Розробник:

Огонькова Наталія Максимівна

Група КІТ 101.8б

Варіант №11

**Лабораторна робота № 7**

Об’єктно-орієнтована декомпозиція

**Мета**:

* Використання об’єктно-орієнтованого підходу для розробки об’єкта предметної (прикладної) галузі.

**Вимоги:**

1. Використовуючи об’єктно-орієнтований аналіз, реалізувати класи для представлення сутностей відповідно прикладної задачі - domain-об’єктів.
2. Забезпечити та продемонструвати коректне введення та відображення кирилиці.
3. Продемонструвати можливість управління масивом domain-об’єктів.
4. ОПИС ПРОГРАМИ
   1. Опис змінних

Store rec1 = **new** Store(); //перший елемент класу Store

Store rec2 = **new** Store(); //другий елемент класу Store

Container contains = **new** Container(); //змінна для контейнера

1.2 Ієрархія та структура класів

**class** Lab07 – точка входу в програму

**class** Container – клас, в якому містяться контейнери

**class** Store – клас, що містить дані, введені користувачем

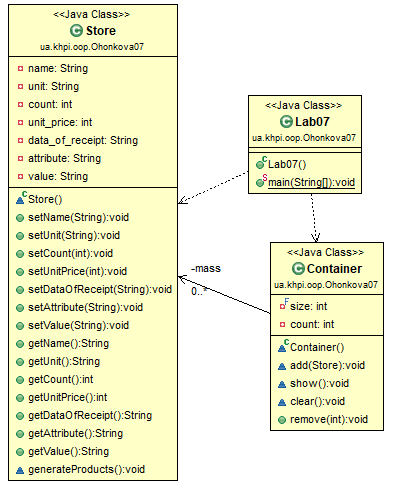


Рисунок 1 – Класи програми

* 1. Текст програми

**class** Lab07

**package** ua.khpi.oop.Ohonkova07;

**public** **class** Lab07 {

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

Container contains = **new** Container();

Store rec1 = **new** Store();

Store rec2 = **new** Store();

rec1.generateProducts();

contains.add(rec1);

System.***out***.println("\n");

rec2.generateProducts();

contains.add(rec2);

contains.show();

}

}

**class** Container

**package** ua.khpi.oop.Ohonkova07;

**import** java.util.Arrays;

**class** Container {

/\*\*

\* First size for array.

\*/

**private** **final** **int** size = 10;

/\*\*

\* Array contains all data.

\*/

**private** **int** count = 0;

**private** Store[] mass = **new** Store[size];

**void** add(**final** Store temp){

**if** (count == mass.length) {

mass = Arrays.*copyOf*(mass, mass.length \* 2);

mass[count++] = temp;

} **else** {

mass[count++] = temp;

}

}

**void** show() {

**if** (count == 0) {

System.***out***.println("Ошибка! Массив пустой!");

} **else** {

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

**for** (**int** i = 0; i < count; i++) {

System.***out***.println("Товар №" + (i+1));

System.***out***.println("Название: " + mass[i].getName());

System.***out***.println("Единица измерения: " + mass[i].getUnit());

System.***out***.println("Количество: " + mass[i].getCount());

System.***out***.println("Цена за единицу: " + mass[i].getUnitPrice());

System.***out***.println("Дата получения: " + mass[i].getDataOfReceipt());

System.***out***.println("Свойства: " + mass[i].getAttribute());

System.***out***.println("Значение: " + mass[i].getValue());

System.***out***.println("\n");

}

}

}

**void** clear() {

**for** (**int** i = 0; i < count; i++) {

mass[i] = **null**;

}

count = 0;

}

**public** **void** remove(**int** index) {

**for** (**int** i = index; i < count; i++) {

mass[index++] = mass[i + 1];

}

count--;

}

}

**class** Store

**package** ua.khpi.oop.Ohonkova07;

**import** java.util.Scanner;

**class** Store {

**private** String name;

**private** String unit;

**private** **int** count;

**private** **int** unit\_price;

**private** String data\_of\_receipt;

**private** String attribute;

**private** String value;

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

**this**.name = name;

}

**public** **void** setUnit(String unit) {

**this**.unit = unit;

}

**public** **void** setCount(**int** count) {

**this**.count = count;

}

**public** **void** setUnitPrice(**int** unit\_price) {

**this**.unit\_price = unit\_price;

}

**public** **void** setDataOfReceipt(String data\_of\_receipt) {

**this**.data\_of\_receipt = data\_of\_receipt;

}

**public** **void** setAttribute(String attribute) {

**this**.attribute = attribute;

}

**public** **void** setValue(String value) {

**this**.value = value;

}

**public** String getName() {

**return** name;

}

**public** String getUnit() {

**return** unit;

}

**public** **int** getCount() {

**return** count;

}

**public** **int** getUnitPrice() {

**return** unit\_price;

}

**public** String getDataOfReceipt() {

**return** data\_of\_receipt;

}

**public** String getAttribute() {

**return** attribute;

}

**public** String getValue() {

**return** value;

}

**void** generateProducts() {

Scanner scan = **new** Scanner(System.***in***);

Scanner scan2 = **new** Scanner(System.***in***);

**int** choose = 0;

System.***out***.print("Введите название товара: ");

name = scan2.nextLine();

System.***out***.print("Введите единицу измирения: ");

unit = scan2.nextLine();

System.***out***.print("Введите количество товара: ");

count = scan.nextInt();

System.***out***.print("Введите цену за единицу товара: ");

unit\_price = scan.nextInt();

System.***out***.print("Введите дату получения: ");

data\_of\_receipt = scan2.nextLine();

System.***out***.print("Свойства: ");

attribute = scan2.nextLine();

System.***out***.print("Значение: ");

value = scan2.nextLine();

}

}

1. ВАРІАНТИ ВИКОРИСТАННЯ

Програму можна використовувати для створення контейнеру з даними, що вводить користувач, стосовно теми «магазин».

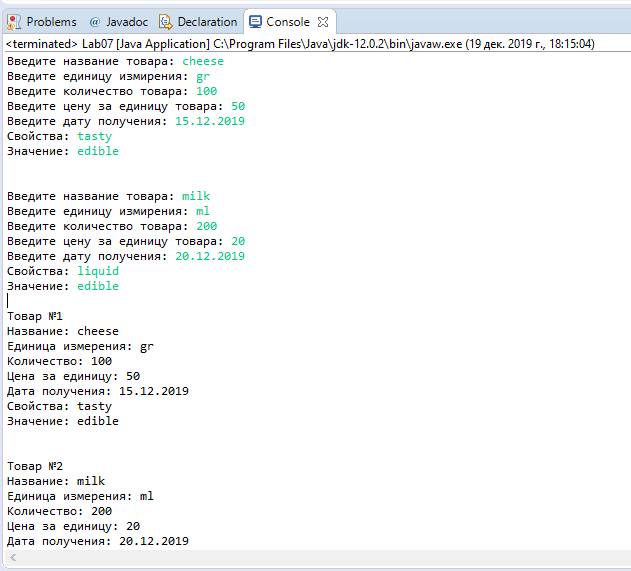


Рисунок 2 – Результат програми

ВИСНОВОК

При виконанні лабораторної роботи набуто практичних навичок, щодо використання об’єктно-орієнтованого підходу для розробки об’єкта предметної (прикладної) галузі. Програма виконується без помилок.