

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ  
«ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

Кафедра «ОБЧИСЛЮВАЛЬНОЇ ТЕХНІКИ ТА ПРОГРАМУВАННЯ»

«Об'єктно-орієнтоване програмування»

*Звіт з лабораторної роботи №7*  
*Тема: «Об'єктно-орієнтована декомпозиція»*

Виконав:  
ст. гр. 1.КІТ102.8а  
Міщенко Д.С.

Перевірив:  
Пугачов Р.В.

Харків – 2019

# **Лабораторна робота №7**

## **Об'єктно-орієнтована декомпозиція**

### ***Мета***

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

## **1 ВИМОГИ**

### **1.1 Розробник**

1. Міщенко Дмитро Сергійович
2. 1.KIT102.8a
3. Варіант 13

### **1.2 Загальне завдання**

#### ***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.

#### ***Прикладні задачі***

13. Готель. Замовлення номера: паспортні дані; дати поселення та виселення; номер (клас, число місць); причина поселення (кількість не обмежена).

## 2 ОПИС ПРОГРАМИ

### 2.1 Засоби ООП

Використана композиція(клас HotelRoom в Hotel), інкапсуляція .

### 2.2 Ієрархія та структура класів

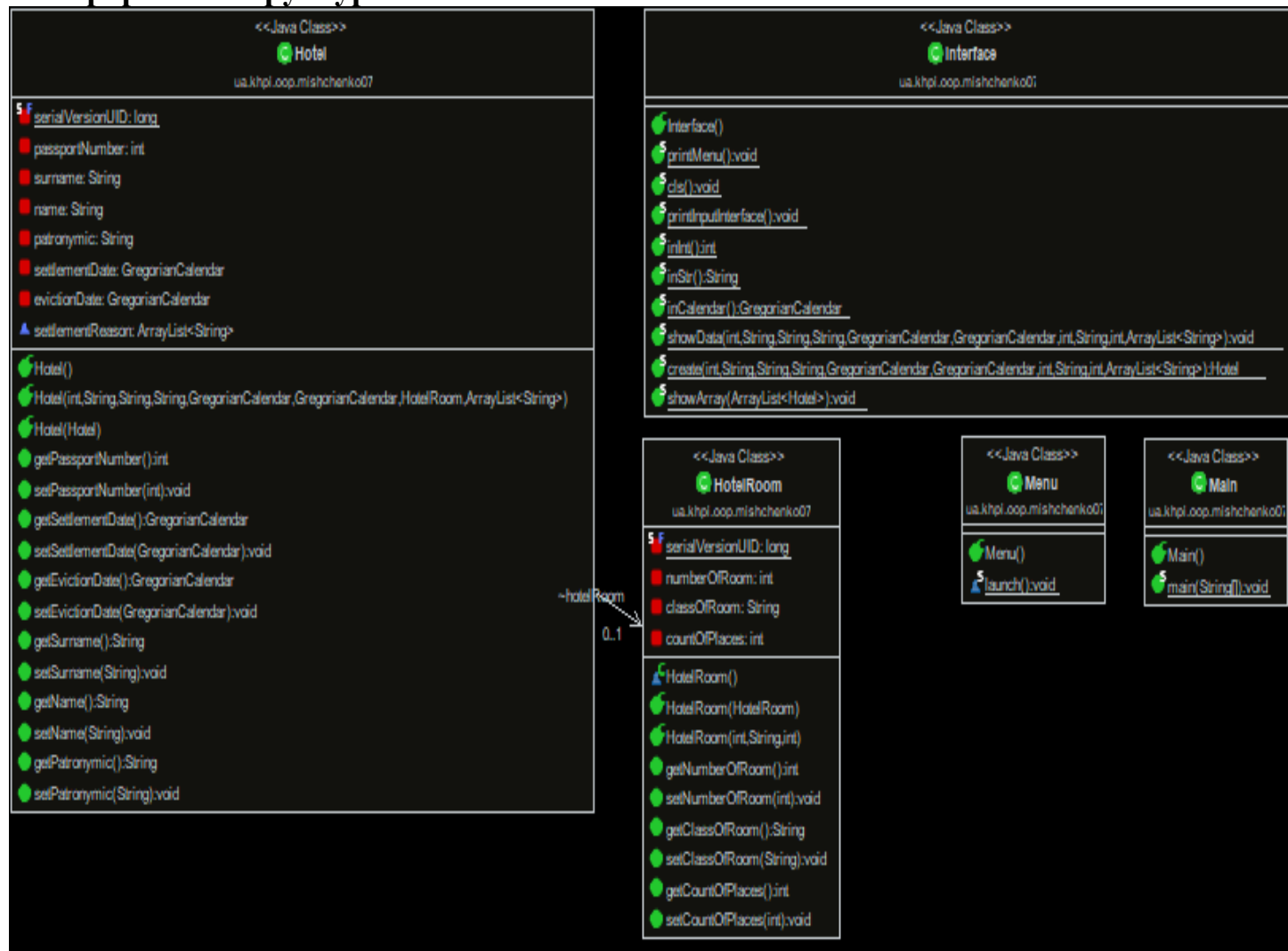


Рисунок 1 – Структура класів

## 2.3 Важливі фрагменти програми

```
public class Hotel implements Serializable {  
  
    private static final long serialVersionUID = 1L;  
  
    private int passportNumber;  
    private String surname;  
    private String name;  
    private String patronymic;  
    private GregorianCalendar settlementDate;  
    private GregorianCalendar evictionDate;  
    HotelRoom hotelRoom;  
  
    ArrayList<String> settlementReason;
```

Рисунок 2 – Поля прикладної галузі

```
static void launch() {  
  
    System.out.println("Добро пожаловать!");  
    ArrayList<Hotel> clients = new ArrayList<Hotel>();
```

Рисунок 3 – Створення масиву об'єктів

```
case 2:  
