**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«Санкт-Петербургский национальный исследовательский университет  
 информационных технологий, механики и оптики»**

**Факультет:** «Информационных технологий и программирования»

**Кафедра:** «Информационных систем»

**Дисциплина:**

«Технологии программирования»

**Отчет по лабораторной работе №2**

**Выполнил студент группы № M3207:**

*Щагина Анастасия Сергеевна*

**Проверил:**   
*Рябчиков Игорь Александрович*

САНКТ -ПЕТЕРБУРГ  
2020

# Упражнение 1 -1. Создание собственных классов

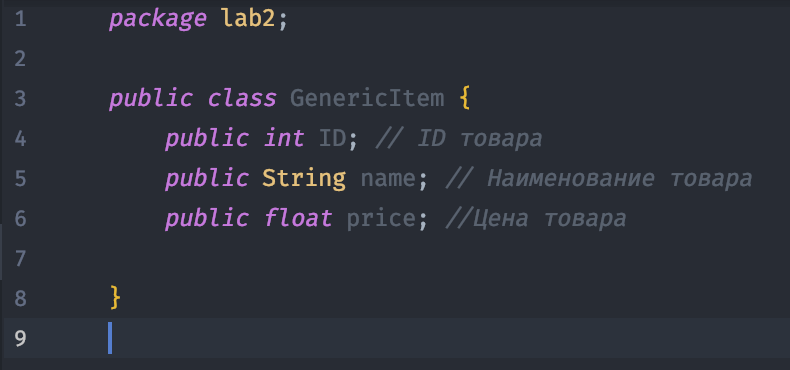
Цель упражнения: Научиться создавать собственные классы, описывать их структуру и использовать в программе.

Описание упражнения: В этом упражнении вам предлагается реализовать класс GenericItem, описывающий отдельный товар из интернет-магазина. Каждый товар характеризуется:

* 1. Уникальным числовым идентификатором
  2. Наименованием
  3. Ценой

Реализуйте в новом проекте класс GenericItem и создайте несколько его экземпляров.

1. Создайте в Eclipse новый проект StockListProject Изображение выглядит как снимок экрана

   Автоматически созданное описание
2. Создайте в проекте StockListProject новый класс GenericItem со следующими полями экземпляра:   
   public int ID; // ID товара  
   public String name; // Наименование товара  
   public float price; //Цена товара 
3. Добавьте в класс GenericItem метод printAll(), выводящий на экран значения всех полей экземпляра класса GenericItem:   
   class GenericItem {  
     public int ID;  
     public String name;  
     public float price;  
     void printAll(){

System.out.printf("ID: %d , Name: %-20s , price:%5.2f \n",ID,name,price);

}  
 }

Изображение выглядит как снимок экрана

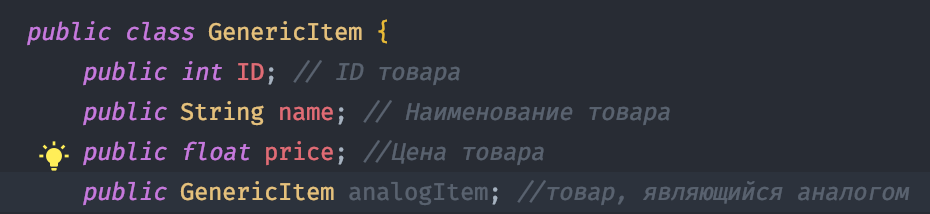
Автоматически созданное описание

1. Создайте стартовый класс по имени Main. Добавьте в этот класс метод main. В методе main создайте три экземпляра класса GenericItem и присвойте их полям различные значения. Распечатайте значения полей для всех экземпляров методом printAll();

Изображение выглядит как снимок экрана

Автоматически созданное описание

1. (Опционально) Предположим, что для каждого товара необходимо хранить информацию о том, какой товар является его аналогом. Подумайте, какое поле необходимо добавить в класс GenericItem для хранения такой информации. Внесите в класс GenericItem соответствующие изменения.



# Упражнение 1-2. Создание перечислений.

Цель упражнения: Научиться работать с перечислениями.

Описание упражнения: Необходимо добавить в класс GenericItem поле перечислимого типа, характеризующее категорию товара (пищевой, одежда, печатная продукция и т.д.)

1. Создайте в проекте StockListProject новое перечисление Category со следующими значениями: FOOD (пищевой), PRINT (печатная продукция), DRESS (одежда), GENERAL (иная категория) public enum Category { FOOD, PRINT, DRESS, GENERAL }
2. Добавьте в класс GenericItem, поле типа Category со значением по умолчанию GENERAL. Внесите изменения в метод printAll, позволяющие печатать категорию товара.

