

# Инструмент мутационного тестирования для языка Scala

---

---

Весенняя практика 2017  
@ JetBrains

---

Ялышев Антон  
Роман Шеин

# Основные понятия

1. Юнит-тест
2. Тестовое покрытие
3. Неполюценный тест
4. Мутационный подход ...

# **"Кто стережёт стражей?"**

(лат. *Quis custodiet ipsos custodes?*)

1. Внедрение дефектов
2. Прогон тестов
3. Ожидание падений тестов
4. Обзор не упавших

# Принцип мутации



1. Условные оп.	<b>if</b> (a == b)	<b>if</b> (a <b>!=</b> b)
2. Мат. операторы	<b>int a = b + c;</b>	<b>int a = b - c;</b>
3. Инкремент	<b>public void</b> foo() { <b>this.i</b> ++;	<b>public void</b> foo() { <b>this.i</b> --;
4. Константы	}	}
5. ...	<b>public static final int</b> CONST_1 = <b>25</b> ;	<b>public static final int</b> CONST_1 = <b>18</b> ;
	<b>public</b> Object foo() { <b>return new</b> Object(); }	<b>public</b> Object foo() { <b>new</b> Object(); <b>return null</b> ; }

# О работе над проектом ...

- Такое пока никто не сделал =)
- Алгоритмы и архитектура
  - когда вводить мутации
  - как обрабатывать результат
- Пригодятся знания/навыки:
  - работа с чужим кодом
  - компиляция и JVM
  - Java

# План работы

- Существующие фреймворки
- Как работает компиляция в Scala
- Спроектируем алгоритм
  - модификация [ исходников | байткода]
  - или плагин для компилятора
  - или ...
- Реализуем и интегрируем в Scala plugin

# Контакты и ссылки

- Почта: [anton.yalyshev@jetbrains.com](mailto:anton.yalyshev@jetbrains.com)
- Наиболее популярный МТ для Java: <http://pitest.org/>
- Scala plugin: <https://github.com/JetBrains/intellij-scala>

---

**Спасибо!**  
**Ваши вопросы ...**

---

---