**Лекция 1**

Фреймворк навязывает использовать собственный стиль разработки (структуру, синтаксис)

**Maven** позволяет:

1 Осуществить управление жизненным циклом

2 Получить доступ к сторонним библиотекам

3 Добавить плагины

4 Быстрее тестировать

Pom (project object model) файл – в нем записывается структура проекта

Артефакт – любая библиотека, хранящаяся в репозитории

Существует три вида репозиториев – центральный (самый главный, mvn репозиторий, в нем содержатся все библиотеки), удалённый (тот, который находится не у нас, неофициальный, друг например написал)

Зависимость (dependency) – стороння библиотека, которая используется в проекте

Плагины

Библиотека GSON для работы с объектами в json формате

groupId к какому региону, организации принадлежит разработку

artefactId указать сборку проекта (название проекта)

после dependencies в билде

Архетип – стандартная компоновка файла (мавен пож все нужды предлагает). В идеале создавать при его помощи

Шаблон который подготавливает файлы под любые проекты

9 фаз:

Clean – удаляются все скомпилированные файлы из каталога target (место, в котором хранятся готовые артефакты)

Validate – идет проверка, вся ли информация доступна для сборки проекта

Compile – компилируются файлы с иходным кодом

Test – запускаются тесты

Package – упаковываются скомпилированные файлы (jar, war)

JDBC - API для независимого соединения БД и Java

Wora – write once run anywhere

Javac - компилятор

Jip -java интерпретатор

Изменение для второго задания