

Домашнее задание по теме
«Понятия ООП: абстракция, инкапсуляция»

Формулировка задания:

1. Создать классы Покупатель (Person) и Продукт (Product).

Характеристики Покупателя: имя, сумма денег и пакет с продуктами (массив объектов типа Продукт). Имя не может быть пустой строкой. Деньги не могут быть отрицательным числом.

Если Покупатель может позволить себе Продукт, то Продукт добавляется в пакет. Если у Покупателя недостаточно денег, то добавление не происходит.

Характеристики Продукта: название и стоимость. Название продукта не может быть пустой строкой, оно должно быть. Стоимость продукта не может быть отрицательным числом.

2. Поля в классах должны быть private, доступ к полям осуществляется через геттеры и сеттеры или конструктор класса.

3. В классах переопределены методы toString(), equals(), hashCode().

4. Создать в классе App метод main и проверить работу приложения. Данные Покупателей и Продукты вводятся с клавиатуры или задаются случайным образом. Продукты в цикле выбираются покупателями по очереди и, пока не введено слово END, наполняется пакет.

5. Обработать следующие ситуации:

- а. Если покупатель не может позволить себе продукт, то напечатайте соответствующее сообщение ("[Имя человека] не может позволить себе [Название продукта]").

- б. Если ничего не куплено, выведите имя человека, за которым следует "Ничего не куплено".

в. В случае неверного ввода (сообщение об исключении: "Деньги не могут быть отрицательными") или пустого имени: (сообщение об исключении: "Имя не может быть пустым").

6. Программа реализуется в отдельной ветке `git homeworks/homework06`. При сохранении состояния программы (коммиты) пишется сообщение с

описанием хода работы по задаче.

В корне папки с программой должен быть файл `.gitignore`.

Программа локально коммитится и публикуется в репозиторий GitHub на проверку.

```
1 package homeworks.homework06;
2
3 import java.util.Arrays;
4 import java.util.Objects;
5
6 public class Person { 7 usages 2 programakc
7     private String name; 10 usages
8     private Double money; 9 usages
9     private Product[] products = new Product[0]; 11 usages
10
11     public Person(String name, Double money) { 1 usage 2 programakc
12         this.name = name;
13         this.money = money;
14     }
15
16     public String getName() { 1 usage 2 programakc
17         return name;
18     }
19
20     public Double getMoney() { no usages 2 programakc
21         return money;
22     }
23
24     public Product[] getProducts() { no usages 2 programakc
25         return products;
26     }
```

```
27
28 @ public void setName(String name) { 1 usage 2 programакс
29     if (!name.isEmpty()) {
30         this.name = name;
31     } else {
32         System.out.println("Имя не может быть пустым");
33     }
34 }
35
36 public void setMoney(Double money) { 1 usage 2 programакс
37     if (money < 0) {
38         System.out.println("Деньги не могут быть отрицательными");
39     } else {
40         this.money = money;
41     }
42 }
43
44 @ public void addProducts(Product product) { 1 usage 2 programакс
45     if (this.money >= product.getCost()) {
46         products = Arrays.copyOf(products, products.length + 1);
47         products[products.length - 1] = product;
48         this.money = this.money - product.getCost();
49         System.out.println(this.name + " купил(-а) " + product.getTitle());
50     } else {
51         System.out.println(this.name + " не может позволить себе " +
52             product.getTitle());
53     }
54 }
```

```
55
56 @Override @programakc
57 public boolean equals(Object o) {
58     if (o == null || getClass() != o.getClass()) return false;
59     Person person = (Person) o;
60     return Objects.equals(name, person.name) && Objects.equals(money,
61         person.money) && Objects.deepEquals(products, person.products);
62 }
63
64 @Override @programakc
65 public int hashCode() {
66     return Objects.hash(name, money, Arrays.hashCode(products));
67 }
68
69 @Override @programakc
70 public String toString() {
71     if (products.length == 0) {
72         return name + " - Ничего не куплено";
73     } else {
74         return name + " - " + Arrays.toString(products);
75     }
76 }
77 }
```

```
1 package homeworks.homework06;
2
3 import java.util.Objects;
4
5 public class Product { 10 usages  ⓘ programakc
6     private String title; 7 usages
7     private Double cost; 6 usages
8
9     public Product(String title, Double cost) { 1 usage ⓘ programakc
10         this.title = title;
11         this.cost = cost;
12     }
13
14     public String getTitle() { 3 usages ⓘ programakc
15         return title;
16     }
17
18     public Double getCost() { 2 usages ⓘ programakc
19         return cost;
20     }
21
22 @ public void setTitle(String title) { 1 usage ⓘ programakc
23     if (!title.isEmpty()) {
24         this.title = title;
25     }
26 }
27
```



```
28 public void setCost(Double cost) { 1 usage 2 programakc
29     if (cost > 0) {
30         this.cost = cost;
31     }
32 }
33
34 @Override 2 programakc
35 3 public boolean equals(Object o) {
36     if (o == null || getClass() != o.getClass()) return false;
37     Product product = (Product) o;
38     return Double.compare(cost, product.cost) == 0 && Objects.equals(title,
39         product.title);
40 }
41
42 @Override 2 programakc
43 3 public int hashCode() {
44     return Objects.hash(title, cost);
45 }
46
47 @Override 2 programakc
48 3 public String toString() {
49     return title;
50 }
51 }
```

```

1 package homeworks.homework06;
2
3 import java.util.Scanner;
4
5 public class App {
6     public static void main(String[] args) {
7
8         Scanner scanner = new Scanner(System.in);
9
10        String[] buyers = scanner.nextLine().split(";");
11        Person[] person = new Person[buyers.length];
12
13        for (int i = 0; i < buyers.length; i++) {
14            person[i] = new Person("", (double)0);
15            person[i].setName(buyers[i].split("=")[0].trim());
16            person[i].setMoney(Double.parseDouble(buyers[i].split("=")[1].trim()));
17        }
18        String[] productCost = scanner.nextLine().split(";");
19        Product[] product = new Product[productCost.length];
20
21        for (int i = 0; i < productCost.length; i++) {
22            product[i] = new Product("", (double)0);
23            product[i].setTitle(productCost[i].split("=")[0].trim());
24            product[i].setCost(Double.parseDouble(productCost[i].split("=")[1].trim()));
25        }

