Languages that was single threaded and became multiple threaded

There have been several programming languages that initially started as single-threaded but eventually introduced support for multithreading. Here are a few examples:

1. **Python:** Python is a widely used high-level programming language. Initially, Python had a Global Interpreter Lock (GIL), which allowed only one thread to execute Python bytecode at a time. This limitation made it difficult to achieve true parallelism in Python. However, there have been efforts to bypass the GIL, such as the introduction of multiprocessing modules, which enable the use of multiple processes instead of threads.

2. **JavaScript:** JavaScript is the primary language for web development. It started as a single-threaded language, executing code in a single thread within the browser. However, with the introduction of HTML5 and the Web Workers API, JavaScript gained support for multithreading. Web Workers allow running scripts in the background without blocking the main UI thread, enabling concurrent execution.

3. **Ruby:** Ruby is a dynamic, object-oriented programming language. Initially, Ruby had a Global Interpreter Lock (GIL) similar to Python, which limited its multithreading capabilities. However, with the development of the JRuby implementation, which runs Ruby on the Java Virtual Machine (JVM), developers gained access to Java's concurrency libraries and were able to utilize Java threads effectively.

4. **PHP:** PHP is a popular server-side scripting language used for web development. Historically, PHP was designed to handle requests in a single-threaded manner. However, with the release of PHP 7, PHP introduced native support for parallelism and multithreading through the Swoole extension. Swoole allows developers to write concurrent PHP code using coroutines and asynchronous I/O operations.