JAVA FOR SMALL TEAMS

Guidance for good server side code

Table of Contents

Introduction

What is This?

This book is an attempt to capture what *"good"* Java code looks like and the practices that help produce it.

This is a problematic document to write.

There is no one right answer to what good code looks like and there are many well-respected books that serve the same purpose such as *Effective Java*, *Clean Code* and others.

So why this document?

It differentiates itself by being :

- Freely distributable
- Open for update - contributions, corrections and updates are encouraged
- Brief - much is left out in an attempt to be easily digestible
- Narrow - it captures one opinion of *"good"* appropriate for a specific context

This last point is important.

We assume a number of things about you and the environment you are working in.

- We assume you are writing server side Java in *small* teams.
- We assume your teams are of mixed experience (some experts, some beginners).
- We assume you are writing code in a general "business" context.
- We assume you expect the code to still be in use in five years' time.

Some of the suggestions may be valid in other contexts, others might constitute terrible advice for those contexts.

It is also just one opinion from many valid alternatives. To be useful it needs to be an opinion that you can agree with and sign up to. If you disagree with something in this book please make your own thoughts known so it can be improved.

Finally, not all the code we work on is perfect. Sometimes we inherit our own mistakes, sometimes we inherit other people's.

The point of this document is not to say that all code must look like this but to have an agreed destination that we are aiming for.

Who is This For?

This document is intended for consumption by anyone involved with writing server side Java code. From developers writing Java for the first time through to seasoned technical leads serving multiple teams.

Some sections will be more relevant to some audiences than others but we encourage everyone to at least skim all sections even if you do not read them in depth.

Structure

The document is split into five sections:

- Process - Discussion on development philosophy and workflow
- Style - Good style and design at a high level
- Specifics - More specific advice on Java language features and gotchas
- Good tests - How to write good tests
- Bad advice - Discussion of some commonly circulated bad advice and patterns

Version

This book is updated often. The latest changes to the book can be viewed online at gitbook.com.

Versioned releases are available for free from the book's website.

If you are reading a PDF or print copy of this book the release version will be displayed on the inside cover. If there is no inside cover then you are reading an unreleased version of the book.

History

Most of the content of this book began life as internal wiki pages at NCR Edinburgh. We started to convert the wiki into this book at the end of 2015 so that it could be easily shared with other parts of our company.

Rather than keep this as an internal document we decided to open it up to everyone in the hope that together we could make it better.