Introductory Asynchronous Programming

Contact Info

Name: Donald Wilcox

Email: dw@jogral.io

1 Overview

An application goes to production, and functions well for some time. Then, as users and resources grow, performance issues happen, and one type of solution that developers will consider is asychronous development (or parallelism). Classically, this is where threading becomes important, but the challenge is that programming is typically synchronous, while this is asynchronous, and it can quickly become confusing to make an asynchronous program work the way it was intended–especially when it was originally synchronous. This course will discuss issues and solutions necessary to make you confident in dealing with async programming.

2 Duration

This is a 4-hour course.

3 Objectives

- Learn what is asynchronous programming and how it works
- Learn when to use asynchronous programming
- Implement a simple HTML fetcher using asynchronous programming and Python

4 Key Takeaways

After this course, you will be able to:

- Know the difference between CPU-bound and IO-bound programs
- Resolve a computational bottleneck
- Resolve an I/O bottleneck
- Create a web page fetcher that can process multiple URLs in parallel

5 Prerequisites

- Requires a working knowledge of Python or another programming language

6 Required Materials

For this, you will need the following:

- Workbook (provided)
- Laptop with Docker

7 Schedule/Topics Discussed

- What is asynchronous?
- Benefits of asynchronous?
- Challenges with asynchronous
- Fitting a program to be asynchronous