```



```
26
27     while (true) {
28         String str = scanner.nextLine();
29
30         if (str.equals("END")) {
31             break;
32         } else {
33             for (Person a: person) {
34                 if (str.split("-")[0].trim().equals(a.getName())) {
35                     for (Product b: product) {
36                         if (str.split("-")[1].trim().equals(b.getTitle())) {
37                             a.addProducts(b);
38                         }
39                     }
40                 }
41             }
42         }
43     }
44
45     for (Person p: person) {
46         System.out.println(p.toString().replace("[", "").replace("]", ""));
47     }
48
49 }
50 }
```

```
"C:\Program Files\Java\jdk-21.0.2\bin\java.exe" -javaagent:G:\JetBrains\IntelliJ\Idea2025.1\lib\idea_rt
.jar=62577 -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8 -classpath
D:\JavaProject\JavaDevHomeworks\out\production\JavaDevHomeworks homeworks.homework06.App

Павел Андреевич = 10000; Анна Петровна = 2000; Борис = 10
Хлеб = 40; Молоко = 60; Торт = 1000; Кофе растворимый = 879; Масло= 150;
Павел Андреевич - Хлеб
Павел Андреевич купил(-а) Хлеб
Павел Андреевич - Масло
Павел Андреевич купил(-а) Масло
Анна Петровна - Кофе растворимый
Анна Петровна купил(-а) Кофе растворимый
Анна Петровна - Молоко
Анна Петровна купил(-а) Молоко
Анна Петровна - Молоко
Анна Петровна купил(-а) Молоко
Анна Петровна - Молоко
Анна Петровна купил(-а) Молоко
Анна Петровна - Торт
Анна Петровна не может позолить себе Торт
Борис - Торт
Борис не может позолить себе Торт
Павел Андреевич - Торт
Павел Андреевич купил(-а) Торт
END
Павел Андреевич - Хлеб, Масло, Торт
Анна Петровна - Кофе растворимый, Молоко, Молоко, Молоко
Борис - Ничего не куплено
```

```
"C:\Program Files\Java\jdk-21.0.2\bin\java.exe" -javaagent:G:\JetBrains\IntelliJ\Idea2025.1\lib\idea_rt
.jar=63136 -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8 -classpath
D:\JavaProject\JavaDevHomeworks\out\production\JavaDevHomeworks homeworks.homework06.App
Женя = 0;
Мороженое = 200
Женя - Мороженое
Женя не может позолить себе Мороженое
END
Женя - Ничего не куплено
```

```
"C:\Program Files\Java\jdk-21.0.2\bin\java.exe" -javaagent:G:\JetBrains\IntelliJ\Idea2025.1\lib\idea_rt
.jar=63623 -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8 -classpath
D:\JavaProject\JavaDevHomeworks\out\production\JavaDevHomeworks homeworks.homework06.App
```

Света = -3

Деньги не могут быть отрицательными

Макароны = 800;

Света - Макароны

Света не может позолить себе Макароны

END

Света - Ничего не куплено