

ЛАБОРАТОРНА РОБОТА № 2

Тема: Створення структури класу заданої предметної області.

Мета роботи: створити ієрархію класів заданої предметної області, робота з статичними методами.

Завдання на лабораторну роботу

Завдання 1. Створити консольний Java проект `java_lab_2_oop` з пакетом `com.education.ztu`

Завдання 2. Створити ієрархію класів відповідно до UML діаграми:

- поля класів повинні бути приховані модифікаторами доступу `private`, `protected`;
- створити конструктор без аргументів та з аргументами;
- створити блок ініціалізації, в якому ініціалізуються значення полів за замовчуванням у разі, якщо викликається конструктор без аргументів;
- створити геттери та сеттери для полів;
- створити статичну змінну `counter` для підрахунку створених екземплярів даного класу та статичний метод `showCounter` для відображення значення змінної `counter`.
- створити `enum Location` та `Gender` і використати їх в полях класів.
- створити інтерфейс `Human` з методами `sayFullName`, `sayAge`, `sayLocation`, `sayGender` та `whoAm` (default)
- створити абстрактний клас `Person` з абстрактним методом `getOccupation` та звичайним методом `getFullInfo`, що імплементує `Human`;
- створити класу `Student`, `Teacher`, `Employee`, що наслідують `Person` та перевизначити необхідні методи та створити свої.
- для `Teacher`, `Employee` додати поле `Car`, що є об'єктом відповідного класу.
- створити в `Car` внутрішній клас `Engine` з методами `startEngine`, `stopEngine`, `isEngineWorks` та реалізувати їх логіку.
- додати до описаної функціональності свою (нові поля та методи).
- в методі `main` класу `Main` створити об'єкти відповідних класів та продемонструвати роботу їх методів.
- продемонструвати роботу оператору `instanceof`.

Лістинг програми

```
package com.education.ztu;
```

```
public class Car {  
    private String brand;  
    private Engine engine;  
  
    public Car(String brand) {  
        this.brand = brand;  
        this.engine = new Engine();  
    }  
}
```

					ДУ «Житомирська політехніка».25.121.01.000 – Лр1			
Змн.	Арк.	№ докум.	Підпис	Дата				
Розроб.		Айсін В.С			Звіт з лабораторної роботи	Літ.	Арк.	Аркушів
Перевір.		Піонтьківський В І					1	11
Керівник						ФІКТ Гр. ІПЗ-23-1[1]		
Н. контр.								
Зав. каф.								

```

public boolean engineIsRunning() {
    return engine.isEngineWorks();
}

public Engine getEngine() {
    return engine;
}

public class Engine {
    private boolean engineWorks;

    public void startEngine() {
        engineWorks = true;
        System.out.println("Engine started.");
    }

    public void stopEngine() {
        engineWorks = false;
        System.out.println("Engine stopped.");
    }

    public boolean isEngineWorks() {
        return engineWorks;
    }
}
}
package com.education.ztu;

public class Employee extends Person {
    private String company="";
    private String position="";
    private Car car=null;
    public Employee() {
        super();
    }
    public Employee(String firstname, String lastname, int age, Gender gender, Location location, String company,
String position, Car car) {
        super(firstname, lastname, age, gender, location);
        this.company = company;
        this.position = position;
        this.car = car;
    }
    public String getOccupation() {
        return "Company: " + this.company + ", Position: " +
            this.position;
    }
    public Car getCar() {
        return car;
    }
}
package com.education.ztu;

public enum Gender {
    MALE,
    FEMALE;
}
package com.education.ztu;

interface Human {
    void sayName();

    void sayAge();

    void sayLocation();

    void sayGender();

    default void whoIAM() {
        System.out.println("Human.");
    }
}
package com.education.ztu;

enum Location {
    KIEV, ZHYTOMYR, VINNYTSYA, RIVNE;
}

