



Міністерство освіти і науки України
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Факультет інформатики та обчислювальної техніки
Кафедра інформаційних систем та технологій

ЛАБОРАТОРНА РОБОТА № 3
з дисципліни «Мова програмування Java»
Тема: «Використання ООП та шаблону MVC»
Варіант 119

Виконала:

Студентка групи IA-31

Соколова Поліна

Мета роботи – закріплення на практиці знань з принципів об'єктно орієнтованого програмування: абстракція, інкапсуляція, поліморфізм та наслідування; створення додатку з використанням архітектурного шаблону model-view-controller.

Завдання 3.3

Написати програму, яка моделює роботу вбудованого процесора турнікету лижного підйомника щодо перевірки доступу по ski-pass.

Хід роботи

Програма має MVC-архітектуру:

Model: зберігає дані карток (SkiPass, TimeSkiPass, LiftSkiPass) та їх стан.

View: консольне меню (Main.java) для взаємодії з користувачем.

Controller: SkiPassSystem реалізує логіку управління картками, перевірку доступу, блокування та статистику.

Опис основних класів:

SkiPass - абстрактний клас, що зберігає спільні атрибути: id, type, isBlocked.

Інкапсуляція: поля приватні або protected, доступ через методи (геттери/сеттери).

Поліморфізм: метод use() реалізується у підкласах для різної логіки доступу.

Клас TimeSkiPass наслідує SkiPass. Містить поля: термін дії (validHours), час активації (activationTime). Метод use() перевіряє, чи не прострочена картка. Геттер getValidHours() використовується для статистики.

Клас LiftSkiPass наслідує SkiPass. Містить поле remainingLifts — кількість залишених підйомів. Метод use() зменшує кількість підйомів на 1. Геттер getRemainingLifts() використовується для статистики.

Клас SkiPassSystem зберігає всі картки (Map<Integer, SkiPass>). Методи: addSkiPass(SkiPass pass) — додавання нової картки, checkAccess(int id) — перевірка

доступу і списання підйомів, `blockSkiPass(int id)` — блокування картки, `printStatistics()` — вивід детальної та сумарної статистики.

`SkiPassType` це enum з переліком типів Ski-pass: на робочі/виходні дні, абонемен.

У `Main` реалізовується консольне меню для користувача:

1. Випустити ski-pass
2. Перевірити доступ
3. Показати статистику
4. Заблокувати ski-pass
5. Вийти

Для створення картки запитується тип, спосіб обліку (за часом / за підйомами) та відповідні параметри.

Також реалізовано unit-тест для перевірки основної функціональності системи `ski-pass`: створюється нова система турнікету і картка на підйоми для робочих днів, на 5 підйомів, далі перевіряється, що перший прохід дозволений, бо картка нова, активна, підйоми є, і що після блокування прохід заборонений.

Тест пройшов успішно:

```
✓  ✓ SkiPassTest (org.example)          32 ms | ✓ 1 test passed  1 test total, 32 ms
    ✓ testAccessAndBlock()               32 ms | C:\Users\polin\.jdks\openjdk-24.0.2+12-
                                               
                                                Process finished with exit code 0
```

Результат роботи:

```
---- Турнікет лижного підйомника ----
1. Випустити ski-pass
2. Перевірити доступ
3. Показати статистику
4. Заблокувати ski-pass
5. Вийти
Вибір дії: 1

Оберіть тип ski-pass:
1. На робочі дні
2. На вихідні дні
3. Абонемент на сезон
Ваш вибір: 2
```

```
Оберіть варіант:  
1. Без обліку кількості підйомів  
2. По кількості підйомів  
Ваш вибір: 2  
  
Оберіть кількість підйомів:  
1. 10  
2. 20  
3. 50  
4. 100  
Ваш вибір: 2  
Створено ski-pass на 20 підйомів. ID: 1
```

```
Статистика SkiPass (всі картки)  
Сумарно: всього=4, дозволено=4, заблоковано=0
```

```
По типах ski-pass  
WEEKDAY: 1 карток  
WEEKEND: 2 карток  
SEASON: 1 карток
```

```
Деталіожної картки  
ID: 1, type: WEEKEND, blocked: false, remaining lifts: 20  
ID: 2, type: WEEKEND, blocked: false, valid for: 4 годин  
ID: 3, type: WEEKDAY, blocked: false, remaining lifts: 10  
ID: 4, type: SEASON, blocked: false, valid for: 2160 годин
```

```
---- Турнікет лижного підйомника ----
```

1. Випустити ski-pass
2. Перевірити доступ
3. Показати статистику
4. Заблокувати ski-pass
5. Вийти

Вибір дії: 2

Введіть ID ski-pass: 1

Прохід дозволено

```
---- Турнікет лижного підйомника ----
```

1. Випустити ski-pass
2. Перевірити доступ
3. Показати статистику
4. Заблокувати ski-pass
5. Вийти

Вибір дії: 4

Введіть ID ski-pass для блокування: 2

Ski-pass 2 заблоковано.

```
Статистика SkiPass (всі картки)  
Сумарно: всього=4, дозволено=3, заблоковано=1
```

```
По типах ski-pass  
WEEKDAY: 1 карток  
WEEKEND: 2 карток  
SEASON: 1 карток
```

```
Деталіожної картки  
ID: 1, type: WEEKEND, blocked: false, remaining lifts: 19  
ID: 2, type: WEEKEND, blocked: true, valid for: 4 годин  
ID: 3, type: WEEKDAY, blocked: false, remaining lifts: 10  
ID: 4, type: SEASON, blocked: false, valid for: 2160 годин
```

Висновок: у результаті виконання лабораторної роботи я створила систему турнікета лижного підйомника з різними типами ski-pass. У ході роботи застосовано практичні знання ООР-принципів: абстракція, інкапсуляція, поліморфізм, наслідування. Побудовано застосунок за шаблоном MVC, що дозволяє розділити дані, логіку та інтерфейс, розроблено консольний інтерфейс, статистику, а також unit-тест, що перевіряє доступ і блокування карток.