## Lesson 1. Intro

Before build tools projects were built by batch/shell scripts. It had drawbacks:

* Platform dependence (unix/windows required different scripts)
* Scripts from on project didn’t fit to another
* No uniform project structure (more onboarding time when you come on new project)

2000 – released Apache Ant which were more convenient than scripts.

2002 – released Apache Maven which was better because supplied uniformed project structure.

2007 – released Gradle which can use Groovy/Kotlin instead of xml.

## Lesson 7. Effective POM

All POM files inherit super POM (like object in java) and parent POM files. To see result POM (effective) there is a goal:

mvn help:effective-po

-----------------------------------------------------------------------------------------------------------------------------------------------------------

## Lesson 8. Dependency management

Classpath – directory where all our dependencies (.jar) are stored. They are not in our project but in maven local repository in our PC (~/.m2/repository). If we add a new dependency in our project maven first look for it in local repo and then in remote repo.

${maven}/conf/settings.xml – here we can change path to local repo (better not to do it)

-----------------------------------------------------------------------------------------------------------------------------------------------------------