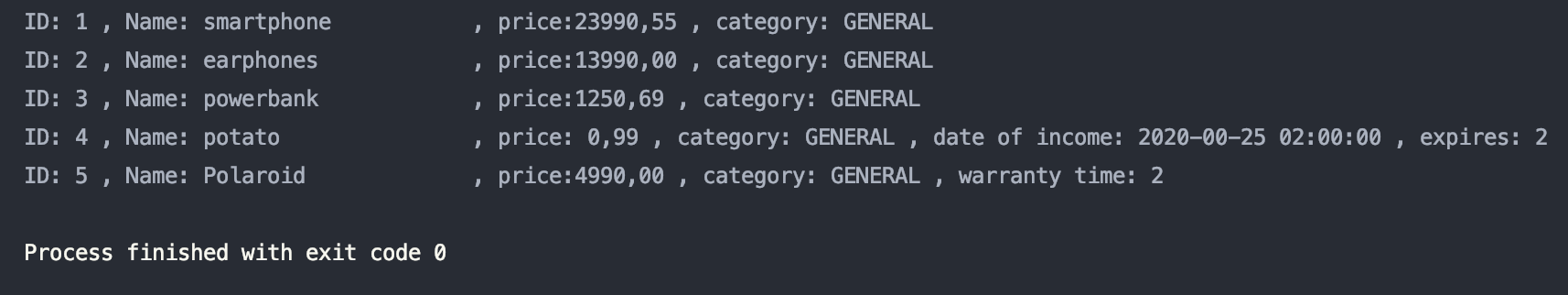
# Упражнение 2-1. Применение наследования

Цель упражнения: Научиться использовать механизмы наследования в Java.

Описание упражнения: В этом упражнении вы примените механизмы наследования языка Java для упрощения структуры программы и уменьшения объема кода.

1. Унаследуйте от класса GenericItem (см. упражнения 6-1, 6-2) классы FoodItem, и TechicalItem со следующими характеристиками:
   1. Класс FoodItem имеет дополнительные поля:
      1. Date dateOfIncome; // дата производства
      2. short expires; // срок годности
   2. Класс TechnicalItem имеет дополнительные поля
      1. short warrantyTime; // гарантийный срок (суток)
2. Перекройте метод printAll в обоих наследниках так, чтобы он выводил на экран помимо общих индивидуальные характеристики объекта. Изображение выглядит как снимок экрана, монитор, экран, телевидение

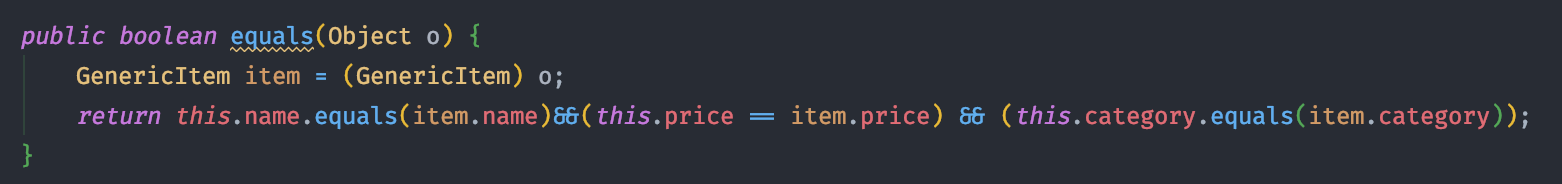
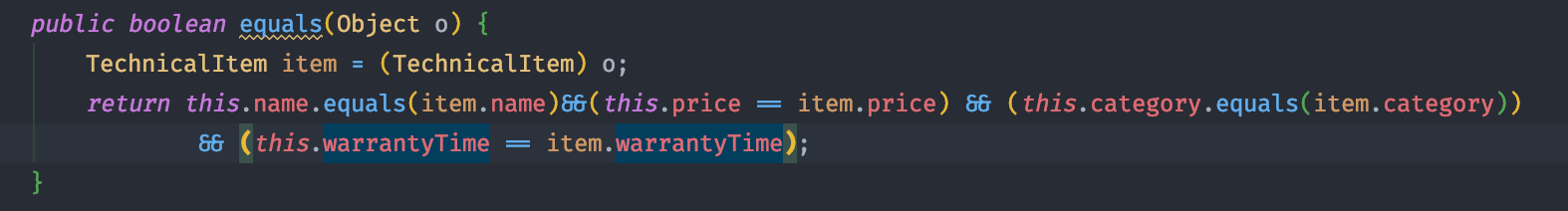
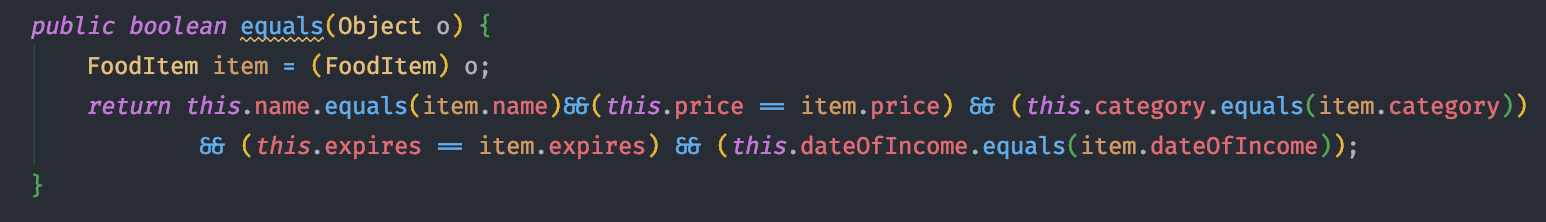
   Автоматически созданное описание
3. В классе Main создайте экземпляры классов FoodItem и TechnicalItem, поместите их в один массив. Переберите в цикле элементы массива и выведите на экран информацию об этих элементах с помощью метода printAll. Изображение выглядит как снимок экрана

