

Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт Космических и информационных технологий

институт

Кафедра «Информатика»

кафедра

ОТЧЕТ О ПРАКТИЧЕСКОЙ РАБОТЕ №3

Аннотирование в Spring

тема

Преподаватель

подпись, дата

А.С. Черниговский

инициалы, фамилия

Студент КИ18-17/16 031830504

номер группы, зачетной книжки

подпись, дата

Е.В. Железкин

инициалы, фамилия

Красноярск 2020

1 Цель работы

Познакомиться с аннотациями в Spring.

2 Задача работы

Взять практическое задание №2 и сконфигурировать контекст приложения без использования xml, а только при помощи аннотаций и java-файла конфигурации. Снабдить ранее спроектированные классы init и destroy методами, а также использовать фабричный метод для любого из классов.

3 Ход работы

1) Реализован интерфейс механизма (рисунок 1):

```
1 package ru.superzloyuser.Clocks;
2
3 public interface Mechanism
4 {
5     String getTime();
6     String getInfo();
7 }
```

Рисунок 1 – Интерфейс механизма, Mechanism.java

2) Реализованы классы, реализующие интерфейс (рисунки 2, 3):

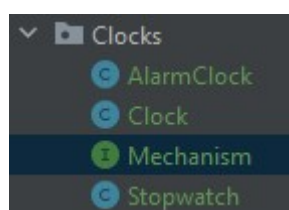


Рисунок 2 – Структура классов

```
1 package ru.superzloyuser.Clocks;
2
3 import java.util.Date;
4
5 public class AlarmClock implements Mechanism
6 {
7     private final String name;
8     private final int alarmCounter;
9     private final int seconds;
10
11     public AlarmClock(String name, int seconds, int alarmCounter)
12     {
13         this.name = name;
14         this.seconds = seconds;
15         this.alarmCounter = alarmCounter;
16     }
17
18     public String getName() { return name; }
19
20     public int getSeconds() { return seconds; }
21
22     public int getAlarmCounter() { return alarmCounter; }
23
24     @Override
25     public String getTime() {
26         return "Time: " + new Date() + "; Alarm setted at " +
27             alarmCounter / 3600 + ":" + alarmCounter % 3600 / 60 +
28             ":" + alarmCounter % 60;
29     }
30
31     @Override
32     public String getInfo() { return "Alarm Clock;"; }
33
34     @Override
35     public String toString() {
36         return "AlarmClock{" +
37             "name=" + name + ", seconds=" + seconds + ", alarmCounter=" + alarmCounter + "}"
38     }
39 }
```

Рисунок 3 – Один из классов – AlarmClock.java

3) Заполнен файл applicationContext.xml (внедрены зависимости бинов, рисунок 4):

```
<bean id="alarmClock" class="ru.superzloyuser.Clocks.AlarmClock">
    <constructor-arg name="name" value="AlarmClock"/>
    <constructor-arg name="seconds" value="10000"/>
    <constructor-arg name="alarmCounter" value="62789"/>
</bean>

<bean id="clock" class="ru.superzloyuser.Clocks.Clock">
    <constructor-arg name="name" value="AlarmClock"/>
    <constructor-arg name="seconds" value="48020"/>
</bean>

<context:property-placeholder location="classpath:someProperties.properties"/>
<bean id="stopwatch" class="ru.superzloyuser.Clocks.Stopwatch">
    <constructor-arg name="name" value="AlarmClock"/>
    <constructor-arg name="seconds" value="${savedTime}"/>
</bean>

<bean id="user1" class="ru.superzloyuser.MechanismUser">
    <constructor-arg name="name" value="User1"/>
    <constructor-arg name="mechanism" ref="alarmClock"/>
</bean>

<bean id="user2" class="ru.superzloyuser.MechanismUser">
    <constructor-arg name="name" value="User2"/>
    <constructor-arg name="mechanism" ref="clock"/>
</bean>

<bean id="user3" class="ru.superzloyuser.MechanismUser">
    <constructor-arg name="name" value="User3"/>
    <constructor-arg name="mechanism" ref="stopwatch"/>
</bean>
```

Рисунок 4 – Внедрение зависимостей

4) Запуск Spring приложения (Рисунок 5):

```
Person: User1; Clock - Alarm Clock;; Time: Thu Dec 03 15:09:43 KRAT 2020; Alarm setted at 17:26:29
Person: User2; Clock - Wrist Watch; Time: Thu Dec 03 20:20:20 KRAT 2020
Person: User3; Clock - Stopwatch; Seconds right now: 43

Total defined beans count: 7
alarmClock
clock
org.springframework.context.support.PropertySourcesPlaceholderConfigurer#0
stopwatch
user1
user2
user3

Process finished with exit code 0
```

Рисунок 5 – Результат выполнения

4 Вывод

В ходе данной лабораторной работы был изучен механизм внедрения зависимостей в Spring.