

Лабораторна робота №7

Об'єктно-орієнтована декомпозиція

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 - domain-об'єктів.
- 2) Забезпечити та продемонструвати коректне введення та відображення кирилиці.
- 3) Продемонструвати можливість управління масивом domain-об'єктів.

1.1 Розробник

- П.І.Б: Абдуллін О. Р.
- Група: КІТ-119а
- Варіант: 1

1.2 Завдання

Кадрове агенство.

Дані про претендента: реєстраційний номер; досвід роботи - набір значень "спеціальність, стаж"; освіта; дата звільнення; вимоги до майбутньої роботи - набір необов'язкових властивостей у вигляді "спеціальність, умови праці, мінімальна зарплата".

2 ОПИС ПРОГРАМИ

2.1 Було використано наступні засоби:

public String getEducation() – геттер

public void setEducation(String education) - сеттер

2.2 Ієрархія та структура класів

Було створено 3 класи

- 1) public class Main - містить метод main.
- 2) public class Challenger - клас, що містить поля претендента на роботу.
- 3) public class RecruitinfAgency – клас, що містить масив типу Challenger та метод вивиду усіх даних.
- 4) public class DemandsToWork – містить інформацію про вимоги претендента до роботи.
- 5) public class WorkExperience – містить інформацію про досвід роботи претендента.

2.3 Важливі фрагменти програми

Клас Challenger

```
public class Challenger {  
    private int registrationNum;  
    private String education;  
    private int dismissalDay;  
    private int dismissalMonth;
```

```

private int dismissalYear;
private WorkExperience workExperience;
private DemandsToWork demandsToWork;
/**
 * Конструктор
 * @param registrationNum ID претендента
 * @param education образование претендента
 * @param dismissalDay день увольнения претендента
 * @param dismissalMonth месяц увольнения претендента
 * @param dismissalYear год увольнения претендента
 * @param workExperience опыт работы претендента
 * @param demandsToWork пожелания к будущей работе
 */
public Challenger(int registrationNum, String education, int dismissalDay, int
dismissalMonth, int dismissalYear, WorkExperience workExperience, DemandsToWork
demandsToWork ) {
    this.registrationNum = registrationNum;
    this.education = education;
    this.dismissalDay = dismissalDay;
    this.dismissalMonth = dismissalMonth;
    this.dismissalYear = dismissalYear;
    this.workExperience = workExperience;
    this.demandsToWork = demandsToWork;
}
/**
 * Геттер ID претендента
 * @return ID претендента
 */
public int getRegistrationNum() {
    return registrationNum;
}
/**
 * Сеттер ID претендента
 * @param registrationNum ID претендента
 */
public void setRegistrationNum(int registrationNum) {
    this.registrationNum = registrationNum;
}
/**
 * Геттер образования претендента
 * @return образование претендента
 */
public String getEducation() {
    return education;
}
/**
 * Сеттер образования претендента
 * @param education Образование претендента
 */
public void setEducation(String education) {
    this.education = education;
}
/**
 * Геттер дня увольнения
 * @return день увольнения
 */
public int getDismissalDay() {
    return dismissalDay;
}
/**
 * Сеттер дня увольнения
 * @param dismissalDay день увольнения

```

```

    */
    public void setDismissalDay(int dismissalDay) {
        this.dismissalDay = dismissalDay;
    }
    /**
     * Геттер месяца увольнения
     * @return месяц увольнения
     */
    public int getDismissalMonth() {
        return dismissalMonth;
    }
    /**
     * Сеттер месяца увольнения
     * @param dismissalMonth месяц увольнения
     */
    public void setDismissalMonth(int dismissalMonth) {
        this.dismissalMonth = dismissalMonth;
    }
    /**
     * Геттер года увольнения претендента
     * @return год увольнения
     */
    public int getDismissalYear() {
        return dismissalYear;
    }
    /**
     * Сеттер года увольнения претендента
     * @param dismissalYear год увольнения
     */
    public void setDismissalYear(int dismissalYear) {
        this.dismissalYear = dismissalYear;
    }
    /**
     * Геттер опыта работы претендента
     * @return
     */
    public WorkExperience getWorkExperience() {
        return workExperience;
    }
    /**
     * Сеттер опыта работы претендента
     * @param workExperience
     */
    public void setWorkExoerience(WorkExperience workExperience) {
        this.workExperience = workExperience;
    }
    /**
     * Геттер требований к будущей работе
     * @return
     */
    public DemandsToWork getDemandsToWork() {
        return demandsToWork;
    }
    /**
     * Сеттер требований к будущей работе
     * @param demandsToWork
     */
    public void setDemandsToWork(DemandsToWork demandsToWork) {
        this.demandsToWork = demandsToWork;
    }
}