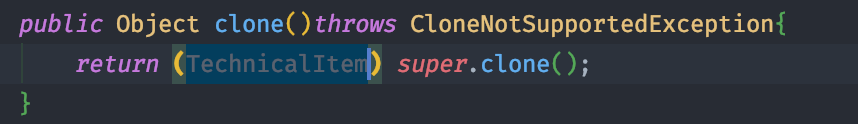
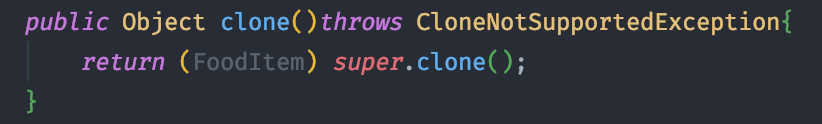
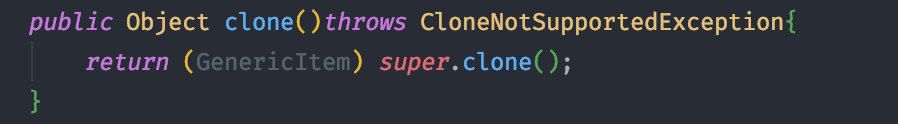
   Автоматически созданное описание
4. Скомпилируйте и выполните проект. 

# Упражнение 2-2(Опционально). Использование методов класса Object.

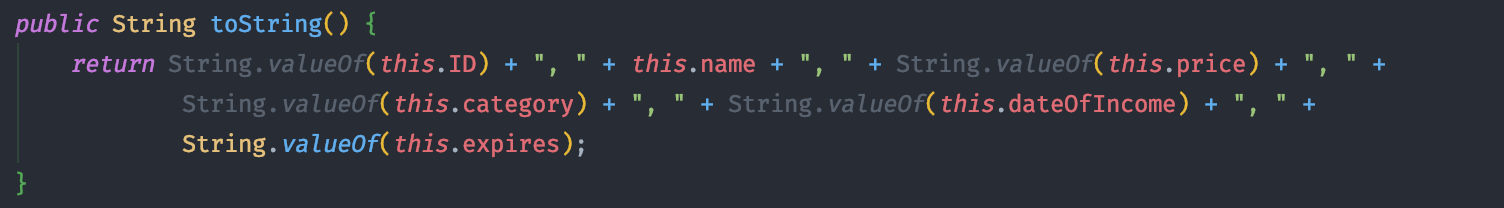
Цель упражнения: Изучить полезные методы, предоставляемые классом Object.

Описание упражнения: В этом упражнении вы воспользуетесь функциями класса Object для сравнения и копирования объекта.

1. Добавьте в классы GenericItem, FoodItem и TechnicalItem реализацию метода   
   public boolean equals(Object o). 
2. Создайте два экземпляра класса FoodItem. Сравните их с помощью метода equals. Выведите на экран результат сравнения. Изображение выглядит как снимок экрана

   Автоматически созданное описание
3. Добавьте в классы GenericItem, FoodItem и TechnicalItem реализацию метода   
   public Object clone(). Клонируйте один из ранее созданных экземпляров класса FoodItem. Сравните с помощью метода equals оригинал и его клон.  Изображение выглядит как снимок экрана

   Автоматически созданное описание
4. Добавьте в классы GenericItem, FoodItem и TechnicalItem реализацию метода   
   public String toString( ). Изображение выглядит как устройство

   Автоматически созданное описаниеИзображение выглядит как монитор

   Автоматически созданное описание

# Упражнение 2-3(Опционально). Расширенное клонирование.

1. Измените реализацию метода clone в классе GenericItem так, чтобы при клонировании товара клонировался его аналог. 