

Gradle

Fernando Camargo

5 de junho de 2017

ZG Soluções

Por que uma ferramenta de Build?

Por que uma ferramenta de Build?

- Projeto independente de IDE

Por que uma ferramenta de Build?

- Projeto independente de IDE
- Automação de build

Ant, Maven e Gradle

- Primeira build tool para Java

- Primeira build tool para Java
- Extrema flexibilidade

- Primeira build tool para Java
- Extrema flexibilidade
- Não impõe convenções em projetos Java

Exemplo de Ant

```
<project>

  <target name="clean">
    <delete dir="build"/>
  </target>

  <target name="compile">
    <mkdir dir="build/classes"/>
    <javac srcdir="src" destdir="build/classes"/>
  </target>

  <target name="jar">
    <mkdir dir="build/jar"/>
    <jar destfile="build/jar/HelloWorld.jar" basedir="build/classes">
      <manifest>
        <attribute name="Main-Class" value="oata.HelloWorld"/>
      </manifest>
    </jar>
  </target>

  <target name="run">
    <java jar="build/jar/HelloWorld.jar" fork="true"/>
  </target>

</project>
```

- Flexível demais → projetos não possuem estrutura padrão

Problemas do Ant

- Flexível demais → projetos não possuem estrutura padrão
- Muito verboso → escreve-se muito para uma build simples

Problemas do Ant

- Flexível demais → projetos não possuem estrutura padrão
- Muito verboso → escreve-se muito para uma build simples
- Não possui gerenciamento de dependências

- Convenção sobre Configuração → escreve-se pouco para uma build simples

- Convenção sobre Configuração → escreve-se pouco para uma build simples
- Gerenciamento de dependências com resolução de dependências transitivas

Estrutura de diretórios

Diretório	Função
src/main/java	Código fonte
src/main/resources	Recursos não compilados
src/test/java	Código de testes
src/test/resources	Recursos não compilados de testes
src/main/webapp	Recursos WEB
target	Resultados de build

Gradle

Conclusões
