

Фазы Maven-проекта

JSP & Servlets
1 уровень, 4 лекция

ОТКРЫТА

5.1 Список фаз проекта

Наконец-то мы добрались до сборки проекта. И тут ты будешь немного удивлен. Ну или сильно, как получится. Maven пересмотрел подход к сборке проекта. И сейчас ты в этом убедишься.

Вся сборка проекта была разбита на фазы, описание которых я приведу в таблице ниже:

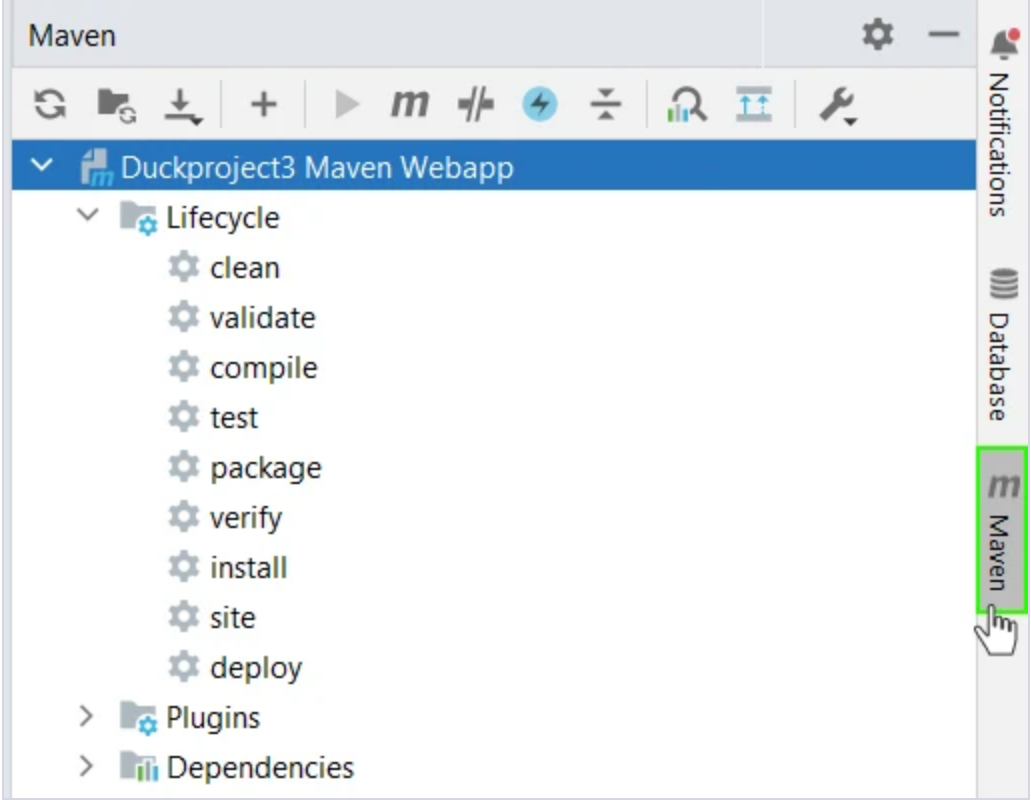
Order	Фаза	
1	validate	проверяет корректность метаинформации о проекте
2	compile	компилирует исходники
3	test	прогоняет тесты классов из предыдущего шага
4	package	упаковывает скомпилированные классы в новый артефакт: jar, war, zip, ...
5	verify	проверяет корректность артефакта и удовлетворение требованиям качества
6	install	кладет артефакт в локальный репозиторий
7	deploy	заливает артефакт на production-сервер или удаленный репозиторий

При этом **шаги четко последовательны**. Если ты скомандуешь Maven выполнить команду package, он перед этим выполнит фазы validate, compile, test и только затем package

В принципе ничего нового тут нет, разве что появились отдельные фазы для контроля качества: validate, test, verify. И целых две фазы для деплоя сборки – install и deploy.

Чтобы запустить определенную фазу, достаточно написать команду **maven phase**. Например, для сборки нужно выполнить команду **maven package**. И тому подобное.

IntelliJ IDEA отлично умеет работать с этими фазами и у нее **справа** есть специальное меню для этих целей:



Тут кроме известных фаз, IDEA отображает еще 2 команды: **clean** и **site**. clean используется для того, чтобы полностью очистить папку target, а site умеет создавать документацию проекта.

5.2 Сборка проекта

Если ты хочешь скомпилировать проект, то нужно запустить фазу compile. Это можно сделать с помощью командной строки: **mvn compile** или через интерфейс IDEA – кликнув по пункту **compile**.

