Java Fundamentals – Concurrency with Multithreading

INTRODUCTION

David Flynn
SOFTWARE ENGINEER, FLYNN IT LTD

Why Are We Here?

Concurrency is a difficult topic to master

- It is present in most Java software
- You're likely to be tested on it
- Concurrent programs are rich and interesting

Our Time Together Is Limited

Additional material:

- Oracle documentation and tutorials
- Library documentation and code
- Stack Overflow

Textbook

- *Java Concurrency In Practice* - Goetz et al.

Understanding Threading and Concurrency

Defining the concepts

- Useful for interviews
- We'll simulate one

Creating and Managing Threads

Basics of Java threading

- Creating
- Managing
- Monitoring

Sharing Memory Across Threads

How to safely share memory

Avoid bugs

- Hard to detect
- Paradoxical
- Difficult to debug

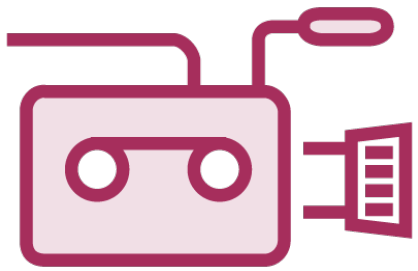
Understanding Mutexes

Avoiding race conditions

- Threads interleaving unexpectedly

Prevent using mutexes

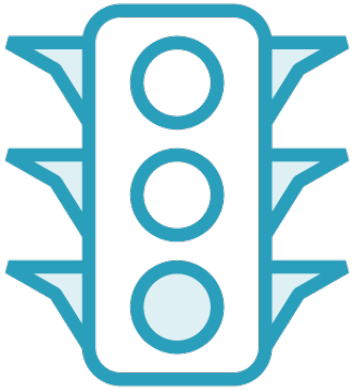
Liveness Issues



Preventing deadlock, livelock and starvation

- We'll look at some solutions
- But no substitute for careful design

Inter-thread Communication and Signalling



Look at producer/consumer model

Implement using:

- wait() and notify()
- BlockingQueues

Thread Pools

Manage threads for us

- Help solve concurrent problems
- By executing tasks

Why Bother with Java Libraries?

And not: Akka, Spring, ...

Need to have familiarity with the basics

Frameworks don't magically solve all problems

You still might use the Java libraries

How to Get the Most Out

Watch the Videos

Examine the demonstration programs

- Tinker with them

Replay and review difficult sections

Thank You!

Hannah McDonald

Peter O'Hanlon

The Pluralsight staff

Enjoy!

We hope you enjoy this as much as we did putting it together