

# Основы Java EE

Основы Maven. Практика

# Основы Java EE

Автор курса



Бондаренко Андрей

# Основы Java EE

После урока обязательно



Повторите этот урок в видео формате на  
[ITVDN.com](http://itvdn.com)



Проверьте как Вы усвоили данный материал на  
[TestProvider.com](http://TestProvider.com)

# Основы Maven. Практика

# Maven

## Maven

- Единая технология создания, сборки и введения проекта.
- Инструмент для управления проектами:
  - сборка
  - тестирование
  - управление зависимостями
  - работа с системами контроля версий – SCM (Source Code Manager)
- Отличия от Ant – не нужно самостоятельно все прописывать, просто настраиваете – остальное делают готовые плагины
- Поддержка всех современных IDE

# Maven

## Понятия Maven

- **Plugins** – плагины, дополнения на все случаи жизни (некоторые подключены по-умолчанию, можно подключать дополнительно)
- **Repository** – хранение артефактов, зависимостей (локальный – папка .m2, удаленный – центральный, дополнительный)
- **Зависимостями** могут пользоваться любые проекты (в каждом проекте не будут дублироваться одинаковые jar, актуальная версия).
- **POM** – Project Object Model – XML файл с настройками (pom.xml), используется мавеном для построения проекта. Могут наследоваться друг от друга. Любой pom по-умолчанию наследуется от главного pom (похоже на java с наследованием всех объектов от Object).
- **Artifact** – любой файл в результате работы maven (сборка проекта) – jar, war, ear и пр.
- **Coordinates** – идентификаторы (координаты) артефакта
  - groupId – org.hibernate
  - artifactId – hibernate.core
  - version – 3.6.9. Final
- **Arhitecture** – шаблон нового проекта (структура, название папок и пр.).

# Maven Project

## Структура проекта Maven

- `src/main/java` – хранение кода java
- `src/main/resources` – хранение ресурсных файлов (txt, properties, sql и т.д).
- `src/main/webapp` – хранение веб ресурсов приложения (view), таких как html, css, jsp, jsf.
- `src/test/java` – тест кода java.

# Install Maven

## Установка Maven в IntelliJ Idea

- Вместе с IntelliJ устанавливается определенная версия maven – можно использовать ее или скачать последнюю – актуальную.
- В первых запусках Maven скачивает все необходимые зависимости, плагины и пр. – может занять время.
- Желательно сразу создавать проект как Maven, чтобы не переносить и перенастраивать все файлы
- Можно все делать через командную строку, но лучше привыкнуть к IDE.



# Maven

## Maven

- Построение приложения подчиняется жизненному циклу (default(build), clean, site).
- Phase - фаза
- Goal - задача

# Disadvantages Maven

## Недостатки Maven

- Сложность в освоении (в начале).
- Мало документации, решение лучше искать на форумах (например [stackoverflow.com](http://stackoverflow.com))
- Путаница в артефактах – какой и где использовать.
- Не информативные ошибки.
- Стандартность, формализованность – для дополнительных нестандартных действий необходимо создать свою реализацию.

