

## Лабораторна робота №3

### ООП.

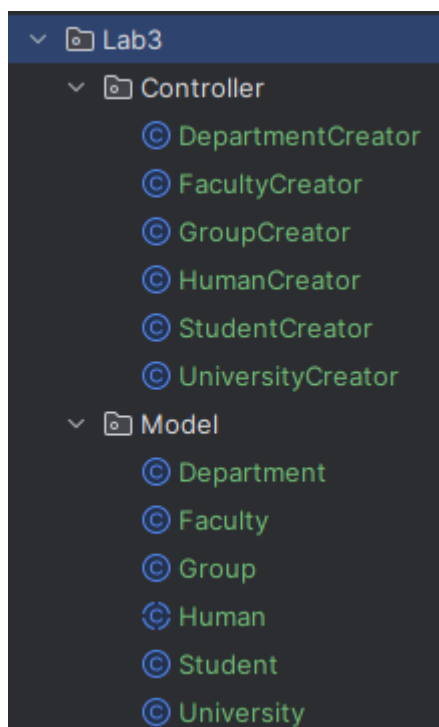
#### Хід роботи

Створити програму що буде створювати та обробляти комплексний об'єкт під назвою університет(university). Програма повинна складатися з трьох частин: модель вид та контролер згідно з парадигмою mvc (Model View Controller). Кожній з цих груп повинна відповідати package з відповідною назвою. В моделі повинні знаходитись усі класи що відповідають за структурні підрозділи університету. Серед них: університет, факультет, кафедра, група, студент, людина (Human). Усі вони повинні містити назву типу string та голову типу Human. Студент також повинен бути породжений від Human. Human повинен мати поля ім'я, прізвище, побатькові та стать. Усі поля повинні бути строковими окрім поля стать. Стать повинна використовувати спеціальний enum типу Sex(стать).

В цій лабораторній роботі група View Нам не потрібна.

Що стосується групи контроллер (controller) то вона повинна містити менеджери що дозволяють нам створити відповідні підрозділи наприклад StudentCreator, FacultyCreator, GroupCreator та інші, кожен з яких повинен використовувати можливості нижчого за рівнем створювача. Програма повинна також містити клас Run, в якому буде знаходитись точка входу та методи, що повинні дати можливість створити університет. Процес створення університету повинен бути зроблений в методі createTypycalUniversity.

В програмі активно рекомендується використовувати абстрактні класи та інтерфейси



Model:

## Department

```
package org.KostiuchenkoD.Lab3.Model;

import java.util.ArrayList;
import java.util.List;

public class Department {
    private String name;
    private Human head;
    private List<Group> groups;

    public Department(String name, Human head) {
        this.name = name;
        this.head = head;
        this.groups = new ArrayList<>();
    }

    public void addGroup(Group group) {
        groups.add(group);
    }

    public String getName() {
        return name;
    }

    public Human getHead() {
        return head;
    }

    public List<Group> getGroups() {
        return groups;
    }

    @Override
    public String toString() {
        return "Department: " + name + ", Head: " + head + ", Groups: " +
groups;
    }
}
```

## Faculty

```
package org.KostiuchenkoD.Lab3.Model;

import java.util.ArrayList;
import java.util.List;

public class Faculty {
    private String name;
    private Human head;
    private List<Department> departments;
```

```

public Faculty(String name, Human head) {
    this.name = name;
    this.head = head;
    this.departments = new ArrayList<>();
}

public void addDepartment(Department department) {
    departments.add(department);
}

public String getName() {
    return name;
}

public Human getHead() {
    return head;
}

public List<Department> getDepartments() {
    return departments;
}

@Override
public String toString() {
    return "Faculty: " + name + ", Head: " + head + ", Departments: " +
departments;
}
}

```

## Group

```

package org.KostiuchenkoD.Lab3.Model;

import java.util.ArrayList;
import java.util.List;

public class Group {
    private String name;
    private Human head;
    private List<Student> students;

    public Group(String name, Human head) {
        this.name = name;
        this.head = head;
        this.students = new ArrayList<>();
    }

    public void addStudent(Student student) {
        students.add(student);
    }

    public String getName() {
        return name;
    }

    public Human getHead() {
        return head;
    }
}

```

```

    public List<Student> getStudents() {
        return students;
    }

    @Override
    public String toString() {
        return "Group: " + name + ", Head: " + head + ", Students: " +
students;
    }
}

```

## Human

```

package org.KostiuchenkoD.Lab3.Model;

```

```

public abstract class Human {
    private String firstName;
    private String lastName;
    private String patronymic;
    private Sex sex;

    public Human(String firstName, String lastName, String patronymic, Sex
sex) {
        this.firstName = firstName;
        this.lastName = lastName;
        this.patronymic = patronymic;
        this.sex = sex;
    }

    public String getFirstName() {
        return firstName;
    }

    public String getLastName() {
        return lastName;
    }

    public String getPatronymic() {
        return patronymic;
    }

    public Sex getSex() {
        return sex;
    }

    public enum Sex {
        MALE, FEMALE
    }

    @Override
    public String toString() {
        return lastName + " " + firstName + " " + patronymic + " (" + sex +
") ";
    }
}

```

## Student

```
package org.KostiuchenkoD.Lab3.Model;

public class Student extends Human {
    public Student(String firstName, String lastName, String patronymic, Sex
sex) {
        super(firstName, lastName, patronymic, sex);
    }
    @Override
    public String toString() {
        return super.toString();
    }
}
```

