

```
1 namespace ResumenFunciones
2 {
3     public class StringFunctions
4     {
5         public static string StringFunction1(string a, string b)
6         {
7             return a + b;
8         }
9
10        public static string StringFunction2(int number)
11        {
12            string result = "0";
13
14            for (int i = 0; i < number; i++)
15            {
16                result += "," + (i + 1);
17            }
18            return result;
19        }
20
21        public static string StringFunction3(int number)
22        {
23            string result = "1";
24            int aux = 1;
25
26            for (int i = 0; i < number; i++)
27            {
28                aux *= 2;
29                result += "," + aux;
30            }
31            return result;
32        }
33
34        public static void Concatenate(string text1, string text2,
35                                     string text3)
36        {
37            Console.WriteLine("Frase1 , 2 y 3");
38            Console.WriteLine(text1 + ", " + text2 + ", " + text3);
39
40            Console.WriteLine("Frase1 y 2");
41            Console.WriteLine("{0} , {1}", text1, text2);
42
43            Console.WriteLine("Frase2 y 3");
44            Console.WriteLine($"{text2} , {text3}");
45
46            string textResult = String.Format("Frase2 y 3", text2,
47                                             text3);
48            Console.WriteLine(textResult);
49
50            public static void SingularText(string text1)
51            {
52                Console.WriteLine(@"Todo este texto es literal / & + text1. ➤
```

```
        Se consigue con @");
52     Console.WriteLine("El texto que inicia con \"\$\" permite  ➤
        introducir variables dentro del formato");
53     Console.WriteLine("Este texto escapa \"con barras  ➤
        inclinadas\" (\\)\");
54     Console.WriteLine("El parametro \"\\r\" sirve para mover el  ➤
        cursor a la izquierda del todo");
55     Console.WriteLine("El parametro \"\\n\" sirve para cambiar  ➤
        de linea");
56     Console.WriteLine("El parametro \"\\b\" crea un espacio");
57     Console.WriteLine("El parametro \"\\t\" crea un tabulado");
58 }
59
60 public static void StringReplace(string text, string word,  ➤
        string replace)
61 {
62     Console.WriteLine("El metodo \"string.Replace(texto,  ➤
        alternative)\" reemplaza el 'texto' por un  ➤
        'alternative'");
63     string result = text.Replace(word, replace);
64     Console.WriteLine(result);
65 }
66
67 public static void StringIndexOf(string text)
68 {
69     Console.WriteLine("El metodo \"string.IndexOf(text)\" nos  ➤
        indica la posicion del string 'text'");
70     int result = text.IndexOf("=");
71     Console.WriteLine(text);
72     Console.WriteLine("El = ocupa la posicion numero: {0}",  ➤
        result);
73 }
74
75 public static void StringSubstring(string text, int index1, int  ➤
        index2)
76 {
77     Console.WriteLine("El metodo \"string.Substring(index1,  ➤
        index2)\" devuelve el texto que se encuentra entre index1  ➤
        y index2");
78     Console.WriteLine(text);
79     Console.WriteLine(text.Substring(index1, index2));
80     Console.WriteLine(text.Substring(index1));
81 }
82
83 public static void StringRemove(string text)
84 {
85     Console.WriteLine(text);
86
87     string result = text.Remove(5);
88     Console.WriteLine(result);
89
90     result = text.Remove(5, 7);
91     Console.WriteLine(result);
```

```
92     }
93
94     public static void StringChangeCase(string text)
95     {
96         Console.WriteLine(text);
97         Console.WriteLine("Puedes devolver todo el texto en      ↗
98         minúsculas: " + text.ToLower());
99         Console.WriteLine("Puedes devolver todo el texto en      ↗
100         mayúsculas: " + text.ToUpper());
101     }
102
103     public static void StringTrim(string text)
104     {
105         Console.WriteLine("La funcion \"text.Trim()\" elimina los  ↗
106         espacios en blanco antes y despues de los caracteres");
107     }
108 }
```