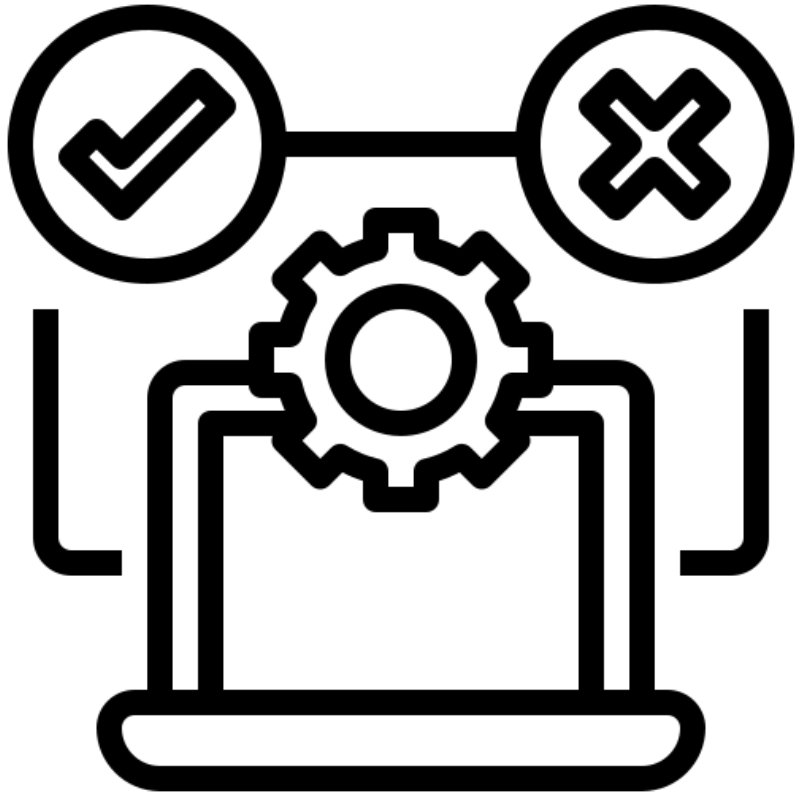
АВТОМАТИЗИРОВАННОЕ ТЕСТИРОВАНИЕ

Лекция 2: Unit-тесты



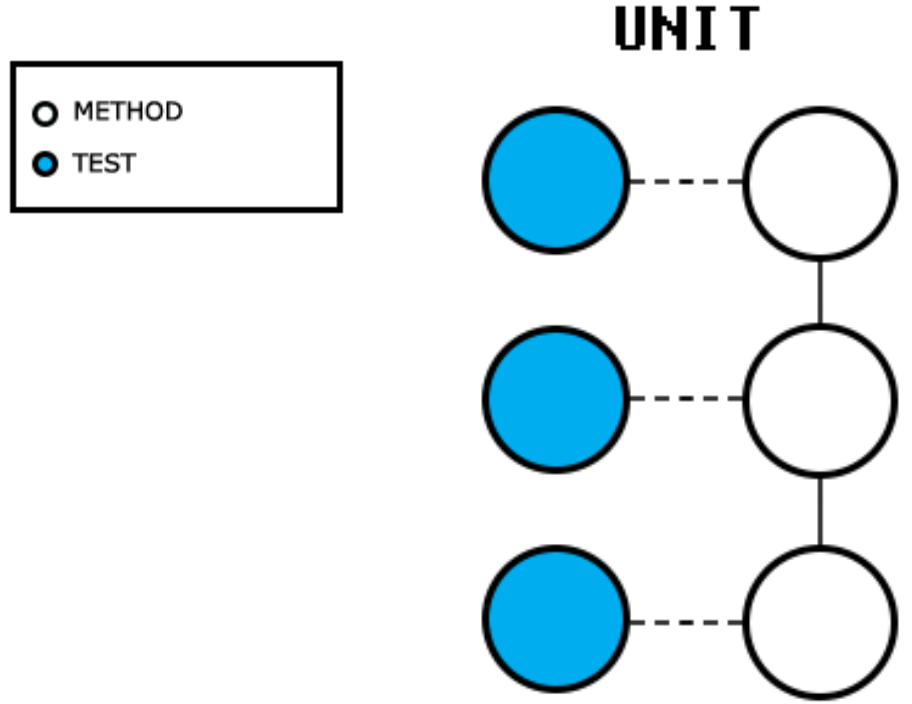
БАШАРИНА ЕКАТЕРИНА АЛЕКСАНДРОВНА

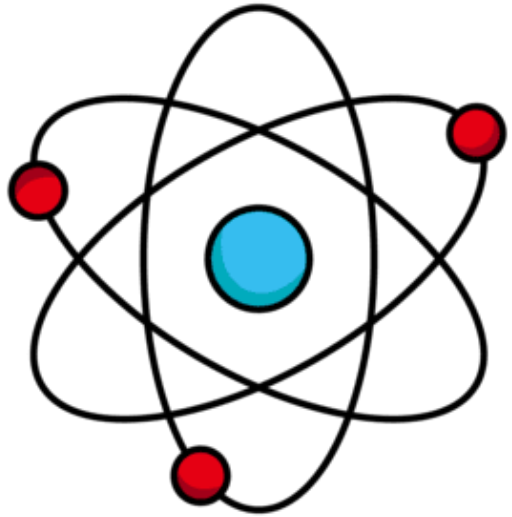
Unit-тестирование (оно же модульное)

- Повторение: что такое Unit test?
- Аннотации before-test-after
- Тестовые фреймворки. Junit, TestNG
- Создание простейших тестов и запуск их через Maven

Юнит тестирование

- Тестирование атомарных частей кода
- Большое количество тестов
- Всегда автоматизированы
- Независимы





Как «работает» юнит тест?

Некоторый модуль программы

Ожидаемое поведение

Фактическое поведение

Что требуется для запуска теста?

- Модуль для тестирования
- Тестовый сценарий
- Входные данные (!)
- Ожидаемый результат

Assertion

- **Assert (англ. утверждать) – конструкция для проверки предположений**

- **Пример:**

```
bool  uselessBool = True
Assert(uselessBool == True)
```

-> результат ассерта положительный – тест пройден

```
bool  uselessBool = False
Assert(uselessBool == True)
```

-> результат ассерта отрицательный – тест не пройден

Какие бывают Assert`ы?

Всё зависит от ЯП ☺

- **AssertTrue / AssertFalse**
- **AssertEquals / AssertArrayEquals**
- **AssertDoesNotThrow**

…

Аннотации Before-Test-After

- Не каждый тест может функционировать сам по себе! Иногда нужны вспомогательные методы

