Чернівецький національний університет імені Юрія Федьковича

Інститут фізико-технічних та комп’ютерних наук

Кафедра Математичних проблем управління і кібернетики

Лабораторна робота № 3

“Тема: Спадкування. Інтерфейси.”

з дисципліни “ Java”

Варіант № 18

Виконав:

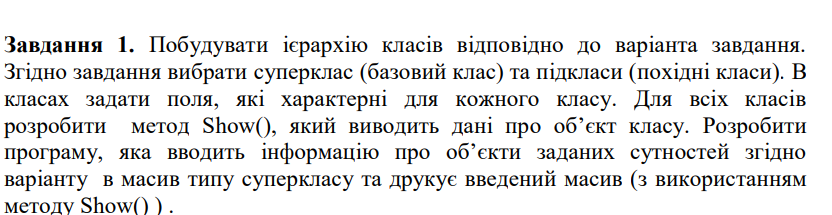
ст. гр. 241 Фрасинюк О. Б.

Прийняв:

Викладач Лазорик В.В.

Чернівці – 2024

https://github.com/VLazorykOOP/jlab3-Oleh-Frasyniuk



3.Робітник,кадри,інженер,адміністрація.

**Код:**

import java.util.Objects;

class Worker {

    protected String name;

    protected int age;

    protected String position;

    public Worker(String name, int age, String position) {

        this.name = name;

        this.age = age;

        this.position = position;

    }

    public void Show() {

        System.out.println("Name: " + name + ", Age: " + age + ", Position: " + position);

    }

}

class Engineer extends Worker {

    private String specialization;

    private int experience;

    public Engineer(String name, int age, String position, String specialization, int experience) {

        super(name, age, position);

        this.specialization = specialization;

        this.experience = experience;

    }

    @Override

    public void Show() {

        super.Show();

        System.out.println("Specialization: " + specialization + ", Experience: " + experience + " years");

    }

}

class Administration extends Worker {

    private String department;

    private String responsibility;

    public Administration(String name, int age, String position, String department, String responsibility) {

        super(name, age, position);

        this.department = department;

        this.responsibility = responsibility;

    }

    @Override

    public void Show() {

        super.Show();

        System.out.println("Department: " + department + ", Responsibility: " + responsibility);

    }

}

class HR extends Worker {

    private int recruitmentCount;

    private boolean policyDevelopment;

    public HR(String name, int age, String position, int recruitmentCount, boolean policyDevelopment) {

        super(name, age, position);

        this.recruitmentCount = recruitmentCount;

        this.policyDevelopment = policyDevelopment;

    }

    @Override

    public void Show() {

        super.Show();

        System.out.println("Recruitment Count: " + recruitmentCount + ", Policy Development: " + policyDevelopment);

    }

}

public class Main {

    public static void main(String[] args) {

        Worker[] workers = new Worker[3];

        workers[0] = new Engineer("Alice", 30, "Senior Engineer", "Software Development", 8);

        workers[1] = new Administration("Bob", 45, "Manager", "Finance", "Budgeting");

        workers[2] = new HR("Carol", 35, "HR Specialist", 50, true);

        for (Worker worker : workers) {

            worker.Show();

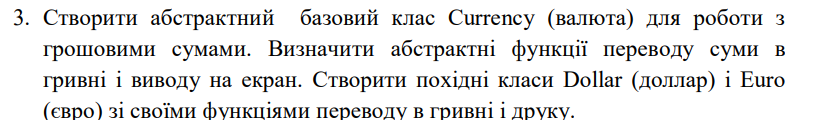
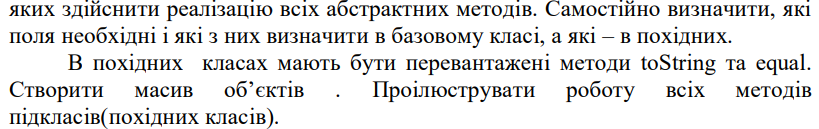
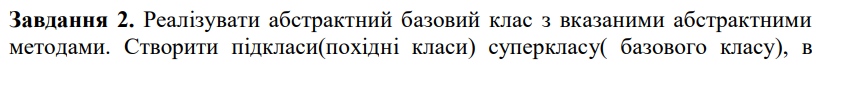
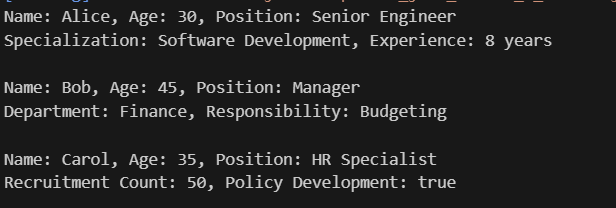
            System.out.println();

        }

    }

}

**Результат:**



**Код:**

abstract class Currency {

    protected double amount;

    public Currency(double amount) {

        this.amount = amount;

    }

    // Абстрактний метод для переводу суми у гривні

    public abstract double toUAH();

    // Абстрактний метод для виводу інформації на екран

    public abstract void display();

    @Override

    public String toString() {

        return "Currency amount: " + amount;

    }

    @Override

    public boolean equals(Object obj) {

        if (this == obj) return true;

        if (obj == null || getClass() != obj.getClass()) return false;

        Currency currency = (Currency) obj;

        return Double.compare(currency.amount, amount) == 0;

