**[Type Checking: typeof, GetType, or is?](http://stackoverflow.com/questions/983030/type-checking-typeof-gettype-or-is)**

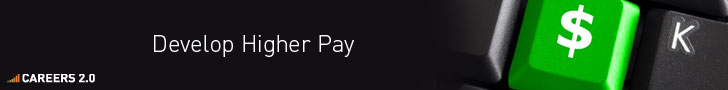
[](http://engine.adzerk.net/r?e=eyJhdiI6NDE0LCJhdCI6NCwiY20iOjI5MTEsImNoIjoxMTc4LCJjciI6NTkyMSwiZGkiOiI2NDMwZjFjODBjOGQ0YTZmYWNkYTU3ZWIzNzM1NWE5OSIsImRtIjoxLCJmYyI6MTAxODYsImZsIjo3NDcyLCJrdyI6ImMjIiwibnciOjIyLCJyZiI6Imh0dHBzOi8vd3d3Lmdvb2dsZS5sdS8iLCJydiI6MCwicHIiOjE2MDQsInN0Ijo4Mjc3LCJ6biI6NDMsInVyIjoiaHR0cDovL2NhcmVlcnMuc3RhY2tvdmVyZmxvdy5jb20vIn0&s=LSGGBgtdkEwgwAIGi7RUmSCJQp4)

|  |  |  |  |
| --- | --- | --- | --- |
| up vote197down vote[favorite](http://stackoverflow.com/questions/983030/type-checking-typeof-gettype-or-is)  **90** | I've seen many people use the following code:  Type t = typeof(obj1);  if (t == typeof(int))  // Some code here  But I know you could also do this:  if (obj1.GetType() == typeof(int))  // Some code here  Or this:  if (obj1 is int)  // Some code here  Personally, I feel the last one is the cleanest, but is there something I'm missing? Which one is the best to use, or is it personal preference?  [c#](http://stackoverflow.com/questions/tagged/c%23)   |  |  | | --- | --- | | [share](http://stackoverflow.com/q/983030)|[improve this question](http://stackoverflow.com/posts/983030/edit) | asked Jun 11 '09 at 19:10  [[https://www.gravatar.com/avatar/954e2482079fcfe67dfbdddc30d43d93?s=32&d=identicon&r=PG](http://stackoverflow.com/users/116176/jasonh)](http://stackoverflow.com/users/116176/jasonh)  [jasonh](http://stackoverflow.com/users/116176/jasonh) **5,744**32844 | |
|  | |  |  |  |  | | --- | --- | --- | --- | | |  |  | | --- | --- | | **1** |  | | I would put some brackets around // Some code here – [Dolphin](http://stackoverflow.com/users/110672/dolphin) [Jun 11 '09 at 19:18](http://stackoverflow.com/questions/983030/type-checking-typeof-gettype-or-is#comment791186_983030) | | |  |  | | --- | --- | | **2** |  | | Don't forget as! – [RCIX](http://stackoverflow.com/users/117069/rcix) [Oct 7 '09 at 23:52](http://stackoverflow.com/questions/983030/type-checking-typeof-gettype-or-is#comment1390947_983030) | | |  |  | | --- | --- | | **14** |  | | as isn't really type checking though... – [jasonh](http://stackoverflow.com/users/116176/jasonh" \o "5744 reputation) [Oct 8 '09 at 0:48](http://stackoverflow.com/questions/983030/type-checking-typeof-gettype-or-is#comment1391098_983030) | |