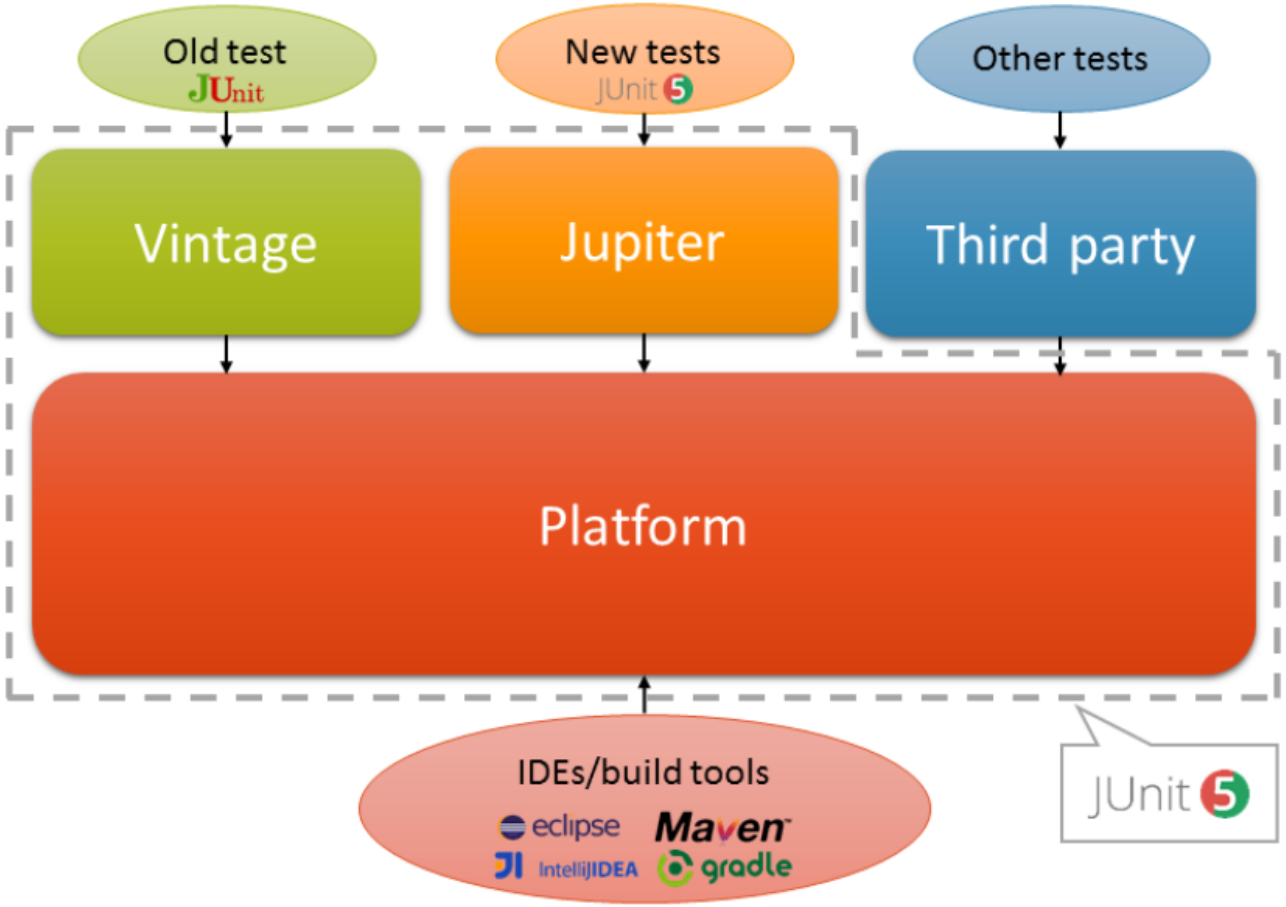
- BeforeTest – инструкции, которые выполняются ДО теста
- Test – непосредственно сам тест
- AfterTest – инструкции, которые выполняются ПОСЛЕ теста

Тестовые фреймворки

- **Стандартизация**
- **Упрощение процесса написания**
- **Широкая применимость**

JUnit

- **Базовый тестовый фреймворк на Java**
- **Большие различия между версиями! (Junit4 и JUnit5 слабо совместимы)**



TestNG

- **Фреймворк на основе JUnit, но намного более «продвинутый»**
- **Гибкость и расширяемость**
- **Акцент на покрытии и группировке тестов**

JUnit vs TestNG

- Аннотации @Before и @After общие для классов и методов
- Аннотация @Ignore для пропуска теста
- Группировка аннотацией @Suite
- Параметризация через аннотацию @ParameterizedTest и @...Source

- Отдельные аннотации @BeforeClass и @BeforeMethod
- Параметр enabled в аннотации @Test
- Группировка напрямую из pom.xml
- Параметризация через @DataProvider

Написание тестов!

- **Создаём проект:**

```
mvn archetype:generate
-DgroupId=org.example
-DartifactId=test-app
-DarchetypeArtifactId=maven-archetype-quickstart
-DarchetypeVersion=1.4
-DinteractiveMode=false
```

Изменяем в pom.xml зависимости и версии

- **В результате будет сгенерирован проект:**

Спасибо за внимание!