

Контест 10

11 янв 2022, 18:23:16

старт: 16 мар 2021, 12:30:00

финиш: 22 мар 2021, 23:59:59

длительность: 6д. 11ч.

начало: 16 мар 2021, 12:30:00

конец: 22 мар 2021, 23:59:59

G. Singleton for Everyone

Ограничение времени	2 секунды
Ограничение памяти	64Mb
Ввод	стандартный ввод
Вывод	стандартный вывод

Сегодня на Васю снизошло озарение, он заметил вопиющую несправедливость: для того, чтобы использовать паттерн "Singleton" тип должен изначально разрабатываться, согласно этому паттерну. Но иногда, разработчики типа даже не могут представить, что их тип будет удобнее использовать именно как синглтон.

Вася понял, что для того, чтобы исправить эту ситуацию необходимо написать свой обобщённый класс, который позволит использовать объект любого типа (в котором определён конструктор без параметров) как синглтон, то есть, объект будет создан при первом обращении и всегда будет возвращаться один и тот же объект.

Для этого он предлагает разработать обобщённый класс `Singleton<T>` со следующими членами:

- закрытое статическое поле типа соответствующего типизирующему параметру.
- открытое статическое свойство типа соответствующего типизирующему параметру, возвращающее один и тот же объект в `get`-аксессоре и выбрасывающее исключение типа `NotSupportedException` с текстом "This operation is not supported" при попытке присваивания свойству значения.

Работа с входными и выходными данными реализована в коде шаблона и направлена на проверку корректности реализации обобщённого класса `Singleton`.

Пример

Ввод <input type="text"/>	Вывод <input type="text"/>
add int 25	25 13
add int 13	2
add str 2	
print int	
print str	

Примечания

В систему необходимо сдавать только файл `Singleton.cs`.

Обратите внимание, что описанный обобщённый тип будет работать корректно только со ссылочными типами, поэтому настоятельно рекомендуется ограничить типизирующий параметр только ссылочными типами.

Реализация `Singleton<T>` не должна быть безопасной при работе в многопоточном окружении (Вася еще не прошел многопоточность).

Набрать здесь

Отправить файл

```
1 using System;
2
3 class Singleton<T> where T : class, new()
4 {
5     private static T t;
6     public static T Instance
7     {
8         get
9         {
10             if (t == null) t = new T();
11
12             return t;
13         }
14         set
15         {
16             throw new NotSupportedException("This operation is not supported");
17         }
18     }
19 }
20
```

ОтправитьПредыдущаяСледующая