# Maven

## Maven

Lifecycle

Phase

Phase

Plugin

Plugin

Goal

Goal

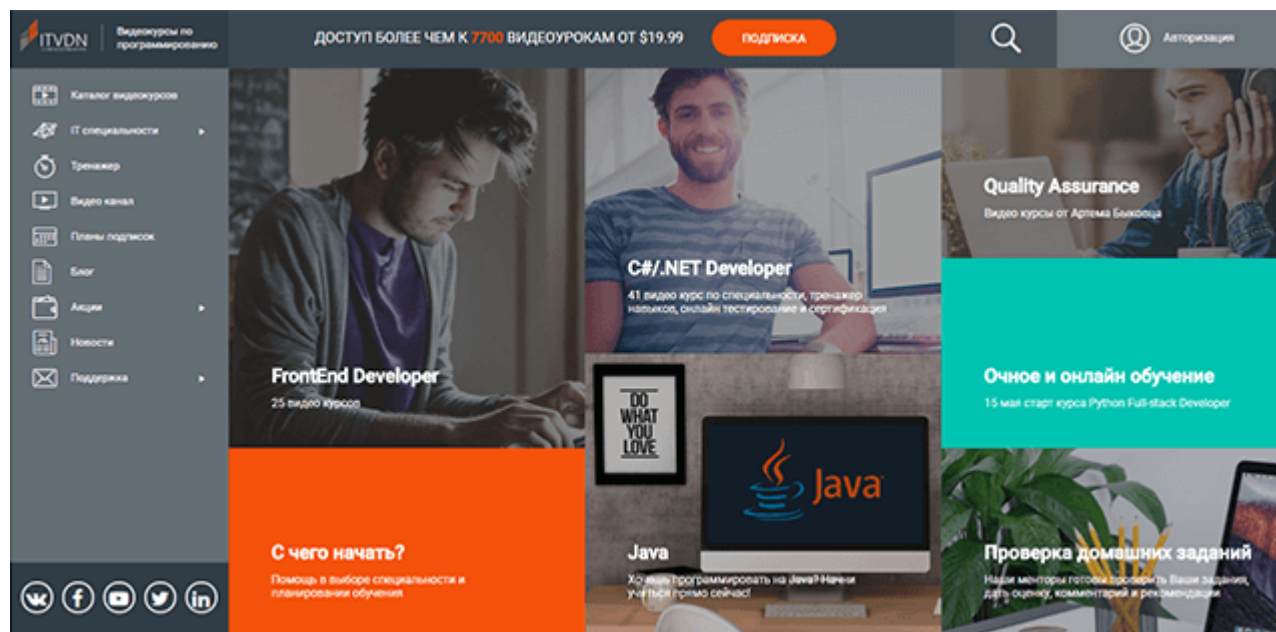
Goal

Goal

Каждый элемент может вызываться отдельно

# Смотрите наши уроки в видео формате

ITVDN.com



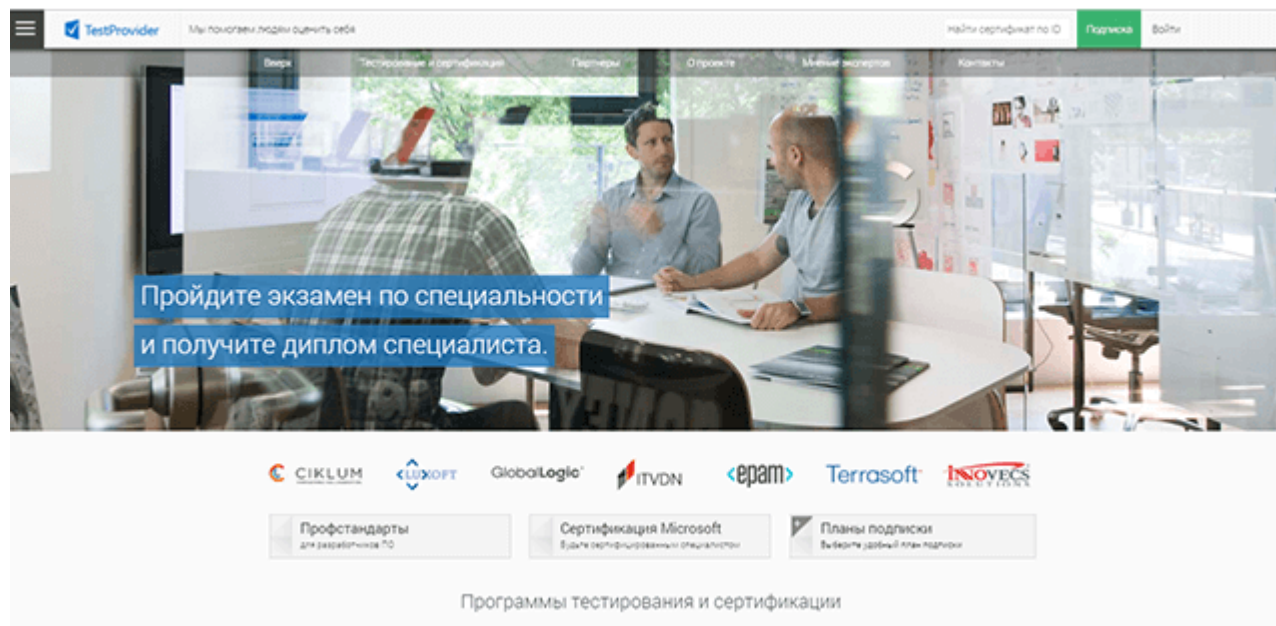
Посмотрите этот урок в видео формате на образовательном портале [ITVDN.com](http://ITVDN.com) для закрепления пройденного материала.

Курсы записаны сертифицированными тренерами, которые работают в учебном центре CyberBionic Systematics и другими высококвалифицированными разработчиками.



# Проверка знаний

TestProvider.com



TestProvider – это online сервис проверки знаний по информационным технологиям. С его помощью Вы можете оценить Ваш уровень и выявить слабые места. Он будет полезен как в процессе изучения технологии, так и для общей оценки знаний IT специалиста.

После каждого урока проходите тестирование для проверки знаний на [TestProvider.com](https://testprovider.com)

Успешное прохождение финального тестирования позволит Вам получить соответствующий Сертификат.



Q&A

# Информационный видеосервис для разработчиков программного обеспечения