После этого Maven запустит сборку проекта и ты увидишь лог процесса сборки подобный этому:

```
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 0.742 s
[INFO] Finished at: 2016-09-19T22:41:26+04:00
[INFO] Final Memory: 7M/18M
[INFO] -----
```

Если что-то пошло не так, то лог будет типа такого:

```
[ERROR] Failed to execute goal org.apache.maven.plugins:maven-compiler-plugin:3.8.0:compile (default-compile)
[ERROR]
[ERROR] To see the full stack trace of the errors, re-run Maven with the -e switch.
[ERROR] Re-run Maven using the -X switch to enable full debug logging.
[ERROR]
[ERROR] For more information about the errors and possible solutions, please read the following articles:
[ERROR] [Help 1]
http://cwiki.apache.org/confluence/display/MAVEN/MojoExecutionException
```

В логе будет куча полезной информации, со временем ты научишься ее понимать и ценить.

5.3 Циклы работы

Все команды maven делятся на три группы – lifecycles. Их называют **жизненными циклами**, так как они задают порядок фаз, которые выполняются во время сборки или определенного жизненного цикла, потому что не все действия Maven являются сборкой.

Всего жизненных циклов три:

- **clean;**
- **default;**
- **site.**

И у каждого из них есть свой порядок фаз. У clean он самый короткий:

- 1. **pre-clean**;
- 2. **clean**;
- 3. **post-clean**.

Были добавлены скрытые дополнительные фазы pre-clean и post-clean, чтобы сценарий очистки можно было сделать более гибким.

Затем идет **жизненный цикл site**, который, как вы уже знаете, предназначен для автоматической генерации документации по проекту. Он состоит из четырех фаз:

- 1. **pre-site**
- 2. **site**;
- 3. **post-site**;
- 4. **site-deploy**.

Стандартные жизненные циклы могут быть дополнены функционалом с помощью **Maven-плагинов**. Об этом расскажем позднее, ведь это очень интересная тема, которая заслуживает отдельной лекции.

И самый длинный список фаз у сценария default:

- 1. **validate**;
- 2. **generate-sources**;
- 3. **process-sources**;
- 4. **generate-resources**;
- 5. **process-resources**;
- 6. **compile**;
- 7. **process-test-sources**;
- 8. **process-test-resources**;
- 9. **test-compile**;
- 10. **test**;
- 11. **package**;
- 12. **install**;
- 13. **deploy**.

Тут есть все те же фазы, о которых вы уже знаете, но добавились еще несколько необязательных.

Во-первых, популярный в больших проектах этап – **generate-sources**: генерация Java-кода на основе XML, например. И парный ему **process-sources**, который что-то с этим кодом делает.

Во-вторых, генерация ресурсов – **generate-resources** и парный ему метод **process resources**. Ты будешь часто видеть какие-нибудь действия, привязанные к этим фазам в крупных проектах.

И наконец – тестирование. Оно имеет три дополнительных необязательных фазы, которые помогают сделать запуск фазы тестирования максимально гибким: process-test-sources, process-test-resources, test-compile.

< Предыдущая лекция

Следующая лекция >



Комментарии (5)

популярные

новые

старые

Введите текст комментария

ОНННВАВУ

Уровень 33, Учалы, Russian Federation

24 октября, 16:18

...

Ребят, поделитесь пж какими-нибудь хорошими ресурсами, видосами с ютуба по maven'y

Ответить

Ольга

Уровень 41, Москва, Россия

25 октября, 23:53

...

Добрый вечер. Мне понравилось это видео <https://www.youtube.com/watch?v=IAbZVA4tK6M>
В целом Немчинский неплохо объясняет

Ответить

comrade_b

Уровень 38, Амстердам, Нидерланды

1 ноября, 13:09

...

Я еще не посмотрел полностью видос, и я только вначале этого квеста... но меня смущает дата видео - 2013, а еще больше смущает дата презентации в видео - 2007. Итого 15 лет... Мне кажется, надо что-то поновее будет поискать. А то пахнуло динозаврами

Ответить

Ольга

Уровень 41, Москва, Россия

позавчера, 18:15

...

Основы мавена не изменились с тех пор, мне понравилось, что объясняют доходчиво для новичков. В целом конечно дело каждого.

Ответить

Модератор

Безработный с 24.12.2021

29 мая, 02:15

...

за это лайк

Ответить

ОБУЧЕНИЕ

- Курсы программирования
- Курс Java
- Помощь по задачам
- Подписки
- Задачи-игры

СООБЩЕСТВО

- Пользователи
- Статьи
- Форум
- Чат
- Истории успеха
- Активности

КОМПАНИЯ

- О нас
- Контакты
- Отзывы
- FAQ
- Поддержка



JavaRush — это интерактивный онлайн-курс по изучению Java-программирования с нуля. Он содержит 1200 практических задач с проверкой решения в один клик, необходимый минимум теории по основам Java и мотивирующие фишки, которые помогут пройти курс до конца: игры, опросы, интересные проекты и статьи об эффективном обучении и карьере Java-девелопера.

ПОДПИСЫВАЙТЕСЬ

ЯЗЫК ИНТЕРФЕЙСА

 Русский



СКАЧИВАЙТЕ НАШИ ПРИЛОЖЕНИЯ



"Программистами не рождаются" © 2022 JavaRush