|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ **ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ**

КАФЕДРА **КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ (ИУ6)**

НАПРАВЛЕНИЕ ПОДГОТОВКИ **09.04.01 Информатика и вычислительная техника**

**Отчет по лабораторной работе 4**

***по дисциплине «Языки программирования для работы с большими данными»***

Студент ИУ6-21М **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Гришина А. М.**

(Группа) (Подпись, дата) (И.О.Фамилия)

Руководитель **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Степанов П. В.**

(Подпись, дата) (И.О.Фамилия)

*2020 г.*

**Задание 1:**

**Вариант 1:** Создать класс City (город) с внутренним классом, с помощью объектов которого можно хранить информацию о проспектах, улицах, площадях.

**Код программы:**

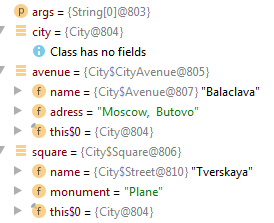
Файл Main.java:

package com.company;  
  
public class Main {  
  
 public static void main(String[] args) { *//Variant 1* City city = new City();  
 City.CityAvenue avenue = city.new CityAvenue(City.Avenue.*Balaclava*, "Moscow, Butovo");  
 City.Square square = city.new Square(City.Street.*Tverskaya*, "Plane");  
 }  
}

Файл City.java:

package com.company;  
import java.util.Date;  
  
public class City {  
 public enum Avenue{  
 *Balaclava*,  
 *Budyonny*,  
 *Vernadsky* }  
 public enum Street{  
 *Tverskaya*,  
 *Taganskaya*,  
 *Kantemirovskaya* }  
  
 public class CityAvenue{  
 public Avenue name;  
 public String adress;  
  
 public CityAvenue(){}  
 public CityAvenue(Avenue city, String adress){  
 this.name = city;  
 this.adress = adress;  
 }  
 }  
 public class Square{  
 public Street name;  
 public String monument;  
  
 public Square(){}  
 public Square(Street name, String monument){  
 this.name = name;  
 this.monument = monument;  
 }  
 }  
  
 public City(){}  
}

Проверка правильности работы:



**Вывод:** Был создан класс, который хранит информацию в классах, созданных в нем. Программа работает корректно.

**Задание 2:**

**Вариант 8:** Реализовать абстрактные классы или интерфейсы, а также наследование и полиморфизм для следующих классов nterface Корабль <- class Грузовой Корабль <- class Танкер.

**Код программы:**

Файл Main.java:

package com.company;  
  
public class Main {  
  
 public static void main(String[] args) { *//variant 8* Tanker bigTheater = new Tanker("Black Pearl", "Rum", 400,  
 100, "Moscow");  
 bigTheater.Raid();  
 bigTheater.change\_allowed("coast of Malta");  
 bigTheater.Raid();  
 bigTheater.Raid();  
 }  
}

Файл Ship.java:

package com.company;  
  
public interface Ship {  
 void allowed();  
 void prohibited();  
}

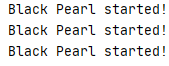
Файл CargoShip.java:

package com.company;  
  
public abstract class CargoShip implements Ship {  
 private int width;  
 private int heidth;  
 private String cargo;  
  
 public CargoShip(int width, int heidth, String cargo){  
 this.width = width;  
 this.heidth = heidth;  
 this.cargo = cargo;  
 }  
  
 public void Raid(){  
 if(this.cargo != "danger"){  
 allowed();  
 } else {  
 prohibited();  
 }  
 }  
 @Override  
 public void allowed(){  
 System.*out*.println("Sea access allowed!");  
 }  
 @Override  
 public void prohibited(){  
 System.*out*.println("Sea access prohibited!");  
 }  
  
 }

Файл Tanker.java:

package com.company;  
  
public class Tanker extends CargoShip {  
 public String name;  
 public String cargo;  
  
 public Tanker(String name, String cargo, int width, int heidth, String Raid){  
 super(width, heidth, cargo);  
 this.name = name;  
 }  
  
 @Override  
 public void allowed(){  
 System.*out*.println(this.name + " started!");  
 }  
  
 public void change\_allowed(String departure){  
 this.cargo = departure;  
 }  
}

Проверка правильности работы:



**Вывод:** Для реализации класса Корабль, был создан интерфейс Танкер и абстрактный класс Грузовой Корабль. Программа работает корректно.