```

Клас RecruitingAgency

```
public class RecruitingAgency {
    Challenger[] mas = new Challenger[3];

    /**
     * Метод вывода массива данных
     */
    public void printAll() {
        for(int i = 0 ; i < mas.length; i++) {
            System.out.println("ID: " + mas[i].getRegistrationNum());
            System.out.println("Образование: " + mas[i].getEducation());
            System.out.println("Дата увольнения: " +
mas[i].getDismissalDay()+"/"+mas[i].getDismissalMonth()+"/"+mas[i].getDismissalYear()
);
            System.out.println("---Опыт работы---");
            System.out.println("Место предыдущей работы: " +
mas[i].getWorkExperience().getSpecialization());
            if(mas[i].getWorkExperience().getExperience() <= 4)
                System.out.println("Стаж: " +
mas[i].getWorkExperience().getExperience() + " год(а)");
            else
                System.out.println("Стаж: " +
mas[i].getWorkExperience().getExperience() + " лет");
            System.out.println("---Желания по будущей работе---" );
            if(mas[i].getDemandsToWork().getMinSalary() == 0 &&
mas[i].getDemandsToWork().getSpecialization() == null &&
mas[i].getDemandsToWork().getConditions() == null)
                System.out.println("Претендент не имеет никаких желаний по
будущей работе");
            else {
                if( mas[i].getDemandsToWork().getMinSalary() != 0)
                    System.out.println("Желаемая минимальная зарплата: "
+ mas[i].getDemandsToWork().getMinSalary());
                else
                    System.out.println("Желаемая минимальная зарплата:
Претендент не имеет пожеланий к этому пункту " );
                if(mas[i].getDemandsToWork().getSpecialization() != null)
                    System.out.println("Желаемая будущая работа: " +
mas[i].getDemandsToWork().getSpecialization());
                else
                    System.out.println("Желаемая будущая работа:
Претендент не имеет пожеланий к этому пункту");
                if(mas[i].getDemandsToWork().getConditions() != null)
                    System.out.println("Желаемые условия будущей работы:
" + mas[i].getDemandsToWork().getConditions());
                else
                    System.out.println("Желаемые условия будущей работы:
Претендент не имеет пожеланий к этому пункту");
            }
            System.out.println("-----");
        }
    }
}
```

Клас DemandsToWork

```
public class DemandsToWork {
    private String specialization;
    private int minSalary;
    private String conditions;
```

```

    public DemandsToWork(String specialization, int minSalary, String conditions)
    {
        this.specialization =specialization;
        this.minSalary = minSalary;
        this.conditions = conditions;
    }

    public String getSpecialization() {
        return specialization;
    }
    public void setSpecialization(String specialization) {
        this.specialization = specialization;
    }

    public int getMinSalary() {
        return minSalary;
    }
    public void setMinSalary(int minSalary) {
        this.minSalary = minSalary;
    }

    public String getConditions() {
        return conditions;
    }
    public void setConditions(String conditions){
        this.conditions = conditions;
    }
}

```

Клас WorkExpereince

```

public class WorkExperience {
    private String specialization;
    private int experience;

    public WorkExperience (String specialization, int experience)
    {
        this.specialization = specialization;
        this.experience = experience;
    }

    public String getSpecialization() {
        return specialization;
    }
    public void setSpecialization(String specialization) {
        this.specialization = specialization;
    }

    public int getExperience() {
        return experience;
    }
    public void setExperience(int experience) {
        this.experience = experience;
    }
}

```

РЕЗУЛЬТАТ ВИКОНАННЯ РОБОТИ ПРОГРАММИ

```
ID: 39821
Образование: Среднее не полное
Дата увольнения: 13/5/2020
---Опыт работы---
Место предыдущей работы: Кассир
Стаж: 4 год(а)
---Желания по будущей работе---
Желаемая минимальная зарплата: 7800
Желаемая будущая работа: Менеджер по продажам
Желаемые условия будущей работы: Наличие кофеварки на работе.
-----
ID: 15301
Образование: Высшее образование
Дата увольнения: 15/10/2014
---Опыт работы---
Место предыдущей работы: Учитель
Стаж: 14 лет
---Желания по будущей работе---
Претендент не имеет никаких желаний по будущей работе
-----
ID: 3004
Образование: Высшее образование
Дата увольнения: 14/3/2020
---Опыт работы---
Место предыдущей работы: Бухгалтер
Стаж: 38 лет
---Желания по будущей работе---
Желаемая минимальная зарплата: 15000
Желаемая будущая работа: Бухгалтер
Желаемые условия будущей работы: Офис в центре.
-----
```

Рисунок 7.1 – Результат работы програми

```
ID: 39821
Образование: Среднее не полное
Дата увольнения: 13/5/2020
---Опыт работы---
Место предыдущей работы: Кассир
Стаж: 4 год(а)
---Желания по будущей работе---
Желаемая минимальная зарплата: 7800
Желаемая будущая работа: Менеджер по продажам
Желаемые условия будущей работы: Наличие кофеварки на работе.
-----
ID: 15301
Образование: Высшее образование
Дата увольнения: 15/10/2014
---Опыт работы---
Место предыдущей работы: Учитель
Стаж: 14 лет
---Желания по будущей работе---
Желаемая минимальная зарплата: 9300
Желаемая будущая работа: Претендент не имеет пожеланий к этому пункту
Желаемые условия будущей работы: Претендент не имеет пожеланий к этому пункту
-----
ID: 3004
Образование: Высшее образование
Дата увольнения: 14/3/2020
---Опыт работы---
Место предыдущей работы: Бухгалтер
Стаж: 38 лет
---Желания по будущей работе---
Желаемая минимальная зарплата: 15000
Желаемая будущая работа: Бухгалтер
Желаемые условия будущей работы: Рядом с метро.
-----
```

Рисунок 7.2 – Результат работы програми

Висновок

Під час виконання лабораторної роботи було набуто навички роботи з domain-об'єктами та об'єктно-орієнтовною декомпозицією у середовищі Eclipse IDE.