## University

```
package org.KostiuchenkoD.Lab3.Model;

import java.util.ArrayList;
import java.util.List;

public class University {
    private String name;
    private Human head;
    private List<Faculty> faculties;

    public University(String name, Human head) {
        this.name = name;
        this.head = head;
        this.faculties = new ArrayList<>();
    }

    public void addFaculty(Faculty faculty) {
        faculties.add(faculty);
    }

    public String getName() {
        return name;
    }

    public Human getHead() {
        return head;
    }

    public List<Faculty> getFaculties() {
        return faculties;
    }

    @Override
    public String toString() {
        return "University: " + name + ", Head: " + head + ", Faculty: " +
faculties;
    }
}
```

## Controller:

## DepartmentCreator

```
package org.KostiuchenkoD.Lab3.Controller;

import org.KostiuchenkoD.Lab3.Model.*;

public class DepartmentCreator {
    public Department createDepartment(String name, Human head) {
        return new Department(name, head);
    }
    public void addGroup(Department department, Group group) {
        department.addGroup(group);
    }
}
```

## FacultyCreator

```
package org.KostiuchenkoD.Lab3.Controller;

import org.KostiuchenkoD.Lab3.Model.*;

public class FacultyCreator {
    public Faculty createFaculty(String name, Human head) {
        return new Faculty(name, head);
    }
    public void addDepartment(Faculty faculty, Department department) {
        faculty.addDepartment(department);
    }
}
```

## GroupCreator

```
package org.KostiuchenkoD.Lab3.Controller;

import org.KostiuchenkoD.Lab3.Model.*;

public class GroupCreator {
    public Group createGroup(String name, Human head) {
        return new Group(name, head);
    }
    public void addStudent(Group group, Student student) {
        group.addStudent(student);
    }
}
```

## HumanCreator

```
package org.KostiuchenkoD.Lab3.Controller;

import org.KostiuchenkoD.Lab3.Model.Human;

public class HumanCreator {
    public Human createHuman(String firstName, String lastName, String
patronymic, Human.Sex sex) {
        return new Human(firstName, lastName, patronymic, sex) {};
    }
}
```

## StudentCreator

```
package org.KostiuchenkoD.Lab3.Controller;

import org.KostiuchenkoD.Lab3.Model.*;

public class StudentCreator {
    public Student createStudent(String firstName, String lastName, String
patronymic, Human.Sex sex) {
        return new Student(firstName, lastName, patronymic, sex);
    }
}
```

## UniversityCreator

```
package org.KostiuchenkoD.Lab3.Controller;

import org.KostiuchenkoD.Lab3.Model.*;

public class UniversityCreator {
    public University createUniversity(String name, Human head) {
        return new University(name, head);
    }
    public void addFaculty(University university, Faculty faculty) {
        university.addFaculty(faculty);
    }
}
```

## Run.Main

```
class Run {
    public static void Main() {
        University university = createTypicalUniversity();
        System.out.println(university);
    }

    public static University createTypicalUniversity() {
        // Створення людей
        HumanCreator humanCreator = new HumanCreator();
        Human rector = humanCreator.createHuman("Денис", "Костюченко",
"Олександрович", Human.Sex.MALE);
        Human dean = humanCreator.createHuman("Олена", "Сидорова",
"Олексіївна", Human.Sex.FEMALE);
        Human departmentHead = humanCreator.createHuman("Михайло",
"Коваленко", "Михайлович", Human.Sex.MALE);
        Human groupLeader = humanCreator.createHuman("Анна", "Іванова",
"Петрівна", Human.Sex.FEMALE);

        // Створення студентів
        StudentCreator studentCreator = new StudentCreator();
        Student student1 = studentCreator.createStudent("Олег", "Коваль",
"Олегович", Human.Sex.MALE);
        Student student2 = studentCreator.createStudent("Марія", "Шевченко",
"Ігорівна", Human.Sex.FEMALE);

        // Створення групи
        GroupCreator groupCreator = new GroupCreator();
        Group group = groupCreator.createGroup("Група 123-21-1",
groupLeader);
    }
}
```

```

group.addStudent(student1);
group.addStudent(student2);

// Створення кафедри
DepartmentCreator departmentCreator = new DepartmentCreator();
Department department = departmentCreator.createDepartment("Кафедра
інформаційних технологій та комп'ютерної інженерії", departmentHead);
department.addGroup(group);

// Створення факультету
FacultyCreator facultyCreator = new FacultyCreator();
Faculty faculty = facultyCreator.createFaculty("Факультет
інформаційних технологій", dean);
faculty.addDepartment(department);

// Створення університету
UniversityCreator universityCreator = new UniversityCreator();
University university =
universityCreator.createUniversity("Національний технічний університет
«Дніпровська політехніка», rector);
university.addFaculty(faculty);

return university;
}
}

```

### Результат

University: Національний університет, Head: Костюченко Денис Олександрович (MALE), Faculty: [Faculty: Факультет інформаційних технологій, Head: Сидорова Олена Олексіївна (FEMALE), Departments: [Department: Кафедра інформатики, Head: Коваленко Михайло Михайлович (MALE), Groups: [Group: Група 101, Head: Іванова Анна Петрівна (FEMALE), Students: [Коваль Олег Олегович (MALE), Шевченко Марія Ігорівна (FEMALE)]]]]]