

Programación Avanzada

Métodos



Métodos

- **Bloque de código que realiza una tarea específica y que sólo se ejecuta al ser invocado o llamado**
 - **tienen un nombre**
 - **pueden recibir parámetros**
 - **pueden devolver un valor**

Métodos

□ Se utilizan para:

- dividir un problema en problemas más pequeños (divide y vencerás)
- reutilizar el código
- aclarar o autodocumentar el código
- facilitar el mantenimiento

Métodos

□ **Declaración:**
public class MiClase{

...

public static void miMétodo(){

...

}

...

}

Métodos

□ **Declaración:**
public class MiClase{

...

public static void miMétodo(){

...

}

Visibilidad pública, puede invocarse dentro y fuera de la clase donde está declarada

...

}

Métodos

□ **Declaración:**
public class MiClase{

...

public static void miMétodo(){

...

}

...

}

Los métodos estáticos pueden invocarse sin crear o instanciar un objeto de la clase donde están declarados

Métodos

□ **Declaración:**
public class MiClase{

...

public static void miMétodo(){

...

}

...

}

Tipo de dato que retorna el método. En este caso nada o vacío

Métodos

□ Declaración:
public class MiClase{

...

public static void miMétodo(){

...

}

**Nombre del
método**

...

}

Métodos

□ **Declaración:**
public class MiClase{

...

public static void miMétodo(){

...

}

**Sin
parámetros**



...

}

Métodos

□ Invocación o llamada:

```
public class MiClase{  
    public static void main(String[] args){  
  
        ...  
        miMétodo();  
    }  
  
    ...  
}
```

Métodos

□ Invocación o llamada:

- Java ejecuta la o las sentencias indicadas en el método
- Al finalizar, continúa la ejecución de la siguiente sentencia en el contexto dónde se invocó a el método

Métodos - Ejemplo

□ Declaración:
`public class MiClase{`

`...`

```
    public static void saludo( ){  
        System.out.println("Hola");
```

```
    }
```

```
    ...
```

```
}
```

Métodos - Ejemplo

- Invocación o llamada:

```
public class MiClase{  
    public static void main(String[] args){  
        ...  
        saludo();  
    }  
    ...  
}
```

L Hola

Métodos - Ejemplo

□ Declaración:

```
public class MiClase{
```

```
...
```

```
    public static void syso(String texto){  
        System.out.println(texto);
```

```
    }
```

```
    ...
```

```
}
```

Métodos - Ejemplo

□ Declaración:

```
public class MiClase{
```

```
...
```

```
    public static void syso(String texto){  
        System.out.println(texto);
```

```
    }
```

```
    ...
```

```
}
```

parámetro

Métodos - Ejemplo

- Invocación o llamada:

```
public class MiClase{  
    public static void main(String[] args){  
        ...  
        syso("Hola Mundo Java!");  
    }  
    L  HoLa Mundo Java!  
    ...  
}
```

Métodos - Ejemplo

□ Declaración:

```
public static int mayor(int a, int b){  
    int elMayor = 0;  
    if (a > b){  
        elMayor = a;  
    }else{  
        elMayor = b;  
    }  
    return elMayor;  
}
```

Métodos - Ejemplo

□ Declaración:

```
public static int mayor(int a, int b){  
    int elMayor = 0;  
    if (a > b){  
        elMayor = a;  
    }else{  
        elMayor = b;  
    }  
    return elMayor;  
}
```

Retorna un
valor entero

Métodos - Ejemplo

□ Declaración:

```
public static int mayor(int a, int b){  
    int elMayor = 0;  
    if (a > b){  
        elMayor = a;  
    }else{  
        elMayor = b;  
    }  
    return elMayor;  
}
```

Indica el valor
que retorna el
método

Métodos - Ejemplo

□ Declaración:

```
public static int mayor(int a, int b){  
    int elMayor = 0;  
    if (a > b){  
        elMayor = a;  
    }else{  
        elMayor = b;  
    }  
    return elMayor;  
}
```

Dos parámetros
enteros a y b

Métodos - Ejemplo

- Invocación o llamada:

```
public class MiClase{  
    public static void main(String[] args){  
        syso("El mayor es " + mayor(33, 25));  
    }  
    L El mayor es 33  
    ...  
}
```