

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
«ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

Кафедра «ОБЧИСЛЮВАЛЬНОЇ ТЕХНІКИ ТА ПРОГРАМУВАННЯ»

«Об'єктно-орієнтоване програмування»

Звіт з лабораторної роботи №11
Тема: «Регулярні вирази. Перевірка даних»

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

Перевірив:
Пугачов Р.В.

Харків – 2019

Лабораторна робота №11

Регулярні вирази. Перевірка даних

Мета

Ознайомлення з принципами використання регулярних виразів для перевірки рядка на відповідність шаблону.

1 ВИМОГИ

1.1 Розробник

1. Міщенко Дмитро Сергійович
2. КІТ-118в
3. Варіант 13

1.2 Загальне завдання

Вимоги

Продемонструвати ефективне (оптимальне) використання регулярних виразів для перевірки коректності (валідації) даних, що вводяться, перед записом в domain-об'єкти відповідно до призначення кожного поля для заповнення розробленого контейнера:

- при зчитуванні даних з текстового файла в автоматичному режимі;
- при введенні даних користувачем в діалоговому режимі.

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

Поліморфізм,інкапсуляція

2.2 Ієрархія та структура даних

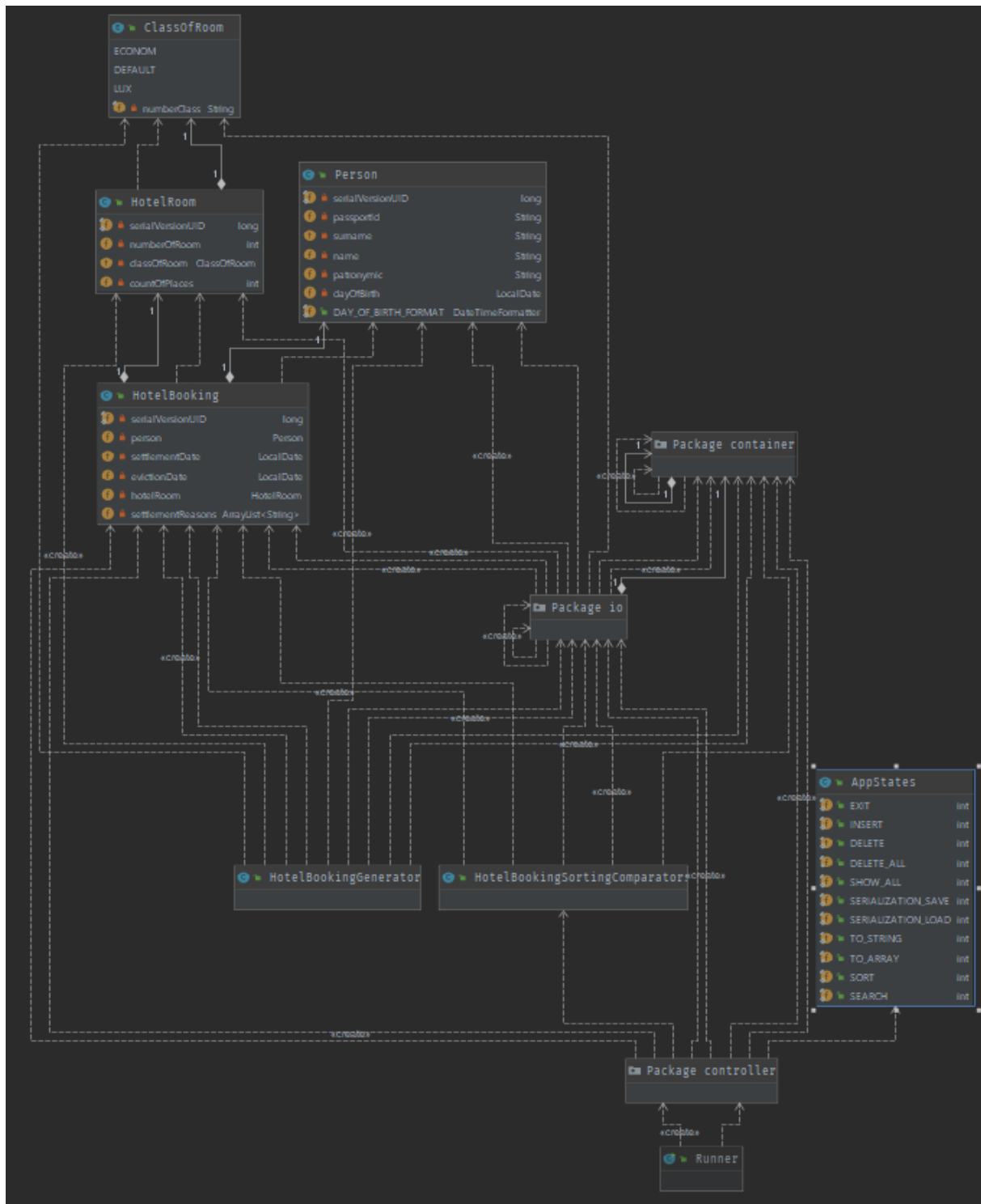


Рисунок 1 – Ієрархія класів

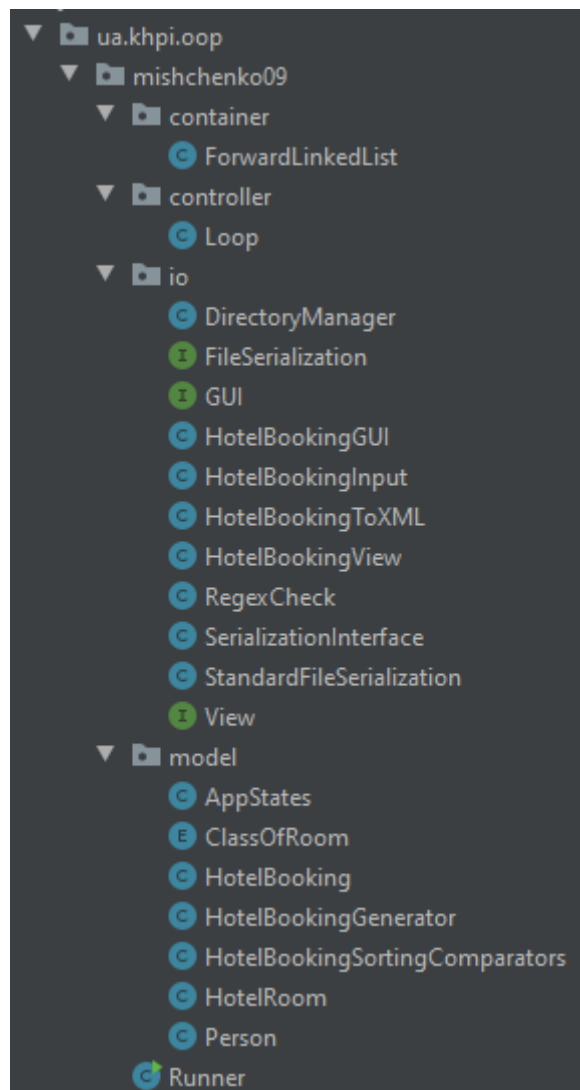


Рисунок 2 – Структура пакету

2.2 Важливі фрагменти програми

Розроблені методи валідації даних (рис. 3)

```
public static boolean checkPassportID(String value) {  
    return value.matches( regex: "[\\d]{9}|[A-Z]{2}[\\d]{6}");  
}  
  
public static boolean checkSNP(String value) {  
    return value.matches( regex: "^([A-Z]{1})[a-z]*");  
}  
  
public static boolean checkNumberOfRoom(String value) {  
