**Лабораторна робота №7**

[**Об'єктно-орієнтована декомпозиція**](https://oop-khpi.gitlab.io/#tasks)

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

**1 ВИМОГИ**

1. Використовуючи об'єктно-орієнтований аналіз, реалізувати класи для представлення сутностей відповідно [прикладної задачі](https://oop-khpi.gitlab.io/%D0%B7%D0%B0%D0%B2%D0%B4%D0%B0%D0%BD%D0%BD%D1%8F/task07/#_4) - domain-об'єктів.
2. Забезпечити та продемонструвати коректне введення та відображення кирилиці.
3. Продемонструвати можливість управління масивом domain-об'єктів.

**1.1Розробник**

- П.І.Б : Наймитенко С.І.

- Група: КІТ-119а

- Варіант 15

**1.3 Задача**

**Автосалон**

Автомобіль: марка, рік випуску, технічні характеристики у вигляді "параметр - значення" (серед яких витрата палива в міському і заміському циклі), технічний стан, ціна.

**2 ОПИС ПРОГРАМИ**

**2.2 Ієрархія та структура класів**

Було створено 3 класи:

* publicclassHotelRoom – клас, що містить необхідні поля та методи прикладної області “Готель”.
* publicclassRoomBooking – клас, що містить масив об’єктів, а також метод виводу даних.
* public class Main – містить лише метод main.

**Важливі фрагменти програми**

**Клас CarsShop**

**public** **class** CarsShop {

**private** String car\_brand;

**private** **int** year\_create;

**private** String specifications;

**private** **int** price;

**private** Technical\_condition obj;

**private** **class** Technical\_condition

{

**private** **int** city\_fuel\_consumption;

**private** **int** fuel\_consumption\_outside\_the\_city;

**public** **void** setCity\_fuel\_consumption(**int** city\_fuel\_consumption)

{

**this**.city\_fuel\_consumption = city\_fuel\_consumption;

}

**public** **void** setFuel\_consumption\_outside\_the\_city(**int** fuel\_consumption\_outside\_the\_city)

{

**this**.fuel\_consumption\_outside\_the\_city = fuel\_consumption\_outside\_the\_city;

}

**public** **int** getCity\_fuel\_consumption()

{

**return** fuel\_consumption\_outside\_the\_city;

}

**public** **int** getFuel\_consumption\_outside\_the\_city()

{

**return** fuel\_consumption\_outside\_the\_city;

}

**public** Technical\_condition(**int** city\_fuel\_consumption, **int** fuel\_consumption\_outside\_the\_city)

{

**this**.city\_fuel\_consumption = city\_fuel\_consumption;

**this**.fuel\_consumption\_outside\_the\_city = fuel\_consumption\_outside\_the\_city;

}

}

**public** **void** setCar\_brand(String car\_brand)

{

**this**.car\_brand = car\_brand;

}

**public** **void** setYear\_create(**int** year\_create)

{

**this**.year\_create = year\_create;

}

**public** **void** setSpecifications(String specifications)

{

**this**.specifications = specifications;

}

**public** **void** setPrice(**int** price)

{

**this**.price = price;

}

**public** String getCar\_brand()

{

**return** car\_brand;

}

**public** **int** getYear\_create()

{

**return** year\_create;

}

**public** String getSpecifications()

{

**return** specifications;

}

**public** **int** getPrice()

{

**return** price;

}

**public** CarsShop(String car\_brand, **int** year\_create, String specifications, **int** price, **int** city\_fuel\_consumption, **int** fuel\_consumption\_outside\_the\_city)

{

**this**.car\_brand = car\_brand;

**this**.year\_create = year\_create;

**this**.specifications = specifications;

**this**.price = price;

Technical\_condition new\_obj = **new** Technical\_condition(city\_fuel\_consumption,fuel\_consumption\_outside\_the\_city);

obj = new\_obj;

}

**public** **void** Show()

{

System.***out***.println("Марка автомобиля - "+ car\_brand +"\r\n" +

"Цена автомобиля "+ price + "\r\n" +

"Технічні характеристики: витрата палива в міському циклі - " +obj.getCity\_fuel\_consumption() +" витрата палива в заміському циклі - "+ obj.getFuel\_consumption\_outside\_the\_city() + "\r\n" +

"Год выпуска автомобиля "+ year\_create + "\r\n" +

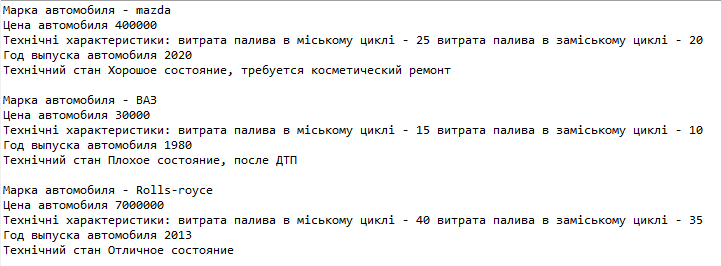
"Технічний стан"+ specifications + "\r\n" );

}

}

**3 ВИСНОВКИ**

Результат роботи програми:



**ВИСНОВКИ**

У результаті виконання лабораторної роботи було набуто навичок роботи з domain-об‘єктами; було набуто навичок з об‘єктно-орієнтованою декомпозицією у середовищі JavaEclipse.