```

		Айсін В.С			ДУ «Житомирська політехніка».25.121.01.000 – Лр1	Арк.
		Піонкірський В І				2
Змн.	Арк.	№ докум.	Підпис	Дата		

```

package com.education.ztu;

public class Main {
    public static void main(String[] args) {
        Student student = new Student("Michael", "Brown", 22, Gender.MALE,
            Location.KIEV, "Software Engineering", "MIT");
        Teacher teacher = new Teacher("Emily", "Davis", 50, Gender.FEMALE,
            Location.ZHYTOMYR, "Physics", "Harvard", new Car("Ford"));
        Employee employee = new Employee("Daniel", "Clark", 35, Gender.MALE,
            Location.VINNYTSYA, "InnoTech", "Project Manager", new Car("Tesla"));
        student.sayName();
        teacher.sayAge();
        employee.sayLocation();
        System.out.println(student.getFullInfo());
        System.out.println(teacher.getFullInfo());
        System.out.println(employee.getFullInfo());
        System.out.println(student.getOccupation());
        System.out.println(teacher.getOccupation());
        System.out.println(employee.getOccupation());
        System.out.println("Teacher is a person.");
        teacher.getCar().getEngine().startEngine();
        System.out.println("Is engine running? " + teacher.getCar().engineIsRunning());
    }
}

package com.education.ztu;

abstract class Person implements Human {
    protected String firstname = "";
    protected String lastname = "";
    protected int age = 0;
    protected Gender gender = null;
    protected Location location = null;
    protected static int counter = 0;
    {
        counter++;
    }
    public Person() {}
    public Person(String firstname, String lastname, int age, Gender gender, Location location) {
        this.firstname = firstname;
        this.lastname = lastname;
        this.age = age;
        this.gender = gender;
        this.location = location;
    }
    public abstract String getOccupation();
    public String getFullInfo() {
        return firstname + " " + lastname + ", Age: " + age + ", Gender: " + gender + ", Location: " + location;
    }
    public static void showCounter() {
        System.out.println("Persons: " + counter);
    }
    public void sayName() {
        System.out.println(this.firstname + " " + this.lastname);
    }
    public void sayAge() {
        System.out.println("Age: " + this.age);
    }
    public void sayLocation() {
        System.out.println("Location: " + this.location);
    }
    public void sayGender() {
        System.out.println("Gender: " + this.gender);
    }
}

package com.education.ztu;

public class Student extends Person {
    private String speciality="";
    private String university="";
    public Student() {
        super();
    }
    public Student(String firstname, String lastname, int age, Gender gender, Location location, String special-
ity, String university) {
        super(firstname, lastname, age, gender, location);
        this.speciality = speciality;

```

		Айсін В.С			ДУ «Житомирська політехніка».25.121.01.000 – Лр1	Арк.
		Піонківський В І				3
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        this.university = university;
    }
    public String getOccupation() {
        return "Speciality: " + this.speciality + ", University: " + this.university;
    }
}
package com.education.ztu;

public class Teacher extends Person {
    private String subject="";
    private String university="";
    private Car car=null;
    public Teacher() {
        super();
    }
    public Teacher(String firstname, String lastname, int age, Gender gender, Location location, String subject,
String university, Car car){
        super(firstname, lastname, age, gender, location);
        this.subject = subject;
        this.university = university;
        this.car = car;
    }
    public String getOccupation() {
        return "Subject: " + this.subject + ", University: " + this.university;
    }
    public Car getCar() {
        return car;
    }
}
}

```

Результат виконання

```

Michael Brown
Age: 50
Location: VINNYTSYA
Michael Brown, Age: 22, Gender: MALE, Location: KIEV
Emily Davis, Age: 50, Gender: FEMALE, Location: ZHYTOMYR
Daniel Clark, Age: 35, Gender: MALE, Location: VINNYTSYA
Speciality: Software Engineering, University: MIT
Subject: Physics, University: Harvard
Company: InnoTech, Position: Project Manager
Teacher is a person.
Engine started.
Is engine running? true

```

Main

Завдання 3. Створити клас Operation з статичними методами addition, subtraction, multiplication, division, average, maximum, minimum, що приймають необмежену кількість аргументів через varargs.

в методі main класу Main2 продемонструвати роботу методів класу Operation

- вивести всі значення enam Location.

