

Евгений Гродно
36 уровень

30.04.2018 20055 22

JUnit part II

Статья из группы Random
1689440 участников

Вы в группе

продолжение
начало тут -> [JUnit part I](#)



Еще хочу вам показать, как работает метод `Assert.fail(String)` – если будет вызван этот метод, тест будет провален. Удобно в случае если другие методы класса `Assert` не могут сделать нам проверку. Допустим нам нужно запретить принимать `new String(“”)`. Попробуем это сделать на конструкторе запретим передавать `name = “” || null, age = 0; Sex = null`. Поехали ...

Я добавил дополнительные поля для тестирования

```
1 private User user;
2 private User user1;
3 private User user2;
4
5 private User userNotAdd;
6 private User userNotAdd1;
```

и изменил метод `setUp()`

```
1 @Before
```

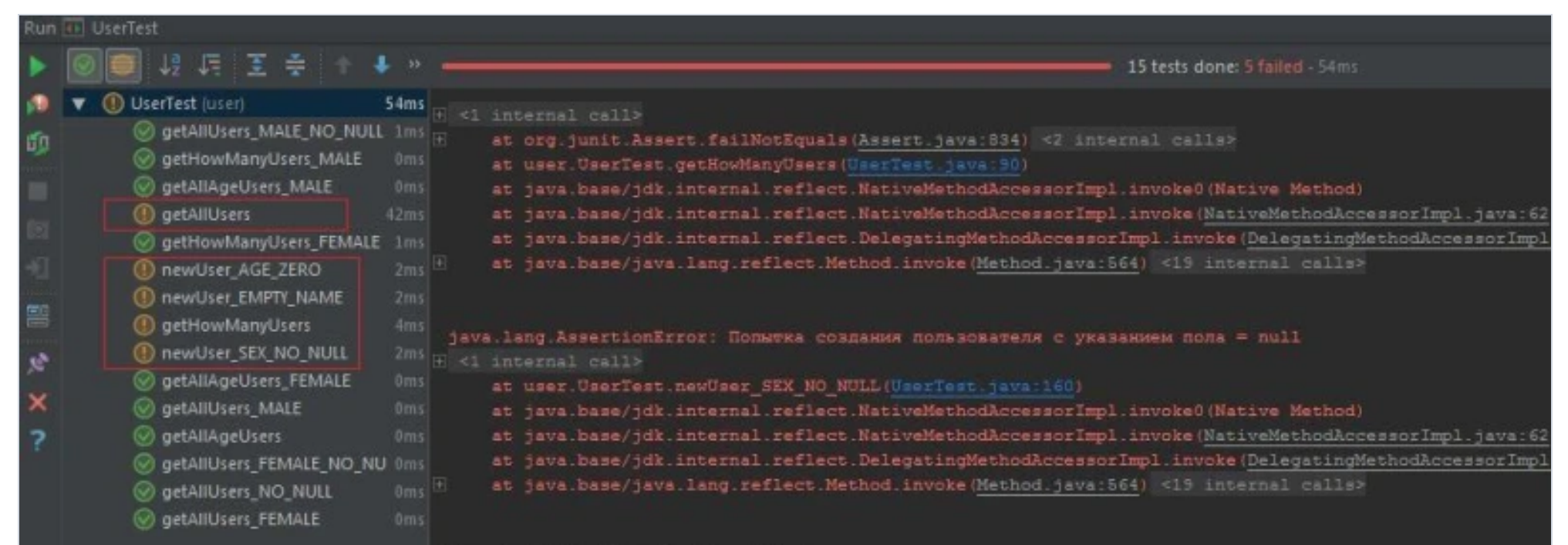
```
4      user1 = new User("Марина", 34, Sex.FEMALE);
5      user2 = new User("Алина", 7, Sex.FEMALE);
6
7      userNotAdd = new User("", 0, null);
8      userNotAdd1 = new User(null, 0, null);
9  }
```

и добавил три метода теста

```
1  @Test
2  public void newUser_EMPTY_NAME() {
3      for (User user : User.getAllUsers()){
4          if (user.getName() != null && user.getName().isEmpty()) {
5              Assert.fail("Попытка создания пользователя с пустым именем");
6          }
7      }
8  }
9
10 @Test
11 public void newUser_AGE_ZERO() {
12     for (User user : User.getAllUsers()) {
13         if (user.getAge() <= 0) {
14             Assert.fail("Попытка создания пользователя с не допустимым возрастом");
15         }
16     }
17 }
18
19 @Test
20 public void newUser_SEX_NO_NULL() {
21     for (User user : User.getAllUsers()) {
22         if (user.getSex() == null) {
23             Assert.fail("Попытка создания пользователя с указанием пола = null");
24         }
25     }
26 }
```

В тестах мы проходим по списку добавленных пользователей, и смотрим есть ли не допустимые данные.

Запускаем тесты оптом, и видим такую картину...

