

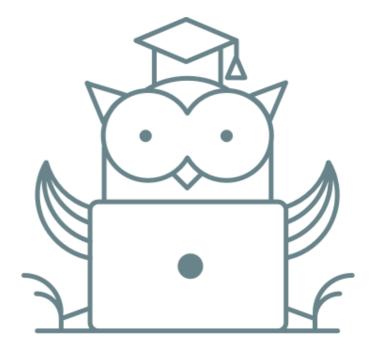
ОНЛАЙН-ОБРАЗОВАНИЕ



# Подготовительный курс Java

Модуль 2.3 «Unit-тестирование»

Виталий Чибриков преподаватель





## Концепция Unit-тестирования

Unit-тестирование это одна из стадий разработки ПО при которой независимые части приложения, которые называют юнитами, проверяют на соответствие задачам, которые они должны решать.

Процесс тестирования состоит из трех частей:

- Инициализация юнита
- Воздействие на юнит
- Проверка соответствия нового состояния ожидаемому





- 100% control of environment
- created by developer
- can be launched before commit (or push)
- can be launched in Continuous Integration
- Code Coverage Metrics
- total isolation
- order independent
- one test one check
- repeatable.





## Подключение библиотек

Библиотека это файл (обычно с расширением jar), который нужно скачать и подключить в проект.

https://github.com/junit-team/junit4/wiki/download-and-install





## Git

#### Clone

git clone URL

Например: <a href="https://github.com/vitaly-chibrikov/otus-prejava-2018-01">https://github.com/vitaly-chibrikov/otus-prejava-2018-01</a>

### **Update**

git pull





### Git

#### Add a file

```
git status  // «nothing to commit, working tree clean»
vim test  // git status » test
git add .  // git status » new file: test
git reset test  // git status » test
git add .
git commit -m "file test added" .
git push URL
```





### Git

#### Reset and revert

```
vim test  // git status » modified: test.txt
git commit -m "new changes" .
git status  // branch is ahead of 'origin/master' by 1 commit.
git log
git reset <ID>  // git status » modified: test.txt
git reset --hard <ID>
git status  // «nothing to commit, working tree clean»
```





### Maven

Система контроля зависимостей и сборки проекта

https://mvnrepository.com/

```
<dependency>
     <groupId>junit</groupId>
          <artifactId>junit</artifactId>
          <version>4.12</version>
          <scope>test</scope>
</dependency>
```





### Использования библиотеки JUnit

Перед использованием

1. Чем тестировать

https://mvnrepository.com/artifact/junit/junit/4.12

https://mvnrepository.com/artifact/org.hamcrest/hamcrest-all/1.3

- 2. Что тестировать. Например, классы из М2.2
- 3. Где писать тесты. В директории test на уровне src





### Использования библиотеки JUnit

Каждый тест состоит из тех действий:

- Инициализация юнита
- Воздействие на юнит
- Проверка соответствия нового состояния ожидаемому





## Использования библиотеки JUnit

Тесты могут проверять не только нормальную работу приложения, но и исключительные ситуации

