



НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ

Факультет компьютерных наук
Департамент программной инженерии

ОБЕСПЕЧЕНИЕ КАЧЕСТВА И ТЕСТИРОВАНИЕ

Семинар 4: Модульное тестирование (xUnit)
практика

Москва, 2020



JUNIT НАЧАЛО РАБОТЫ

1. Подключить библиотеку JUnit
2. Импортировать необходимые классы аннотаций
3. Импортировать статические методов класса Assert



БАЗОВЫЕ МЕТОДЫ КЛАССА ASSERT

- `fail(String)` - Указывает на то, что бы тестовый метод завалился при этом выводя текстовое сообщение.
- `assertEquals` - проверяет, что два значения совпадают
- `assertArrayEquals` - сравнения содержимого массивов
- `assertTrue` - проверяет, что логическое условие истинно.
- `assertNull` - проверяет, что объект является пустым
- `assertSame` - проверяет, что обе переменные относятся к одному объекту



БАЗОВЫЕ АННОТАЦИИ

- @Test
- Обозначает тестовые методы, где размещаются сами проверки.

JUnit 4

- Методы должны быть `public void`. +параметры:
- `expected` — задает ожидаемое исключение
- `timeout` — задает время, по истечению которого тест считается провалившимся.

JUnit 5

- Методы должны быть `void`.

БАЗОВЫЕ АННОТАЦИИ

JUnit 4

- `@Before`
- Методы, которые будут вызваны до исполнения теста, методы должны быть `public void`.

JUnit 5

- `@BeforeEach`
- Методы, которые будут вызваны до исполнения теста, методы должны быть `void`

БАЗОВЫЕ АННОТАЦИИ

JUnit 4

- `@After`
- Методы, которые будут вызваны после выполнения теста, методы должны быть `public void`.

JUnit 5

- `@AfterEach`
- Методы, которые будут вызваны после выполнения теста, методы должны быть `void`.

БАЗОВЫЕ АННОТАЦИИ

JUnit 4

- `@BeforeClass`
- Методы, которые будут вызваны до создания экземпляра тест-класса, методы должны быть `public static void`.

JUnit 5

- `@BeforeAll`
- Методы, которые будут вызваны до создания экземпляра тест-класса, методы должны быть `static void`.



БАЗОВЫЕ АННОТАЦИИ

- Аннотации для отключения некоторого теста.
- Если поместить эту аннотацию на класс, то все тесты в этом классе будут отключены.

JUnit 4

`@Ignore`

JUnit 5

`@Disabled`



АННОТАЦИИ JUNIT5

- `@DisplayName("name")` - Объявляет отображаемое имя для тестового класса или метода

тестирования

- `@RepeatedTest(5)` - объявляет, что метод является тестовым шаблоном для повторяемого теста
- `@ParameterizedTest` - аннотирует параметризованный тест

<code>@ValueSource</code>	<code>@EnumSource</code>	<code>@MethodSource</code>	<code>@CsvSource</code>
short, byte, int, long, float, double, char, java.lang.String, java.lang.Class	Enum constants	аргументы возвращает фабричный метод	аргументы как список значений, разделенные запятыми



ПОКРЫТИЕ КОДА

- **Покрытие кода** — мера, используемая при тестировании программного обеспечения. Она показывает процент исходного кода программы, который был выполнен в процессе тестирования.



ПОКРЫТИЕ КОДА

- На основе структурных элементов тестируемой системы, которые выполняются или задействуются в ходе тестирования.
- На основе структуры входных данных, используемых во время тестирования.
- На основе элементов требований, проверяемых при выполнении тестов.
- На основе явно сформулированных предположений об ошибках, выявление которых должны обеспечить тесты.
- На основе произвольных моделей устройства или функционирования тестируемой системы



ИНСТРУМЕНТЫ

- Jcov
- JaCoCo
- Clover
- EMMA
- Serenity



ЛИТЕРАТУРА

1. Кент Бек – «Экстремальное программирование. Разработка через тестирование»
2. Gerard Meszaros – «xUnit Test Patterns»
3. JUnit – <https://junit.org/>
4. TestNG – <https://testng.org>
5. JaCoCo – <https://www.jacoco.org>
6. Цикломатическая сложность – https://ru.wikipedia.org/wiki/Цикломатическая_сложность

СПАСИБО! ВОПРОСЫ?



НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