|  |
| --- |
|  |
| МИНОБРНАУКИ РОССИИ |
| *Федеральное государственное бюджетное образовательное учреждение высшего образования* ***«МИРЭА – Российский технологический университет»***  **РТУ МИРЭА** |

**Институт информационных технологий (ИТ)**

**Кафедра инструментального и прикладного программного обеспечения (ИиППО)**

**Дисциплина «Программирование на языке Джава»**

**ОТЧЕТ**

**ПО ПРАКТИЧЕСКОМУ ЗАНЯТИЮ №7**

Выполнил студент группы ИВБО-07-19 Гудаев И.И.

Принял Степанин П.С.

Практические работы выполнены «\_\_\_»\_\_\_\_\_\_\_2020г.

«\_\_\_\_\_\_\_\_\_\_\_\_\_\_» «\_\_\_»\_\_\_\_\_\_\_2020г.

Отметка о выполнении

**Москва – 2020 г.**

## **Задание**

1. Создать абстрактный класс, описывающий посуду(Dish). С помощью наследования реализовать различные виды посуды, имеющие свои свойства и методы. Протестировать работу классов.  
2. Создать абстрактный класс, описывающий собак(Dog). С помощью наследования реализовать различные породы собак. Протестировать работу классов.  
3. Создать абстрактный класс, описывающий мебель. С помощью наследования реализовать различные виды мебели. Также создать класс FurnitureShop, моделирующий магазин мебели.

4.Протестировать работу всех классов.

## **Ход Работы**

В ходе выполнения работы были получены следующие исходные коды:

**package** com.company;

**public** **abstract** **class** Dish {

**protected** boolean isClean, isBroken, isFull;

**protected** int depth;

**protected** **String** name;

**public** boolean getIsClean() {

**return** isClean;

}

**public** boolean getIsBroken() {

**return** isBroken;

}

**public** void setBroken(boolean broken) {

isBroken = broken;

}

**public** boolean isFull() {

**return** isFull;

}

**public** void setFull(boolean full) {

isFull = full;

}

**public** void setClean(boolean clean) {

**this**.isClean = clean;

}

**public** int getDepth() {

**return** depth;

}

**public** void setDepth(int depth) {

**this**.depth = depth;

}

**public** **String** getName() {

**return** name;

}

Листинг 1.1. Задание 1 (класс Dish)

**public** void setName(**String** name) {

**this**.name = name;

}

void broke()

{

isBroken = **true**;

}

void repair()

{

isBroken = **false**;

}

**public** **abstract** void fill(**String** F);

}

Листинг 1.2. Задание 1 (класс Dish)

**package** com.company;

**public** **class** Bowl **extends** Dish{

**private** **String** soupName;

**public** **String** getSoupName() {

**return** soupName;

}

**public** void setSoupName(**String** soupName) {

**this**.soupName = soupName;

}

**public** void eat()

{

isClean = **false**;

isFull = **false**;

}

**@Override**

**public** void fill(**String** F) {

soupName = F;

isClean = **false**;

}

**@Override**

**public** **String** toString() {

**return** "Bowl{" +

"soupName='" + soupName + '\'' +

", isClean=" + isClean +

", isBroken=" + isBroken +

", isFull=" + isFull +

", depth=" + depth +

", name='" + name + '\'' +

'}';

}

}

Листинг 2. Задание 1 (класс Bowl)

**package** com.company;

**public** **class** Cup **extends** Dish {

**private** **String** drinkName;

**public** **String** getDrinkName() {

**return** drinkName;

}

**public** void setDrinkName(**String** drinkName) {

**this**.drinkName = drinkName;

}

**public** void drink()

{

isClean = **false**;

isFull = **false**;

}

**@Override**

**public** void fill(**String** F) {

drinkName = F;

isClean = **false**;

}

**@Override**

**public** **String** toString() {

**return** "Cup{" +

"drinkName='" + drinkName + '\'' +

", isClean=" + isClean +

", isBroken=" + isBroken +

", isFull=" + isFull +

", depth=" + depth +

", name='" + name + '\'' +

'}';

}

}

Листинг 3. Задание 1 (класс Cup)

**package** com.company;

**public** **class** Main {

**public** **static** void main(**String**[] args) {

// write your code here

Bowl bowl = **new** Bowl();

Cup cup = **new** Cup();

cup.fill("Кефир");

bowl.fill("Борщ");

**System**.out.println("�' кружку налит напиток: " + cup.getDrinkName());

**System**.out.println("�' глубокую тарелку налит суп: " + bowl.getSoupName());

cup.drink();

bowl.eat();

**System**.out.println(cup.toString());

**System**.out.println(bowl.toString());

}

}

Листинг 4. Задание 1 (класс Main)

**package** com.company;

**public** **abstract** **class** Dog {

**protected** java.lang.**String** name, breed, color, size;

**protected** int age;

void Dog(java.lang.**String** **Name**, java.lang.**String** Breed, int Age)

