AulaVirtual

4.D. Estructuras de salto.

✔ Hecho

1. Estructuras de salto.

1.3. Sentencia Return.

Ya sabemos cómo modificar la ejecución de bucles y estructuras condicionales múltiples, pero ¿Podríamos modificar la ejecución de un método? ¿Es posible hacer que éstos detengan su ejecución antes de que finalice el código asociado a ellos? Sí es posible, a través de la sentencia `return` podremos conseguirlo.

La sentencia `return` puede utilizarse de dos formas:

- Para terminar la ejecución del método donde esté escrita, con lo que transferirá el control al punto desde el que se hizo la llamada al método, continuando el programa por la sentencia inmediatamente posterior.

- Para devolver o retornar un valor, siempre que junto a `return` se incluya una expresión de un tipo determinado. Por tanto, en el lugar donde se invocó al método se obtendrá el valor resultante de la evaluación de la expresión que acompañaba al método.

En general, una sentencia `return` suele aparecer al final de un método, de este modo el método tendrá una entrada y una salida. También es posible utilizar una sentencia `return` en cualquier punto de un método, con lo que éste finalizará en el lugar donde se encuentre dicho `return`. No será recomendable incluir más de un `return` en un método y por regla general, deberá ir al final del método, como hemos comentado.

El valor de retorno es opcional, si lo hubiera debería de ser del mismo tipo o de un tipo compatible al tipo del valor de retorno definido en la cabecera del método, pudiendo ser desde un entero a un objeto creado por nosotros. Si no lo tuviera, el tipo de retorno sería `void`, y `return` serviría para salir del método sin necesidad de llegar a ejecutar todas las instrucciones que se encuentran después del `return`.

Para saber más

En el siguiente programa java encontrarás el código de un programa que obtiene la suma de dos números, empleando para ello un método sencillo que retorna el valor de la suma de los números que se le han pasado como parámetros. Presta atención a los comentarios y fíjate en las conversiones a entero de la entrada de los operandos por consola.

```
import java.io.*;
/**
 *
 * Uso de return en un método
 */
public class sentencia_return {

    private static BufferedReader stdin = new BufferedReader( new InputStreamReader(System.in));

    public static int suma(int numero1, int numero2)
    {
        int resultado;
        resultado = numero1 + numero2;
        return resultado; //Mediante return devolvemos el resultado de la suma
    }
```

```
    public static void main(String[] args) throws IOException {
        //Declaración de variables
        String input; //Esta variable recibirá la entrada de teclado
        int primer_numero, segundo_numero; //Estas variables almacenarán los operandos

        // Solicitamos que el usuario introduzca dos números por consola
        System.out.print ("Introduce el primer operando:");
        input = stdin.readLine(); //Leemos la entrada como cadena de caracteres
        primer_numero = Integer.parseInt(input); //Transformamos a entero lo introducido
        System.out.print ("Introduce el segundo operando: ");
        input = stdin.readLine(); //Leemos la entrada como cadena de caracteres
        segundo_numero = Integer.parseInt(input);  //Transformamos a entero lo introducido

        //Imprimimos los números introducidos
        System.out.println ("Los operandos son: " + primer_numero + " y " + segundo_numero);
        System.out.println ("obteniendo su suma... ");

        //Invocamos al método que realiza la suma, pasándole los parámetros adecuados
        System.out.println ("La suma de ambos operandos es: " + suma(primer_numero,segundo_numero));
    }
}
```

Autoevaluación

¿Qué afirmación es correcta?

- Con return, se puede finalizar la ejecución del método en el que se encuentre.
- Con return, siempre se retornará un valor del mismo tipo o de un tipo compatible al definido en la cabecera del método.
- Con return, puede retornarse un valor de un determinado tipo y suele hacerse al final del método. Además, el resto de respuestas también son correctas.

◄ 4.C. Estructuras de repetición.

Ir a...

4.E. Actividades propuestas. ►