    Interface.showArray(clients);  
    System.out.println("Введите индекс клиента для удаления");  
    int index = Interface.inInt();  
    clients.remove(index-1);  
    break;  
  
case 3:  
    clients.clear();  
    System.out.println("Нажмите любую клавишу для продолжения...");  
    new java.util.Scanner(System.in).nextLine();  
    break;  
  
case 4:  
    Interface.showArray(clients);  
    System.out.println("Нажмите любую клавишу для продолжения...");  
    new java.util.Scanner(System.in).nextLine();  
    break;
```

Рисунок 4 – Методи управління масивом

### 3 ВАРІАНТИ ВИКОРИСТАННЯ

```

Возможные варианты работы с данными:
1. Добавить нового клиента
2. Удалить клиента
3. Очистить список клиентов
4. Показать список
5. Выход
4
////////////////////////////////////1\\////////////////////////////////////
|  Номер паспорта  |      Фамилия      |      Имя      |
|  123456789      |      Петров       |      Иван      |
|                  |                   |                |
Причины заселения
1. Работа вне места работы
2. Временное проживание за границей

```

### Рисунок 5 – Виведення масиву об'єктів

```

Возможные варианты работы с данными:
1. Добавить нового клиента
2. Удалить клиента
3. Очистить список клиентов
4. Показать список
5. Выход

1
1. Ввести номер паспорта
2. Ввести фамилию
3. Ввести имя
4. Ввести отчество
5. Ввести дату заселения
6. Ввести дату выселения
7. Ввести номер помещения
8. Ввести класс номера
9. Ввести количество мест в номере
10. Ввести причину заселения
11. Показать введенные данные
12. Подтвердить ввод и добавить клиента
0. Выход

```

### Рисунок 6 – Варіант введення даних

## **ВИСНОВКИ**

В результаті виконання лабораторної роботи використано об'єктно-орієнтований підхід для розробки об'єкта предметної (прикладної) галузі.