    }

}

class Dollar extends Currency {

    private static final double EXCHANGE\_RATE = 41.5; // Курс обміну на гривні

    public Dollar(double amount) {

        super(amount);

    }

    @Override

    public double toUAH() {

        return amount \* EXCHANGE\_RATE;

    }

    @Override

    public void display() {

        System.out.println("Dollar amount: $" + amount + " = " + toUAH() + " UAH");

    }

    @Override

    public String toString() {

        return "Dollar: $" + amount;

    }

    @Override

    public boolean equals(Object obj) {

        if (!super.equals(obj)) return false;

        return obj instanceof Dollar;

    }

}

class Euro extends Currency {

    private static final double EXCHANGE\_RATE = 44.0; // Курс обміну на гривні

    public Euro(double amount) {

        super(amount);

    }

    @Override

    public double toUAH() {

        return amount \* EXCHANGE\_RATE;

    }

    @Override

    public void display() {

        System.out.println("Euro amount: e" + amount + " = " + toUAH() + " UAH");

    }

    @Override

    public String toString() {

        return "Euro: e" + amount;

    }

    @Override

    public boolean equals(Object obj) {

        if (!super.equals(obj)) return false;

        return obj instanceof Euro;

    }

}

public class Main {

    public static void main(String[] args) {

        Currency[] currencies = new Currency[4];

        currencies[0] = new Dollar(100);

        currencies[1] = new Euro(50);

        currencies[2] = new Dollar(200);

        currencies[3] = new Euro(80);

        for (Currency currency : currencies) {

            currency.display();

            System.out.println(currency.toString());

            System.out.println();

        }

        // Перевірка equals

        System.out.println("Equality check:");

        System.out.println(currencies[0].equals(currencies[2])); // Dollar 100 == Dollar 200

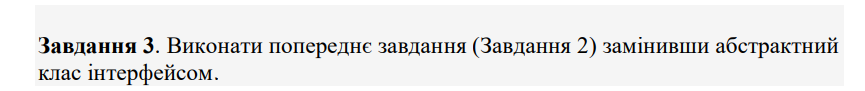
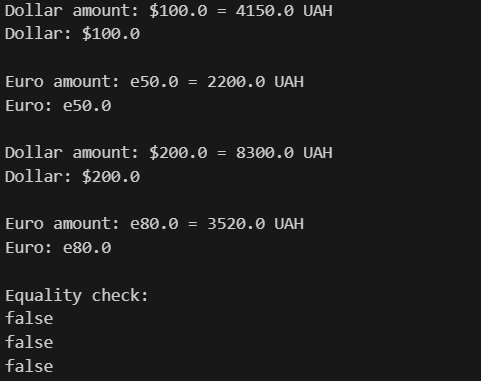
        System.out.println(currencies[1].equals(currencies[3])); // Euro 50 == Euro 80

        System.out.println(currencies[0].equals(currencies[1])); // Dollar != Euro

    }

}

**Результат:**



**Код:**

interface Currency {

    double toUAH();

    void display();

}

class Dollar implements Currency {

    private static final double EXCHANGE\_RATE = 40.5; // Курс обміну на гривні

    private double amount;

    public Dollar(double amount) {

        this.amount = amount;

    }

    @Override

    public double toUAH() {

        return amount \* EXCHANGE\_RATE;

    }

    @Override

    public void display() {

        System.out.println("Dollar amount: $" + amount + " = " + toUAH() + " UAH");

    }

    @Override

    public String toString() {

        return "Dollar: $" + amount;

    }

    @Override

    public boolean equals(Object obj) {

        if (this == obj) return true;

        if (obj == null || getClass() != obj.getClass()) return false;

        Dollar dollar = (Dollar) obj;

        return Double.compare(dollar.amount, amount) == 0;

    }

}

class Euro implements Currency {

    private static final double EXCHANGE\_RATE = 44.0; // Курс обміну на гривні

    private double amount;

    public Euro(double amount) {

        this.amount = amount;

    }

    @Override

    public double toUAH() {

        return amount \* EXCHANGE\_RATE;

    }

    @Override

    public void display() {

        System.out.println("Euro amount: e" + amount + " = " + toUAH() + " UAH");

    }

    @Override

    public String toString() {

        return "Euro: e" + amount;

    }

    @Override

    public boolean equals(Object obj) {

        if (this == obj) return true;

        if (obj == null || getClass() != obj.getClass()) return false;

        Euro euro = (Euro) obj;

        return Double.compare(euro.amount, amount) == 0;

    }

}

public class Main {

    public static void main(String[] args) {

        Currency[] currencies = new Currency[4];

        currencies[0] = new Dollar(100);

        currencies[1] = new Euro(50);

        currencies[2] = new Dollar(200);

        currencies[3] = new Euro(80);

        for (Currency currency : currencies) {

            currency.display();

            System.out.println(currency.toString());

            System.out.println();

        }

        // Перевірка equals

        System.out.println("Equality check:");

        System.out.println(currencies[0].equals(currencies[2])); // Dollar 100 == Dollar 200

        System.out.println(currencies[1].equals(currencies[3])); // Euro 50 == Euro 80

        System.out.println(currencies[0].equals(currencies[1])); // Dollar != Euro

    }

}

**Результат:**