```

public class Main2 {
    public static void main(String[] args) {
        int[] numbers = {1, 2, 3, 4};
    }
}

```

		Айсін В.С			ДУ «Житомирська політехніка».25.121.01.000 – Лр1	Арк.
		Піонтиківський В І				4
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        System.out.println("Addition: " + Operation.addition(numbers));
        System.out.println("Subtraction: " + Operation.subtraction(numbers));
        System.out.println("Multiplication: " + Operation.multiplication(numbers));
        System.out.println("Division: " + Operation.division(numbers));
        System.out.println("Average: " + Operation.average(numbers));
        System.out.println("Maximum: " + Operation.maximum(numbers));
        System.out.println("Minimum: " + Operation.minimum(numbers));
        System.out.println("\nAll Location:");
        for (Location location : Location.values()) {
            System.out.println(location);
        }
    }
}
package com.education.ztu;

class Operation {
    public static int addition(int... numbers) {
        int sum = 0;
        for (int num : numbers) {
            sum += num;
        }
        return sum;
    }

    public static int subtraction(int... numbers) {
        if (numbers.length == 0) return 0;
        int result = numbers[0];
        int i = 1;
        while (i < numbers.length) {
            result -= numbers[i];
            i++;
        }
        return result;
    }

    public static int multiplication(int... numbers) {
        if (numbers.length == 0) return 0;
        int result = 1;
        for (int num : numbers) {
            result *= num;
        }
        return result;
    }

    public static double division(int... numbers) {
        if (numbers.length == 0) return 0;
        double result = numbers[0];
        int i = 1;
        while (i < numbers.length) {
            if (numbers[i] != 0) {
                result /= numbers[i];
            } else {
                System.out.println("Cannot divide by zero.");
                return 0;
            }
            i++;
        }
        return result;
    }

    public static double average(int... numbers) {
        if (numbers.length == 0) return 0;
        return (double) addition(numbers) / numbers.length;
    }

    public static int maximum(int... numbers) {
        if (numbers.length == 0) return Integer.MIN_VALUE;
        int max = numbers[0];
        int i = 1;
        while (i < numbers.length) {
            if (numbers[i] > max) {
                max = numbers[i];
            }
            i++;
        }
        return max;
    }
}

```

		Айсін В.С			ДУ «Житомирська політехніка».25.121.01.000 – Лр1	Арк.
		Піонтківський В І				5
Змн.	Арк.	№ докум.	Підпис	Дата		

```

public static int minimum(int... numbers) {
    if (numbers.length == 0) return Integer.MAX_VALUE;
    int min = numbers[0];
    int i = 1;
    while (i < numbers.length) {
        if (numbers[i] < min) {
            min = numbers[i];
        }
        i++;
    }
    return min;
}
}

```

```

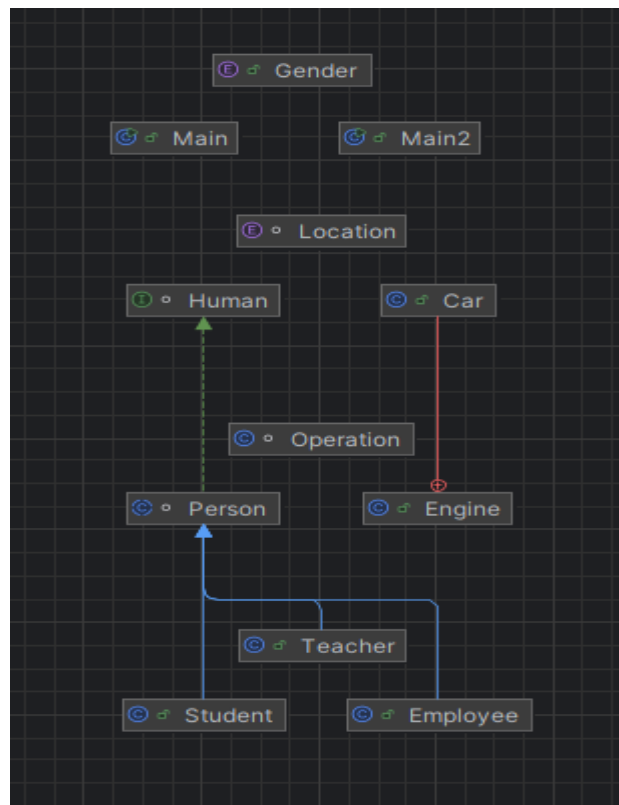
Addition: 10
Subtraction: -8
Multiplication: 24
Division: 0.041666666666666664
Average: 2.5
Maximum: 4
Minimum: 1

All Location:
KIEV
ZHYTOMYR
VINNYTSYA
RIVNE

```

Main2

Завдання 4. Створити UML діаграму створеної структури ієрархії класів та зберегти як картинку.



		Айсін В.С			ДУ «Житомирська політехніка».25.121.01.000 – Лр1	Арк.
		Піонтківський В І				6
Змн.	Арк.	№ докум.	Підпис	Дата		

Завдання 5. В GitLab проекті *Java_labs_ztu*, створити директорію *Lab_2* та запусити в *Lab_2* виконану лабораторну роботу. Надати доступ для перевірки викладачу.

		Айсін В.С			ДУ «Житомирська політехніка».25.121.01.000 – Лр1	Арк.
		Піонтківський В І				7
Змн.	Арк.	№ докум.	Підпис	Дата		