

Выполнил(а) Демидов И.А., № группы P3107, оценка \_\_\_\_\_  
Фамилия И.О. студента не заполнять

<b>JUnit: тестирование методов, вызывающих System.exit()</b>		
<b>ФИО автора статьи (или e-mail)</b> @rikki_tikki	<b>Дата публикации (не старше 2019 года)</b> "23" ноября 2022 г.	<b>Размер статьи (от 400 слов)</b> 947
<b>Прямая полная ссылка на источник или сокращённая ссылка (bit.ly, tr.im и т.п.)</b> <a href="https://habr.com/ru/company/otus/blog/701174/">https://habr.com/ru/company/otus/blog/701174/</a>		
<b>Теги, ключевые слова или словосочетания</b> java, junit, тестирование методов, enum, перечисления		
<b>Перечень фактов, упомянутых в статье</b> <ol style="list-style-type: none"> <li>1. В определенных ситуациях нам может потребоваться, чтобы метод вызывал System.exit() и завершал работу приложения.</li> <li>2. Если метод вызывает System.exit(), вызвать его из юнит-тестов и делать ассерты становится трудно, потому что это приведет к завершению юнит-теста.</li> <li>3. Чтобы избежать завершения юнит-теста, можно использовать Security Manager.</li> <li>4. Security Manager предотвратит вызовы System.exit(), а если вызов состоится, выбросит исключение. Затем можно перехватить выброшенное исключение, чтобы сделать ассерты.</li> <li>5. Также написать тест можно с использованием библиотеки System Lambda.</li> <li>6. Эта библиотека помогает тестировать код, вызывающий методы класса System.</li> <li>7. Фреймворк JMockit предоставляет возможность сделать мок класса System. Его можно использовать для того, чтобы изменить поведение System.exit() и предотвратить завершение работы приложения.</li> </ol>		
<b>Позитивные следствия и/или достоинства описанной в статье технологии (минимум три пункта)</b> <ol style="list-style-type: none"> <li>1. Рассматриваются несколько вариантов как тестировать методы, вызывающие System.exit().</li> <li>2. Автор приводит примеры кода, в котором реализован тот или иной вариант тестирования.</li> <li>3. Статья имеет удобную структуру, поэтому она достаточно легко читается и понимается.</li> </ol>		
<b>Негативные следствия и/или недостатки описанной в статье технологии (минимум три пункта)</b> <ol style="list-style-type: none"> <li>1. Несмотря на данные обходные пути, все равно сложно тестировать методы, вызывающие System.exit().</li> <li>2. Автор не объясняет, почему это важная тема и как часто встречаются подобные сложности в тестировании, поэтому не так интересно читать статью.</li> <li>3. Насколько я понял, есть более простые способы избежать трудности, но автор о них не упоминает.</li> </ol>		
<b>Ваши замечания, пожелания преподавателю или анекдот о программистах<sup>1</sup></b>		

<sup>1</sup> Наличие этой графы не влияет на оценку