

Weak references have no effect on object lifetime

 devblogs.microsoft.com/oldnewthing/20130807-00

August 7, 2013



Raymond Chen

The `WeakReference` class lets you retain a reference to an object while still permitting the object to be garbage collected. When that happens, then the `IsAlive` property is `false` and the `Target` property is `null`. ([Related discussion](#).) Note, however, that weak references do not alter the decision of the garbage collector whether or not an object is garbage. It merely lets you observe the garbage collector's decision. Some people think that `WeakReference` means "Treat this as a regular (strong) reference most of the time, but if there is memory pressure, then you can reclaim the object." This type of reference is called a `SoftReference` in Java, but the CLR has no analogous concept as far as I'm aware. In the CLR, weak references do not extend the lifetime of an object.

It's sort of like the Prime Directive from Star Trek: Weak references follow a policy of non-interference with the natural course of the GC.

[Raymond Chen](#)

Follow

