The variable 'MyException' is declared but never used

Ask Question

I need to clear this warning :

63

```
try
{
    doSomething()
}
catch (AmbiguousMatchException MyException)
{
    doSomethingElse()
}
```

☆

7

The complier is telling me :

> The variable 'MyException' is declared but never used

How can I fix this.

c#   .net   exception   frameworks   ambiguity

edited Aug 29 '17 at 6:47

Gilad Green
30.5k   5   33   59

asked Jun 23 '11 at 14:26

Wassim AZIRAR

6 Answers

1. You can remove it like this:

124

```
try
{
    doSomething()
}
catch (AmbiguousMatchException)
{
    doSomethingElse()
}
```

2. Use warning disable like this:

```
try
{
    doSomething()
}
#pragma warning disable 0168
catch (AmbiguousMatchException exception)
#pragma warning restore 0168
{
    doSomethingElse()
}
```

Other familiar warning disable

```
#pragma warning disable 0168 // variable declared but not used.
#pragma warning disable 0219 // variable assigned but not used.
#pragma warning disable 0414 // private field assigned but not used.
```

edited Aug 29 '17 at 6:47

Gilad Green
30.5k　5　33　59

answered Jun 23 '11 at 14:27

Jalal Said
13.5k   5   35   62

---

4   A list of all compiler errors and warnings are available at the MSDN – Jalal Said Jun 23 '11 at 16:38

The solution given by @fparadis2 is better since not advisable to supress warnings when we can fix it – Ram May 17 '13 at 12:34

@dasariramacharanprasad that was my first suggestion, re-read my answer ;) – Jalal Said May 20 '13 at 11:21

1   +1 @Jalal for alternate options. yes you are right! – Ram May 22 '13 at 7:06

solution #2 is not a solution, it is a mask. – Gusdor Mar 3 '14 at 10:19

▲

0

▼

`but never used` means that you should use it after catch() such as writing its value to console, then this warning message will disappear.

```
catch (AmbiguousMatchException MyException)
{
    Console.WriteLine(MyException); // use it here
}
```

answered Oct 8 '18 at 18:42

Ray Chakrit
17   4

▲

You could write the exception out to a log if you've got one running. Might be useful for tracking down any problems.

2

```
Log.Write("AmbiguousMatchException: {0}", MyException.Message);
```

answered Jun 23 '11 at 15:04

Neil
3,111 5 40 76

where will the error message be stored? – Dylan Czenski Jun 2 '16 at 21:19

2

The trouble is, you aren't using your variable `MyException` anywhere. It gets declared, but isn't used. This isn't a problem... just the compiler giving you a hint in case you intended to use it.

answered Jun 23 '11 at 14:28

Brad
117k 29 239 398

You can simply write:

21

```
catch (AmbiguousMatchException)
```

and omit the exception name if you won't be using it in the catch clause.

answered Jun 23 '11 at 14:28

fparadis2
733 4 11

+1 since it resolves the warning! not explained in MSDN. – Ram May 17 '13 at 12:33 ✏

You declare a name for the exception, MyException, but you never do anything with it. Since it's not used, the compiler points it out.

34

You can simply remove the name.

```
catch(AmbiguousMatchException)
{
    doSomethingElse();
}
```

answered Jun 23 '11 at 14:27

Khepri
8,229   4   37   58