}
```

```
public static void main(String[] args) {  
    mostrarFrase();//Llamada al método  
    //la ejecución salta a la implementación del método  
    //Se puede llamar al método todas las veces que se  
    //considere necesario.  
}
```



```
public void mostrarSuma (int a, int b){ int c;  
    c=a+b; System.out.println(a+"+"+b+"="+c);  
}
```

```
public static void main(String[] args) { int x=4,  
    y=2;  
    mostrarSuma(x, y);  
}
```




RETURN

```
public int suma (int a, int b){  
    int c;  
    c=a+b;  
    return c;  
}
```

```
public static void main(String[] args) {  
    int x=4, y=2, z;  
    z=suma(x,y);  
    System.out.println (z);  
}
```