



APRENDERAPROGRAMAR.COM

EJEMPLO MÉTODO LENGTH
DE LA CLASE STRING DEL
API DE JAVA. USAR
MÉTODOS PARA EVITAR
ERRORES. (CU00650B)

Sección: Cursos

Categoría: Curso “Aprender programación Java desde cero”

Fecha revisión: 2029

Resumen: Entrega nº50 curso Aprender programación Java desde cero.

Autor: Alex Rodríguez

USAR MÉTODOS PARA EVITAR ERRORES. EJEMPLO MÉTODO LENGTH DE LA CLASE STRING

Vamos a usar el método *length* de la clase String para tratar de evitar que nos salten errores al emplear el método *substring*. Lo primero que haremos será consultar la documentación de la clase, la signatura del método, y, si es necesario, su detalle. La signatura del método es:



<u>int</u>	<u>length</u> ()	Devuelve el número de caracteres del String.
------------	-------------------	--

El tipo devuelto por el método es *int* (un entero) y el método no requiere parámetros para ser ejecutado. El método no está sobrecargado, ya que hay una única manera de invocarlo. Un ejemplo de invocación puede ser *NumCaracteres = micadena.length()*. Recordar que **siempre que invoquemos un método hemos de incluir los paréntesis**, independientemente de que sea necesario pasar parámetros o no.

Valiéndonos de este método, intenta escribir tu propio código para hacer lo siguiente:

- a) Reescribir la clase *Combinador* que usamos anteriormente de forma que el método *combinacion* devuelva "No aporta cadenas válidas" si se le pasa como parámetro alguna cadena que contenga menos de tres caracteres.
- b) Escribir un nuevo método de la clase, al que podemos denominar *combinadoSiempre* que en caso de que se pasen cadenas con menos de tres caracteres, las combine de todas formas en base a los caracteres disponibles, sea el número que sea.

El código podría ser el siguiente:

```
// Definición de una clase de ejemplo con uso del método length sobre objetos String aprenderaprogramar.com
// Combina las tres primeras letras de dos textos introducidos por el usuario en una sola cadena separada por un espacio
public class Combinador {

    private String texto1;
    private String texto2;
    public Combinador () { texto1 = ""; texto2 = ""; }

    //Método que combina las tres primeras letras de cada cadena
    String Combinacion (String valor_texto1, String valor_texto2) {
        if (valor_texto1.length() >= 3 && valor_texto2.length() >= 3) { //Comprobación
            String combinacion = valor_texto1.substring (0, 3) + " " + valor_texto2.substring (0, 3);
            return combinacion;
        } else { return "No aporta cadenas válidas"; }
    } //Cierre del método
}
```

```
//Método que combina las cadenas aunque contengan menos de 3 caracteres (nuevo método)
String CombinadoSiempre (String valor_texto1, String valor_texto2) {
    int longitud_texto1 = 3; //Si podemos extraeremos tres caracteres
    int longitud_texto2 = 3;
    String CombinadoSiempre = ""; //Cadena vacía, variable local
    if (valor_texto1.length() < 3) { //Si hay menos de 3 caracteres extraemos los que haya
        longitud_texto1 = valor_texto1.length();
    }
    if (valor_texto2.length() < 3) { longitud_texto2 = valor_texto2.length(); }
    CombinadoSiempre = valor_texto1.substring (0,longitud_texto1)+" "+ valor_texto2.substring (0, longitud_texto2);
    return CombinadoSiempre;
} //Cierre del método

} //Cierre de la clase
```

Recuerda que **los espacios cuentan**. No es lo mismo una cadena de longitud cero o cadena vacía, representada por dos comillas sin espacio entre ellas, que una cadena que contenga un espacio, representada por dos cadenas que contengan un espacio entre ellas, cuya longitud o número de caracteres es 1.

EJERCICIO

Crea un objeto de tipo Combinador y prueba a ejecutar los dos métodos disponibles pasando distintos parámetros: cadenas con más de tres caracteres, una cadena con más de tres y otra con menos de tres y ambas con menos de tres caracteres. Fíjate en el resultado que ofrece el método CombinadoSiempre cuando se le pasan como parámetros dos cadenas vacías. ¿Te parece lógico el resultado?

En la clase tenemos definidos dos atributos o campos. ¿Tienen utilidad conocida a la vista del código de la clase? ¿Se usan en los métodos que tenemos definidos?

Puedes comprobar si es correcta tu respuesta consultando en los foros aprenderaprogramar.com.

Próxima entrega: CU00651B

Acceso al curso completo en aprenderaprogramar.com --> Cursos, o en la dirección siguiente:

http://www.aprenderaprogramar.com/index.php?option=com_content&view=category&id=68&Itemid=188