

## Сравнение методов .Parse() и .TryParse()

Метод .Parse() приводит содержимое строки к нужному типу, если приведение невозможно, выбрасывает исключение и прерывает выполнение программы.

```
string txtExample = "5";
intExample = Int32.Parse(txtExample);
Console.WriteLine(intExample);
chek = Int32.TryParse(txtExample, out intExample);
Console.WriteLine(intExample);
```

```
5 13 - 15 50]
08: ldloc.s      txtExample
0a: call         int32 [System.Runtime]System.Int32::Parse(string)
0f: stloc.0       // intExample
```

Как видно из скриншота, сперва содержимое строки загружается в стек, потом выполняется метод, в котором содержимое строки конвертируется в тип Int32. Если конвертирование прошло успешно, следующим шагом результат конвертирования присваивается переменной типа Int32, если конвертирование прошло не успешно, то получим System.FormatException и выполнение программы прервется.

Метод .TryParse пытается привести содержимое строки к нужному типу и возвращает результат этой попытки в виде булевского значения : если приведение удалось - True (при этом результат конвертирования записывается в нужную переменную через адрес), если приведение не удалось –возвращает False (в переменную ничего не передается).

```
string txtExample = "5";
intExample = Int32.Parse(txtExample);
Console.WriteLine(intExample);
chek = Int32.TryParse(txtExample, out intExample);
Console.WriteLine(intExample);
```

```
13 - 17 63]
7: ldloc.s      txtExample
9: ldloc.s      intExample
b: call        bool [System.Runtime]System.Int32::TryParse(string, int32&)
0: stloc.1     // chek
```

Как видно из скриншота: сперва содержимое строки загружается в стек, потом загружается адрес переменной, в которую, в случае успеха, нужно записать конвертированное значение. Затем выполняется метод, в котором содержимое строки пытается сконвертироваться в тип Int32 и записаться в переменную через ее адрес, потом идет запись результата этой попытки: если удалось – True, иначе – False.

Метод .Parse() рекомендуется использовать только в том случае, если разработчик уверен, что

приведение пройдет успешно, либо если потом используется обработка исключений. В остальных случаях рекомендуется использовать метод .TryParse(), т.к. он не прерывает выполнение программы в случае невозможности привести содержимое строки к нужному типу.