**9 Answers**

[active](http://stackoverflow.com/questions/983030/type-checking-typeof-gettype-or-is?answertab=active#tab-top)[oldest](http://stackoverflow.com/questions/983030/type-checking-typeof-gettype-or-is?answertab=oldest#tab-top)[votes](http://stackoverflow.com/questions/983030/type-checking-typeof-gettype-or-is?answertab=votes#tab-top)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| up vote275down voteaccepted | All are different.   * typeof takes a type name (which you specify at compile time). * GetType gets the runtime type of an instance. * is returns true if an instance is in the inheritance tree.   **Example**  class Animal { }  class Dog : Animal { }  void PrintTypes(Animal a) {  print(a.GetType() == typeof(Animal)) // false  print(a is Animal) // true  print(a.GetType() == typeof(Dog)) // true  }  Dog spot = new Dog();  PrintTypes(spot);  What about typeof(T)? Is it also resolved at compile time?  Yes. T is always what the type of the expression is. Remember, a generic method is basically a whole bunch of methods with the appropriate type. Example:  string Foo<T>(T object) { return typeof(T).Name; }  Animal probably\_a\_dog = new Dog();  Dog definitely\_a\_dog = new Dog();  Foo(probably\_a\_dog); // this calls Foo<Animal> and returns "Animal"  Foo<Animal>(probably\_a\_dog); // this is exactly the same as above  Foo<Dog>(probably\_a\_dog); // !!! This will not compile. The parameter expects a Dog, you cannot pass in an Animal.  Foo(definitely\_a\_dog); // this calls Foo<Dog> and returns "Dog"  Foo<Dog>(definitely\_a\_dog); // this is exactly the same as above.  Foo<Animal>(definitely\_a\_dog); // this calls Foo<Animal> and returns "Animal".  Foo((Animal)definitely\_a\_dog); // this does the same as above, returns "Animal"   |  |  |  | | --- | --- | --- | | [share](http://stackoverflow.com/a/983061)|[improve this answer](http://stackoverflow.com/posts/983061/edit) | [edited Jan 20 '12 at 20:01](http://stackoverflow.com/posts/983061/revisions) | answered Jun 11 '09 at 19:15  [[https://www.gravatar.com/avatar/67f11df8aec61f878051a101032c6b2d?s=32&d=identicon&r=PG](http://stackoverflow.com/users/4435/jimmy)](http://stackoverflow.com/users/4435/jimmy)  [Jimmy](http://stackoverflow.com/users/4435/jimmy) **29.5k**666106 | |
|  | |  |  |  |  | | --- | --- | --- | --- | | |  |  | | --- | --- | | **2** |  | | Ah, so if I have a Ford class that derives from Car and an instance of Ford, checking "is Car" on that instance will be true. Makes sense! – [jasonh](http://stackoverflow.com/users/116176/jasonh" \o "5744 reputation) [Jun 11 '09 at 19:19](http://stackoverflow.com/questions/983030/type-checking-typeof-gettype-or-is#comment791198_983061) | |  | To clarify, I was aware of that, but I commented before you added a code sample. I wanted to try to add some plain English clarity to your already excellent answer. – [jasonh](http://stackoverflow.com/users/116176/jasonh" \o "5744 reputation) [Jun 11 '09 at 19:25](http://stackoverflow.com/questions/983030/type-checking-typeof-gettype-or-is#comment791232_983061) | |  | What about typeof(T)? it also resolved at compile time? – [devi](http://stackoverflow.com/users/451495/devi" \o "707 reputation) [Jan 20 '12 at 12:46](http://stackoverflow.com/questions/983030/type-checking-typeof-gettype-or-is#comment11191639_983061) | |  | @Jimmy So what about performance? typeof vs. GetType()? – [Shimmy](http://stackoverflow.com/users/75500/shimmy) [Jan 26 '12 at 4:59](http://stackoverflow.com/questions/983030/type-checking-typeof-gettype-or-is#comment11301298_983061) | | |  |  | | --- | --- | | **1** |  | | @Shimmy if typeof is evaluated at compile time and GetType() is evaluated at runtime, then it makes sense that GetType() incurs a slight performance hit – [Cedric Mamo](http://stackoverflow.com/users/417239/cedric-mamo) [Dec 10 '12 at 13:35](http://stackoverflow.com/questions/983030/type-checking-typeof-gettype-or-is#comment18985332_983061) | |

[](http://engine.adzerk.net/r?e=eyJhdiI6NDE0LCJhdCI6NCwiY20iOjg0NywiY2giOjExNzgsImNyIjoxMDc2OSwiZGkiOiJkNjE2ODA1ODRjOTY0Zjc0YjM0MzBhOTg3NGQ2ZTNiZSIsImRtIjoxLCJmYyI6MTY4ODYsImZsIjoyNDQ0LCJrdyI6ImMjIiwibnciOjIyLCJyZiI6Imh0dHBzOi8vd3d3Lmdvb2dsZS5sdS8iLCJydiI6MCwicHIiOjE1NjgsInN0Ijo4Mjc3LCJ6biI6NDQsInVyIjoiaHR0cDovL2NhcmVlcnMuc3RhY2tvdmVyZmxvdy5jb20vam9icy90ZWxlY29tbXV0ZSJ9&s=BT6HJ8CSpd6mxc8-Bk_tOcNCZzs)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| up vote52down vote | Use typeof when you want to get the type at *compilation time*. Use GetType when you want to get the type at *execution time*. There are rarely any cases to use is as it does a cast and, in most cases, you end up casting the variable anyway.  There is a fourth option that you haven't considered (especially if you are going to cast an object to the type you find as well); that is to use as.  Foo foo = obj as Foo;  if (foo != null)  // your code here  This only uses **one** cast whereas this approach:  if (obj is Foo)  Foo foo = (Foo)obj;  requires **two**.   |  |  |  | | --- | --- | --- | | [share](http://stackoverflow.com/a/983055)|[improve this answer](http://stackoverflow.com/posts/983055/edit) | [edited Feb 22 at 10:16](http://stackoverflow.com/posts/983055/revisions)  [[http://i.stack.imgur.com/01zOh.jpg?s=32&g=1](http://stackoverflow.com/users/1057803/loxdog)](http://stackoverflow.com/users/1057803/loxdog)  [loxdog](http://stackoverflow.com/users/1057803/loxdog) **113**18 | answered Jun 11 '09 at 19:14  [[https://www.gravatar.com/avatar/19ef5c17a12a09f86a0fbeb6f95a3b0f?s=32&d=identicon&r=PG](http://stackoverflow.com/users/34211/andrew-hare)](http://stackoverflow.com/users/34211/andrew-hare)  [Andrew Hare](http://stackoverflow.com/users/34211/andrew-hare) **136k**17311425 | |
|  | |  |  |  |  | | --- | --- | --- | --- | | |  |  | | --- | --- | | **3** |  | | Thanks for the added options on casting! – [jasonh](http://stackoverflow.com/users/116176/jasonh" \o "5744 reputation) [Jun 11 '09 at 19:27](http://stackoverflow.com/questions/983030/type-checking-typeof-gettype-or-is#comment791245_983055) | | |  |  | | --- | --- | | **3** |  | | +1 for as, a great operator. – [Neil Williams](http://stackoverflow.com/users/9617/neil-williams) [Jun 11 '09 at 19:33](http://stackoverflow.com/questions/983030/type-checking-typeof-gettype-or-is#comment791281_983055) | |  | With the changes in .NET 4 does is still perform a cast? – [ahsteele](http://stackoverflow.com/users/61654/ahsteele" \o "8910 reputation) [Mar 21 at 21:04](http://stackoverflow.com/questions/983030/type-checking-typeof-gettype-or-is#comment22049048_983055) | |