CHAPTER 23

Convert Your Codebase

At this point, you're likely to have an existing Java codebase and may be wondering if you should try and convert the Java to Scala.

It may be a lot of effort, but I say it's definitely worth investing in converting the entire codebase to Scala if you're serious about adopting Scala. Don't aim to leave half and half. Accept that it's going to take some time and effort but commit to achieving it. If you don't finish the job, apathy can set in, not to mention that inter-op between old Java and Scala can get really awkward. You'll make your life easier if you see the job through.

IntelliJ IDEA has a convert function, which is a great starting place. It'll let you quickly turn a Java class into Scala, but be prepared: it's not often idiomatic and may not even compile without coercion. Treat any automated conversion as a learning exercise. Don't accept the transformation at face value, but look at the results and apply your judgement to tidy up. For example, rather than create a case class, IntelliJ often creates fields within the Scala class. You'll usually be able to rephrase a verbatim conversion in a more Scala-savvy way.

Consider your build tools. These will need to be updated. SBT and Maven both support mixed Java/Scala projects so there's no excuse not to keep running your continuous integration and test environments. You might also want to consider whether it's worth porting over your existing build to SBT.