

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

Институт Космических и информационных технологий

институт

Кафедра «Информатика»

кафедра

**ОТЧЕТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ №1**

Внедрение зависимостей в Spring

тема

Преподаватель

Студент КИ18-166 031831229

номер группы, зачетной книжки

подпись, дата

подпись, дата

А.С. Черниговский

инициалы, фамилия

В.А. Прекель

инициалы, фамилия

Красноярск 2020

## **1 Цель работы**

Ознакомиться с механизмом внедрения зависимостей в Spring.

## **2 Общая постановка задачи**

В каждом варианте есть сущность (класс), необходимо создать интерфейс (самостоятельно на усмотрение студента) и классы его имплементирующие.

Объекты классов имплементирующих данный интерфейс будут передаваться в

качестве зависимостей. Выполнить связывание иполучить объекты из контекста.

В случае затруднений обратитесь к конспекту второй лекции курса.

Необходимо:

- 1) Реализовать внедрение простых значений через конструктор
- 2) Реализовать внедрение зависимости по ссылке через конструктор
- 3) Интерфейс должен содержать как минимум один метод
- 4) Классы, имплементирующие интерфейс должны содержать как минимум одно поле (у разных классов разные)
- 5) Зависимый класс должен содержать метод, который бы на основе вызова метода у зависимости выводил бы некоторое сообщение в консоль (Например для класса Автомобиля, в который внедряются Двигатели. Двигатели могут выдавать свою мощность, а автомобиль может выводить сообщение с какой скоростью он может двигаться.)
- 6) Реализовать внедрение простых значений из внешнего файла через setter

Вариант 11. ПК.

## **3 Исходный код**

Листинг 1 – `Lab02/src/main/java/com/github/prekel/JavaSpring/Lab02/Program.java`

```

package com.github.prekel.JavaSpring.Lab02;

import com.github.prekel.JavaSpring.Lab02.components.PersonalComputer;
import org.springframework.context.support.ClassPathXmlApplicationContext;

public class Program {
    public static void main(String[] args) {
        var context = new
ClassPathXmlApplicationContext("applicationContext.xml");
        var pc1 = context.getBean("pc1", PersonalComputer.class);
        var pc2 = context.getBean("pc2", PersonalComputer.class);
        var pc3 = context.getBean("pc3", PersonalComputer.class);
        var pc4 = context.getBean("pc4", PersonalComputer.class);
        System.out.println(pc1);
        System.out.println(pc2);
        System.out.println(pc3);
        System.out.println(pc4);

        System.out.println();

        for (String beanName : context.getBeanDefinitionNames()) {
            System.out.println(beanName);
        }

        context.close();
    }
}

```

Листинг

2

—

Lab02/src/main/java/com/github/prekel/JavaSpring/Lab02/components/Cpu.java

```

package com.github.prekel.JavaSpring.Lab02.components;

public interface Cpu {
    int getClock();
}

```

Листинг

3

—

Lab02/src/main/java/com/github/prekel/JavaSpring/Lab02/components/LockedCpu.java

va

```

package com.github.prekel.JavaSpring.Lab02.components;

public class LockedCpu implements Cpu {
    private final int clock;

    @Override
    public String toString() {
        return "LockedCpu{" +
            "clock=" + clock +
            '}';
    }
}

```

3

```

    }

    public LockedCpu(int clock) {
        this.clock = clock;
    }

    @Override
    public int getClock() {
        return clock;
    }
}

```

## Листинг

4

Lab02/src/main/java/com/github/prekel/JavaSpring/Lab02/components/UnlockedCpu.java

```

    package com.github.prekel.JavaSpring.Lab02.components;

    public class UnlockedCpu implements Cpu {
        private final int busClock;
        private int multiplier;

        @Override
        public String toString() {
            return "UnlockedCpu{" +
                "busClock=" + busClock +
                ", multiplier=" + multiplier +
                ", clock=" + getClock() +
                '}';
        }

        public UnlockedCpu(int busClock, int multiplier) {
            this.busClock = busClock;
            this.multiplier = multiplier;
        }

        public int getMultiplier() {
            return multiplier;
        }

        public void setMultiplier(int multiplier) {
            this.multiplier = multiplier;
        }

        @Override
        public int getClock() {
            return busClock * multiplier;
        }

        public int getBusClock() {
            return busClock;
        }
    }

```

**Lab02/src/main/java/com/github/prekel/JavaSpring/Lab02/components/PersonalComputer.java**

```
package com.github.prekel.JavaSpring.Lab02.components;

public class PersonalComputer {
    private final Cpu cpu;

    private String name = "Default PC name";

    @Override
    public String toString() {
        return "PersonalComputer{" +
            "cpu=" + cpu +
            ", name='" + name + '\'' +
            '}';
    }

    public PersonalComputer(Cpu cpu) {
        this.cpu = cpu;
    }

    public PersonalComputer(Cpu cpu, String name) {
        this.cpu = cpu;
        this.name = name;
    }

    public Cpu getCpu() {
        return cpu;
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }
}
```

**Листинг 6 – Lab02/src/main/resources/applicationContext.xml**

```
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:context="http://www.springframework.org/schema/context"
    xsi:schemaLocation="http://www.springframework.org/schema/beans
        http://www.springframework.org/schema/beans/spring-beans.xsd
        http://www.springframework.org/schema/context
        http://www.springframework.org/schema/context/spring-context.xsd">
    <bean id="cpu1"
        class="com.github.prekel.JavaSpring.Lab02.components.LockedCpu">
        <constructor-arg name="clock" value="3600"/>
    </bean>
</beans>
```

```

        </bean>
        <bean id="cpu2"
class="com.github.prekel.JavaSpring.Lab02.components.LockedCpu">
            <constructor-arg name="clock" value="2600"/>
        </bean>
        <bean id="cpu3"
class="com.github.prekel.JavaSpring.Lab02.components.UnlockedCpu">
            <constructor-arg name="busClock" value="100"/>
            <constructor-arg name="multiplier" value="40"/>
        </bean>
        <bean id="cpu4"
class="com.github.prekel.JavaSpring.Lab02.components.UnlockedCpu">
            <constructor-arg name="busClock" value="110"/>
            <constructor-arg name="multiplier" value="20"/>
        </bean>
        <bean id="pc1"
class="com.github.prekel.JavaSpring.Lab02.components.PersonalComputer">
            <constructor-arg name="cpu" ref="cpu1"/>
            <constructor-arg name="name" value="Pc with cpu1"/>
        </bean>
        <bean id="pc2"
class="com.github.prekel.JavaSpring.Lab02.components.PersonalComputer">
            <constructor-arg name="cpu" ref="cpu2"/>
            <constructor-arg name="name" value="Pc with cpu2"/>
        </bean>
        <bean id="pc3"
class="com.github.prekel.JavaSpring.Lab02.components.PersonalComputer">
            <constructor-arg name="cpu" ref="cpu3"/>
            <property name="name" value="Pc with cpu3"/>
        </bean>
        <context:property-placeholder location="classpath:pc4.properties"/>
        <bean id="pc4"
class="com.github.prekel.JavaSpring.Lab02.components.PersonalComputer">
            <constructor-arg name="cpu" ref="cpu4"/>
            <property name="name" value="{pc4.name}"/>
        </bean>
    </beans>

```

## Листинг 7 – Lab02/src/main/resources/pc4.properties

```
pc4.name=Pc with cpu4
```