

Live templates extractor plugin

Руководитель: Шеин Роман



Live templates


- Фича платформы IntelliJ IDEA
- Замена ключевого слова на заранее заданный темплейт
- Темплейты могут использовать специальные функции и переменные
- Могут создаваться пользователем



Live templates

```
public class LiveTemplates {  
    public static void main(String[] args) {
```


```
        sout|
```

```
sout Prints a string to System.out 
```

```
soutm Prints current class and method names to System.out
```

```
soutp Prints method parameter names and values to System...
```

```
soutv Prints a value to System.out
```

```
Press Ctrl+Period to choose the selected (or first) suggestion and insert a dot afterwards >> 
```



Internship 2016

Live templates

```
public class LiveTemplates {  
    public static void main(String[] args) {  
        System.out.println();  
    }  
}
```



Internship 2016

Live templates - проблемы

- Покрытие составлено эмпирически
- DSL, библиотеки, фреймворки нуждаются в своих наборах темплейтов
- Пользователь не хочет запариваться написанием темплейтов (а часто и вовсе об этом не думает)



Пример: ScalaTest

```
import org.scalatest.FeatureSpec
import org.scalatest.GivenWhenThen

class ScalaTestExample extends FeatureSpec with GivenWhenThen {
  feature("Mighty feature") {
    scenario("Some scenario") {
      Given("Some preset condition")
      When("Something happened")
      Then("Some reaction followed")
      assert(true)
    }
  }
}
```



Internship 2016

Решение проблем

1. Выкачивание кодовой базы с GitHub
2. Предобработка данных
3. Выделение из кодовой базы кандидатов в темплейты
4. Выбор пользователем актуальных темплейтов из найденных



Что уже есть

- Прототип алгоритма на Java, выделяющий из набора файлов кандидаты в темплейты
- Представление о конечном результате
- Заинтересованный в выходе работы в production
руководитель



Что необходимо сделать

- Доработать (или переписать) алгоритм поиска кандидатов в темплейты
- Реализовать плагин-экстрактор для Java, Scala, Kotlin и, если получится, C++
- Построить с помощью плагина набор темплейтов для нескольких фреймворков и библиотек



Что ещё можно успеть сделать

- Модифицировать алгоритм для релаксированного поиска дублированного кода
- Реализовать статический анализ кода с возможностью править «bad code practices» на лету
- Сервис для обмена темплейтами
- Что-нибудь ещё...



Спасибо за внимание.

