МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ

РОССИЙСКОЙ ФЕДЕРАЦИИ

ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

**Кафедра ''САПР''**

Отчет о выполненной лабораторной работе № 4

по дисциплине: “Объектно-ориентированное программирование”

Выполнили ст.гр. 20ВА1:

Комаров М.В.

Пачков А.В.

Принял:

Подмарькова Е.М.

**Цель работы:**

Изучить возможности виртуальных функций при наследовании классов на языке Java, абстрактные классы.

**Ход работы:**

Задание:



Код программы:

**Location.java:**

public abstract class Location  
{  
 String location = "Russia";  
 Location(String location)  
 {  
 // this.location = location;  
 }  
 abstract public String getCurrentLocation();  
 abstract public String getParrentLocation();  
 abstract public String getInfo();  
}

**Area.java:**

public class Area extends Location  
{  
 String area = "Moscow region";  
 Area (String area)  
 {  
 super(area);  
 // this.area = area;  
 }  
 @Override  
 public String getCurrentLocation()  
 {  
 return area;  
 }  
 @Override  
 public String getParrentLocation()  
 {  
 return super.location;  
 }  
 @Override  
 public String getInfo()  
 {  
 return "Current is: " + this.getCurrentLocation() + ". Parrent is: " + this.getParrentLocation();  
 }  
  
}

**Megapolis.java:**

public class Megapolis extends Area  
{  
 String megapolis = "Moscow";  
 Megapolis (String megapolis)  
 {  
 super(megapolis);  
 //this.megapolis = megapolis;  
 }  
 @Override  
 public String getCurrentLocation()  
 {  
 return megapolis;  
 }  
 @Override  
 public String getParrentLocation()  
 {  
 return super.area;  
 }  
 @Override  
 public String getInfo()  
 {  
 return "Current is: " + this.getCurrentLocation() + ". Parrent is: " + this.getParrentLocation();  
 }  
}

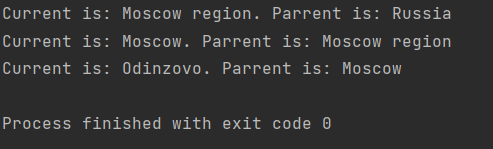
**City.java:**

public class City extends Megapolis  
{  
 String city = "Odinzovo";  
 City(String city)  
 {  
 super(city);  
 //this.city = city;  
 }  
 @Override  
 public String getCurrentLocation()  
 {  
 return city;  
 }  
 @Override  
 public String getParrentLocation()  
 {  
 return super.megapolis;  
 }  
 @Override  
 public String getInfo()  
 {  
 return "Current is: " + this.getCurrentLocation() + ". Parrent is: " + this.getParrentLocation();  
 }  
}

**Main.java:**

public class Main  
{  
 public static void main(String[] argc )  
 {  
 Location[] items = new Location[3];  
 items[0] = new Area("Moscow region");  
 items[1] = new Megapolis("Moscow");  
 items[2] = new City("Odinzovo");  
 for (Location item : items)  
 {  
 System.*out*.println(item.getInfo());  
 }  
 }  
}

Результат:



**Вывод:**

Мы изучили возможности виртуальных функций при наследовании классов на языке Java, изучили абстрактные классы.