Федеральное государственное автономное образовательное учреждение высшего профессионального образования

«Университет ИТМО»

**ОТЧЕТ**

по лабораторной работе

# «Абстрактные классы и интерфейсы»

Выполнил:

Студент гр.K33204

Мороз Илья Олегович

Проверил:

Тьютор

Дубаков Анатолий Алексеевич

Санкт-Петербург

2020 г.

**Homework #3.1 – Абстрактные классы**

В качестве первого задания был выбран вариант 1. Необходимо создать абстрактный класс Player и создать несколько классов-наследников, которые будут наследоваться от него.

**Player.java**

public abstract class Player {  
 protected String key;  
  
 public Player() {  
 key = "До мажор";  
 }  
  
 public Player(String key) {  
 this.key = key;  
 }  
  
 public abstract void play();  
}

Теперь создадим классы-наследники.

**Guitar.java**

public class Guitar extends Player {  
 protected int numberOfString;  
  
 public Guitar() {  
 numberOfString = 6;  
 }  
  
 public Guitar(int numberOfString) {  
 this.numberOfString = numberOfString;  
 }  
  
 public Guitar(String key, int numberOfString) {  
 super(key);  
 this.numberOfString = numberOfString;  
 }  
  
 @Override  
 public void play() {  
 System.*out*.println("Играет гитара с кол-вом струн " + numberOfString + " и нотой " + key);  
 }  
}

**Drum.java**

public class Drum extends Player {  
 protected double size;  
  
 public Drum() {  
 size = 3.0;  
 }  
  
 public Drum(double size) {  
 this.size = size;  
 }  
  
 public Drum(String key, double size) {  
 super(key);  
 this.size = size;  
 }  
  
 @Override  
 public void play() {  
 System.*out*.println("Играет барабан размером " + size + " и нотой " + key);  
 }  
}

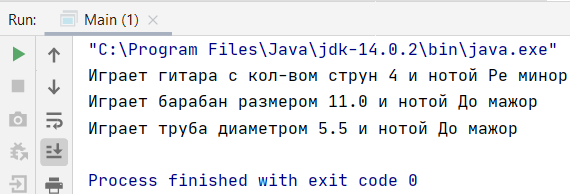
**Trumpet.java**

public class Trumpet extends Player {  
 protected double diameter;  
  
 public Trumpet() {  
 diameter = 3.0;  
 }  
  
 public Trumpet(double diameter) {  
 this.diameter = diameter;  
 }  
  
 public Trumpet(String key, double diameter) {  
 super(key);  
 this.diameter = diameter;  
 }  
  
 @Override  
 public void play() {  
 System.*out*.println("Играет труба диаметром " + diameter + " и нотой " + key);  
 }  
}

**Main.java**

public class Main {  
 public static void main(String[] args) {  
 Player players[] = new Player[3];  
 players[0] = new Guitar("Ре минор", 4);  
 players[1] = new Drum(11.0);  
 players[2] = new Trumpet(5.5);  
  
 for (int i = 0; i < players.length; i++) {  
 players[i].play();  
 }  
 }  
}

**Console**

****

**Homework #3.2 – Интерфейсы**

В качестве второго задания был выбран вариант 1. Необходимо реализовать интерфейс IntSequence и с его помощью написать программу, в которой пользователю предлагается ввод каждого следующего числа.

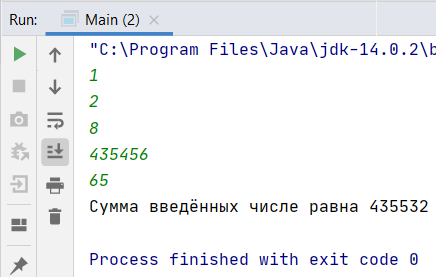
**IntSequence.java**

public interface IntSequence {  
 boolean hasNext();  
  
 int next();  
}

**Main.java**

import java.util.Scanner;  
  
public class Main implements IntSequence {  
 @Override  
 public boolean hasNext() {  
 return true;  
 }  
  
 @Override  
 public int next() {  
 Scanner sc = new Scanner(System.*in*);  
 return sc.nextInt();  
 }  
  
 public static void main(String[] args) {  
 Main main = new Main();  
 int sum = 0;  
  
 for (int i = 0; i < 5; i++) {  
 sum += main.next();  
 }  
  
 System.*out*.println("Сумма введённых числе равна " + sum);  
 }  
}

**Console**

****