

# Boo.Lang.Useful

A todo/wish list for stuff to include in Boo.Lang.Useful.dll - a library of helper classes and functions for boo users.

All the code in Boo.Lang.Useful would be in boo, not C#.

[Useful things about Boo](#) contains documentation on members present in the assembly.

## Boo.Lang.Useful.Attributes

- [AsyncMethodAttribute](#)
- [DisposableAttribute](#)
- [SingletonAttribute](#)
- Implements? - delegates implementation of an interface to a class field instance.

## Boo.Lang.Useful.Macros

- [fireAndForget](#) - used to asynchronously invoke a method without having to worry to call EndMethod to clean up the asynchronous call.
- [performTransaction](#) - in the examples folder
- [timelt](#) - used to time how long a section of code takes.
- [with](#) - in the examples folder included with boo

## Compiler Steps & Pipelines

- [Doc step](#) - generates XML documentation from the docstrings in your boo code, which can be used by NDoc to generate HTML documentation.
- Macro helper scripts - Anything that could help developers creating boo macros/attributes would be nice. See for example the script at the bottom of [Compiler Steps](#), it shows how the AST is transformed after each step, in XML or boo form.
- [Style checker](#) - there's a sample pipeline in the examples folder that checks the style of your boo code (like prefixing private fields with an underscore, for example)
- [AutoImport](#) - in the examples folder

## Helper Classes

- [Set Class](#) - needs to be recoded in boo instead of C#
- Boo.Dynamic? - See the bottom of the [Duck Typing](#) page for examples of python-like and other dynamic IQuackFu-based classes we could create.