    return value.matches( regex: "[\\d]{3,4}");  
}  
  
public static boolean checkCountOfPlaces(String value) {  
    return value.matches( regex: "[1-9]");  
}
```

Рисунок 3 – Методи перевірки даних за допомогою регулярних виразів

Метод повної перевірки об'єкту (рис. 2)

```
public static boolean checkObject(HotelBooking obj) {  
    return checkPassportID(obj.getPerson().getPassportId())  
        && checkSNP(obj.getPerson().getSurname())  
        && checkSNP(obj.getPerson().getName())  
        && checkSNP(obj.getPerson().getPatronymic())  
        && checkNumberOfRoom(String.valueOf(obj.getHotelRoom().getNumberOfRoom()))  
        && checkCountOfPlaces(String.valueOf(obj.getHotelRoom().getCountOfPlaces()))  
        && obj.getSettlementReasons() != null;  
}
```

Рисунок 4 – Метод **checkObject**

Перевірка даних при зчитуванні з файла (рис. 3)

```
if (RegexCheck.checkObject(object)) list.add(object);
```

Рисунок 5 – Виклик метода **checkObject**

3 ВАРІАНТИ ВИКОРИСТАННЯ

```
00:39:02.122 [main] INFO u.k.oop.mishchenko09.controller.Loop - App started  
01. Add new client  
02. Delete client  
03. Clear list  
04. Show list  
05. Save  
06. Load  
07. Get list as string  
08. Get list as array  
09. Sort  
10. Search by reason  
00. Exit  
Choose option
```

Рисунок 6 – Головне меню програми

```
16:30:34.605 [main] ERROR u.k.o.m.io.HotelBookingInput - One or more fields is null  
Enter any key to continue ...
```

Рисунок 7 – Перевірка при додаванні об'єкта

ВИСНОВКИ

В результаті виконання лабораторної роботи відбулось ознайомлення з принципами використання регулярних виразів для перевірки рядка на відповідність шаблону.