

```

static void Main(string[] args)
{
    Console.WriteLine("I am " + 18 + ".");
    Console.WriteLine("I am " + 18.ToString() + ".");
    Console.WriteLine($"I am {18}.");
}

```

```

// [9 13 - 9 51]
IL_0001: ldstr      "I am "
IL_0006: ldc.i4.s    18 // 0x12
IL_0008: stloc.0      // V_0
IL_0009: ldloc.s     V_0
IL_000b: call         instance string [System.Runtime]System.Int32::ToString()
IL_0010: ldstr      "."
IL_0015: call         string [System.Runtime]System.String::Concat(string, string, string)
IL_001a: call         void [System.Console]System.Console::WriteLine(string)
IL_001f: nop

// [10 13 - 10 62]
IL_0020: ldstr      "I am "
IL_0025: ldc.i4.s    18 // 0x12
IL_0027: stloc.0      // V_0
IL_0028: ldloc.s     V_0
IL_002a: call         instance string [System.Runtime]System.Int32::ToString()
IL_002f: ldstr      "."
IL_0034: call         string [System.Runtime]System.String::Concat(string, string, string)
IL_0039: call         void [System.Console]System.Console::WriteLine(string)
IL_003e: nop

// [11 13 - 11 46]
IL_003f: ldstr      "I am {0}."
IL_0044: ldc.i4.s    18 // 0x12
IL_0046: box         [System.Runtime]System.Int32
IL_004b: call         string [System.Runtime]System.String::Format(string, object)
IL_0050: call         void [System.Console]System.Console::WriteLine(string)
IL_0055: nop

```

Как мы видим, упаковка происходит только при интерполяции строки(\$).

При этом возникает ещё один минус для производительности, а именно `String.Format()` работает медленнее, чем `String.Concat()`.

Т.е. интерполяция, хоть и добавляет удобочитаемости, но с другой стороны, ведёт к потере производительности, что незаметно на нескольких строках, но если необходимо много миллионов строк – может повлиять).