{

name = **Name**;

breed = Breed;

age = Age;

}

**public** **String** getName() {

**return** name;

}

**public** void setName(**String** name) {

**this**.name = name;

}

**public** **String** getBreed() {

**return** breed;

}

**public** int getAge() {

**return** age;

}

**public** void bark()

{

**System**.out.println(name + " залаяла");

}

**public** void eat()

{

**System**.out.println(name + " покушала");

}

**public** void bite(**String** b)

{

**System**.out.println(name + " укусила " + b);

}

}

Листинг 5. Задание 2 (класс Dog)

**package** com.company;

**public** **class** Shepherd **extends** Dog{

Shepherd(java.lang.**String** **Name**, java.lang.**String** Breed, int Age)

{

**super**.Dog(**Name**, Breed, Age);

color = "Коричневый";

size = "Средний";

}

}

Листинг 6. Задание 2 (класс Shepherd)

**package** com.company;

**public** **class** ChiHuaHua **extends** Dog{

ChiHuaHua(java.lang.**String** **Name**, java.lang.**String** Breed, int Age, **String** Color)

{

**super**.Dog(**Name**, Breed, Age);

color = Color;

size = "Small";

}

}

Листинг 7. Задание 2 (класс ChiHuaHua)

**package** com.company;

**public** **class** Main {

**public** **static** void main(**String**[] args) {

// write your code here

Shepherd bobik = **new** Shepherd("�'обик", "Овчарка", 5);

ChiHuaHua krisa = **new** ChiHuaHua("Крыса", "ЧиХуаХуа", 1, "Коричневый");

bobik.bark();

krisa.bark();

krisa.bite(bobik.getName());

bobik.bite(krisa.getName());

}

}

Листинг 8. Задание 2 (класс Main)

**package** com.company;

**public** **abstract** **class** Furniture {

**protected** **String** name;

**protected** int cost;

**public** **String** getName() {

**return** name;

}

**public** void setName(**String** name) {

**this**.name = name;

}

**public** int getCost() {

**return** cost;

}

**public** void setCost(int cost) {

**this**.cost = cost;

}

}

Листинг 9. Задание 3 (класс Furniture)

**package** com.company;

**public** **class** Chair **extends** Furniture{

Chair()

{

**this**.cost = 5000;

**this**.name = "Стул";

}

}

Листинг 10. Задание 3 (класс Chair)

**package** com.company;

**public** **class** Sofa **extends** Furniture{

Sofa()

{

**this**.cost = 50000;

**this**.name = “Диван"***;***

}

}

Листинг 11. Задание 3 (класс Sofa)

**package** com.company;

**import** java.util.ArrayList;

**public** **class** FurnitureShop {

**ArrayList**<Sofa> sofaList = **new** **ArrayList**<Sofa>();

**ArrayList**<Chair> chairList = **new** **ArrayList**<Chair>();

FurnitureShop()

{

**for**(int i = 0; i < 5; i++)

{

Sofa sofa = **new** Sofa();

sofaList.add(sofa);

Chair chair = **new** Chair();

chairList.add(chair);

}

}

**public** void addGoods(int S, int C)

{

**for**(int i = 0; i < S; i++)

{

Sofa sofa = **new** Sofa();

sofaList.add(sofa);

**System**.out.println("�"иванов добавлено - " + S)***;***

***}***

**for**(int i = 0; i < C; i++)

{

Chair chair = **new** Chair();

chairList.add(chair);

**System**.out.println("Кресел добавлено - " + C);

}

***}***

Листинг 12.1. Задание 3 (класс FurnitureShop)

**public** void sell(**String** Good)

{

**if**(Good == "Диван"***)***

{

**if**(sofaList.size() != 0)

{

**System**.out.println(Good + " продан");

sofaList.remove(sofaList.size() - 1);

**System**.out.println("Осталось товара " + Good + " " + sofaList.size() + " штук");

}

**else**

{

**System**.out.println(Good + " отсутствует!");

}

}

**else** **if**(Good == "Кресло")

{

**if**(chairList.size() != 0)

{

**System**.out.println(Good + " продан");

chairList.remove(chairList.size() - 1);

**System**.out.println("Осталось товара " + Good + " " + chairList.size() + " штук");

}

**else**

{

**System**.out.println(Good + " отсутствует!");

}

}

***}***

***}***

Листинг 12.2. Задание 3 (класс FurnitureShop)

**package** com.company;

**public** **class** Main {

**public** **static** void main(**String**[] args) {

// write your code here

FurnitureShop shop = **new** FurnitureShop();

shop.sell("�"иван")***;***

shop.sell("Кресло");

shop.addGoods(1, 1);

**for**(int i = 0; i < 6; i++)

{

shop.sell("�"иван")***;***

shop.sell("Кресло");

***}***

***}***

***}***

Листинг 13. Задание 3 (класс Main)

## **Вывод**

В каждом из заданий мы создали абстрактные классы и их наследников, составили для них методы и поля и протестировали работу этих класов.