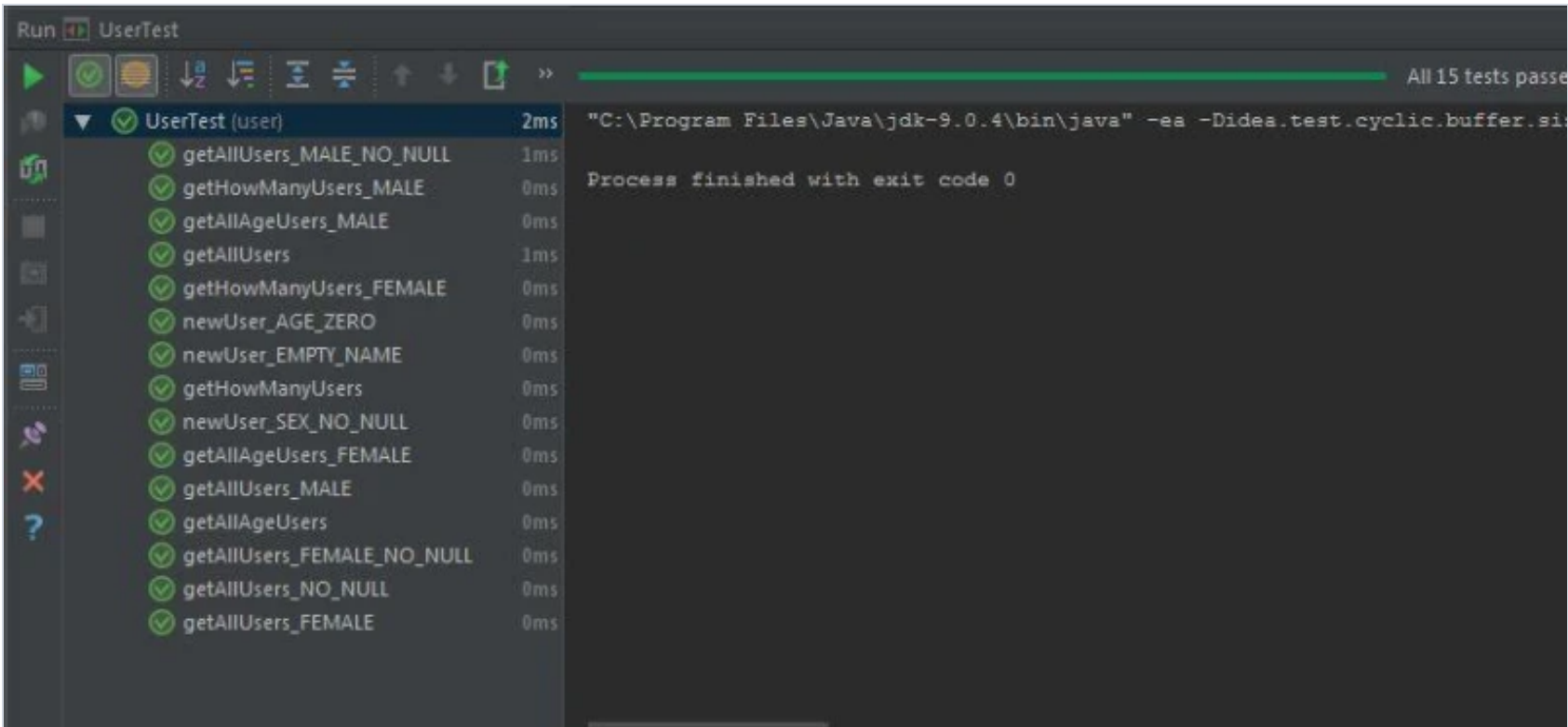


Тут видно, наши тесты для конструктора поломались, а так же зацепило еще пару тестов

Теперь нам нужно изменить конструктор, чтобы он не добавлял в список пользователей, пользователя с недопустимыми параметрами, сделаем это так:

```
1 public User(String name, int age, Sex sex) {
2     if (name != null && !name.isEmpty() && age > 0 && sex != null){
3         this.name = name;
4         this.age = age;
5         this.sex = sex;
6
7         if (!hasUser()) {
8             countId++;
9             this.id = countId;
10            allUsers.put(id, this);
11        }
12    }
13 }
```

Запускаем наши тесты и видим, что у нас все по красоте



Подведем итог, **JUnit** отличный инструмент для покрытия своего кода тестами. **JUnit** сработает еще лучше, в связке с **Maven**, при сборке проекта, **Maven** обязательно проведет все тесты, и потом соберет наш готовый проект, но если тесты будут провалены, проект не будет собран, и мы будем знать причины, и так будет легче разобраться где именно поломка.

Второй плюс **JUnit** это случай **Рефакторинга**, тесты **JUnit** помогут нам проследить чтобы логика нашей программы не поломалась. А в больших проектах без системы автоматического тестирования жизнь не возможна, хотя может и возможна, но двигаться ваш проект будет со скоростью черепахи ... или как то так ...

НАЧАТЬ ОБУЧЕНИЕ



Если мы хорошо покроем тестами наш код, то можем быть уверены, что у нашего клиента, наш продукт будет отлично работать.

P/S Выражаю огромную благодарность Сергееву Виктору!!!

Вот ссылки на видео урок, в котором он нас обучал:

- [Тестирование java](#)
- [Тестирование java \(продолжение\)](#)

Спасибо за внимание!!!
Хорошего вам настроения!!!

НАЧАТЬ ОБУЧЕНИЕ

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

▼

