Jinx Overview

# What is Jinx?

Jinx is a lightweight embedded scripting language, written in modern C++, and intended to be compiled and used from within a host application. The Jinx language is clean and simple, looking like a cross between pseudo-code and natural language phrases, thanks to functions and identifiers that can incorporate whitespace.

It was inspired by languages such as Lua, Python, C++, AppleScript, and a few others. A significant design goal was to make Jinx not only simple to use as a language, but also to make it straightforward to integrate and use via the native API. Jinx scripts are designed to be executed asynchronously as co-routines, or even on separate threads, since the runtime is thread-safe. The language also uses no garbage collection, ensuring predictable CPU usage. This makes it a potential candidate for real-time applications like videogames.

# Documentation

As part of ensuring Jinx is as easy as possible to use, a comprehensive tutorial has been written, and API and library documentation has been provided.

[Tutorial](Tutorial.htm)

[API Documentation](API/html/index.html)

[Core Library Reference](LibCore.htm)

# Current Release Status

Jinx should currently be considered *alpha quality* software. The core language, bytecode, libraries, and APIs may change as issues are found and fixed, and feedback based on actual use and experience may be considered. The library may still contain serious bugs that must be fixed.

# License

Jinx is released under the [MIT license](LICENSE.txt). You are free to use this library in any project, open or closed source, commercial or personal. I would certainly appreciate hearing from anyone who uses Jinx in their project, but there is no obligation to do so.

# Feedback

Feel free to e-mail me at [james.boer@](mailto:james.boer@)gmail.com or contact me via the GitHub project at <https://github.com/JamesBoer/Jinx>. I’m curious to hear if anyone makes use of this scripting library besides me, and how it works out for you.