

## Tema 5: Métodos

- Declaración
- Llamada
- Tipos de parámetros
  - Entrada
  - Salida
  - Entrada/Salida

26/10/2017

1

## Método

- **Método:** Conjunto de instrucciones a las que se les da un determinado nombre de tal manera que sea posible ejecutarlas en cualquier momento sin tenerlas que reescribir sino usando sólo su nombre.
- **Cuerpo:** Contiene las instrucciones que se ejecutan cuando se realiza la llamada al método.
- **Nombre:** Identificador para la nomenclatura del método.

26/10/2017

2

## Método

- **Valor de retorno:** valor devuelto por el método como resultado de su llamada. Es único.
- **Parámetros:** Variables adicionales que necesita el método para poder realizar las instrucciones en él indicadas.

26/10/2017

3

## Sintaxis de declaración de un método

```
<tipoDevuelto> <nombreMétodo>(<lista de parámetros>)
{
  <cuerpo>
}
```

Dónde:

- **tipoDevuelto:** Indica el tipo de datos que devuelve el método. Puede ser cualquier tipo de datos básico o no básico (Clase). Si el método no devuelve valor, se indica mediante la palabra reservada **void**.
- **nombreMétodo:** identificador del método. Debe seguir las reglas de los identificadores y ser suficientemente descriptivos de la acción que realizan.
- **listaParámetros:** lista con el formato:
  - **tipoDato para1, tipo Dato para2,.....** Donde se especifica el tipo de dato y el nombre para cada uno de los parámetros del método. Consideraciones:
    - Para más de un parámetro: Se separan por ,
    - No existen parámetros: Es obligatorio indicar ()
- **cuerpo:** contiene las instrucciones que realiza el método.

26/10/2017

4

## Valor de retorno

- Para conseguir que un método devuelva un valor indicado, es necesario incluir la sentencia:  
`return <expresión>;`
- El tipo de datos devuelto por la expresión y el tipo de datos especificado como valor de retorno del método han de ser iguales
- Método de tipo void:
  - Incluir la palabra `return`; sin ninguna expresión.
  - Omitir la palabra `return` y que el método finalice al llegar a la llave de final de método.

26/10/2017

5

## Llamada a métodos

`<objeto>.<nombreMétodo>(<listaDeArgumentos>)`

- objeto: indica la variable de tipo clase en la que se encuentra el método al que se quiere acceder.
- nombreMétodo: identificador del método que se quiere acceder.
- listaDeArgumentos: Lista de parámetros necesarios para el método.

26/10/2017

6

## Tipos de parámetros

- De entrada: Se copia un valor del parámetro y su valor no se ve afectado en el método.

```
static void sumar(int n1, int n2, ref int res)
{
    res = n1 + n2;
}
```

26/10/2017

7

## Tipos de parámetros

- De entrada/salida (Por referencia): Su valor se puede modificar dentro del método. Se emplea la palabra reservada `ref`. (Método y llamada)

```
static void sumar(int n1, int n2, ref int res)
{
    res = n1 + n2;
}
```

26/10/2017

8

## Tipos de parámetros

- De salida: (Por referencia): Es **obligatorio** modificar el valor del parámetro de salida, además el valor anterior del parámetro no puede ser utilizado dentro del método. Se emplea la palabra reservada *out* (Método y llamada)

```
static void sumar(int n1, int n2, out int res)
{
    res = n1 + n2;
}
```