

Міністерство освіти і науки України
Національний університет «Львівська політехніка»



Лабораторна робота №10

на тему:

«Об'єктно-орієнтована декомпозиція.

Основи введення/виведення Java SE.»

з курсу:

«ООП»

Виконав:

ст. гр. КН-110

Георгій Брусенцов

Прийняв:

Гасько Р.Т

Львів – 2018 р.

Мета роботи:

Використання об'єктно-орієнтованого підходу для розробки об'єкта предметної (прикладної) галузі. Оволодіння навичками управління введенням/виведенням даних з використанням класів Java SE.

Розробник: дивитися вище, Варіант 1

Вимоги:

- 1. Використовуючи об'єктно-орієнтований аналіз, реалізувати класи для представлення сутностей відповідно списку прикладних задач - domain-об'єктів (Прикладні задачі. Список №2. 20 варіантів)*
- 2. Забезпечити та продемонструвати коректне введення та відображення кирилиці.*
- 3. Продемонструвати можливість управління масивом domain-об'єктів.*
- 4. Забезпечити можливість збереження і відновлення масива об'єктів рішення завдання з Прикладні задачі. Список №2. 20 варіантів .*
- 5. Забороняється використання стандартного протокола серіалізації . docs.oracle.com/javase/8/docs/platform/serialization/spec/serialTOC.html*
- 6. Продемонструвати використання моделі Long Term Persistence . docs.oracle.com/javase/tutorial/javabeans/advanced/longpersistence.html*
- 7. Забезпечити діалог з користувачем у вигляді текстового меню.*
- 8. При збереженні та відновленні даних забезпечити діалоговий режим вибору директорії з відображенням вмісту і можливістю переміщення по підкаталогах.*

Приклад коду:

```
package ua.lpnuai.oop.brusentsov10;  
  
public class CV {  
    private Id id = new Id();  
    private Specialization spec = new Specialization();  
    private Seniority senior = new Seniority();
```

```

private Education edu = new Education();
private FiredDate fd = new FiredDate();
private MinimalWage mw = new MinimalWage();

CV(int _id, String _spec, String _senior, String _edu, String _fd){
    System.out.println("Enter ID, Specialization, Seniority, Education,
FiredDate");
    id.setId(_id);
    spec.setSpec(_spec);
    senior.setSenior(_senior);
    edu.setEdu(_edu);
    fd.setDate(_fd);
}

CV(int _id, String _spec, String _senior, String _edu, String _fd, int
_mw){
    id.setId(_id);
    spec.setSpec(_spec);
    senior.setSenior(_senior);
    edu.setEdu(_edu);
    fd.setDate(_fd);
    mw.setMinWag(_mw);
}

@Override
public String toString() {
    if(mw.getMinWag() == 0) {
        return "ID: " + id.getId() + "\n"
            + "Specialization: " + spec.getSpec() + "\n"
            + "Seniority: " + senior.getSenior() + "\n"
            + "Education: " + edu.getEdu() + "\n"
            + "Fired date: " + fd.getDate() + "\n";
    }
    else
        return "ID: " + id.getId() + "\n"
            + "Specialization: " + spec.getSpec() + "\n"
            + "Seniority: " + senior.getSenior() + "\n"
            + "Education: " + edu.getEdu() + "\n"
            + "Fired date: " + fd.getDate() + "\n"
            + "Minimal Wage: " + mw.getMinWag() + "\n";
    }
}

```

Висновки: я навчився використовувати об'єктно-орієнтований підхід для розробки об'єкта предметної (прикладної) галузі та оволодів навичками управління введенням/виведенням даних з використанням класів *Java SE*.