

**Державний вищий навчальний заклад Ужгородський національний
університет Факультет інформаційних технологій**

Лабораторна робота № 2

Тема: класи та об'єкти в Java.

Виконала студентка I курсу спеціальності
«Інженерія програмного забезпечення»
Кіндрат Неля Анатоліївна

Ужгород-2024

Варіант 4

Завдання до роботи:

Спроекувати клас який забезпечить представлення у програмі даних про робітника.

*Даний клас планується використовувати в програмі для нарахування зар.плати

Продемонструвати використання.

Хід роботи

Worker.java

```
public class Worker {  
    private final String name;  
    private final String position;  
    private double hourlyRate;  
    private double hoursWorked;  
  
    // Конструктор  
    public Worker(String name, String position, double hourlyRate) {  
        this.name = name;  
        this.position = position;  
        this.hourlyRate = hourlyRate;  
        this.hoursWorked = 0;  
    }  
  
    // Додавання відпрацьованих годин  
    public void addHours(double hours) {  
        if (hours > 0) {
```

```
        this.hoursWorked += hours;
    } else {
        System.out.println("Час повинен бути позитивним числом.");
    }
}

// Метод для зміни кількості відпрацьованих годин
public void setHoursWorked(double hours) {
    if (hours >= 0) {
        this.hoursWorked = hours;
    } else {
        System.out.println("Кількість годин не може бути негативною.");
    }
}

// Метод для зміни почасової ставки
public void setHourlyRate(double hourlyRate) {
    if (hourlyRate >= 0) {
        this.hourlyRate = hourlyRate;
    } else {
        System.out.println("Почасова ставка не може бути негативною.");
    }
}

// Метод для розрахунку заробітної плати
public double calculateSalary() {
    return hourlyRate * hoursWorked;
}
```

```
}

// Отримання інформації про робітника
public String getInfo() {
    return "Ім'я: " + name + ", Посада: " + position + ", Почасова ставка: " +
    hourlyRate + ", Відпрацьовані години: " + hoursWorked;
}
}
```

Main.java

```
import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        List<Worker> workers = new ArrayList<>();
        String userChoice;

        // Додавання робітників
        do {
            // Введення даних про робітника
            System.out.print("Введіть ім'я робітника: ");
            String name = scanner.nextLine();

            System.out.print("Введіть посаду робітника: ");
```

```
String position = scanner.nextLine();

System.out.print("Введіть почасову ставку: ");
double hourlyRate = scanner.nextDouble();

// Створення об'єкта worker та додавання до списку
Worker worker = new Worker(name, position, hourlyRate);
workers.add(worker);

// Введення відпрацьованих годин
System.out.print("Введіть кількість відпрацьованих годин: ");
double hours = scanner.nextDouble();
worker.addHours(hours);

// Вивід інформації про робітника
System.out.println(worker.getInfo());
System.out.println("Зарплата: " + worker.calculateSalary());

// Запит на повторне введення
System.out.print("Бажаєте додати ще одного робітника? (так/ні): ");
scanner.nextLine(); // Очистити буфер
userChoice = scanner.nextLine();

} while (userChoice.equalsIgnoreCase("так"));

// Цикл для зміни даних про робітників
do {
```

```
System.out.print("Бажаєте змінити інформацію про робітника?  
(так/ні): ");  
  
userChoice = scanner.nextLine();  
  
if (userChoice.equalsIgnoreCase("так")) {  
    System.out.print("Введіть номер робітника (1-" + workers.size() +  
    "): ");  
  
    int workerIndex = scanner.nextInt() - 1;  
  
    if (workerIndex >= 0 && workerIndex < workers.size()) {  
        Worker workerToUpdate = workers.get(workerIndex);  
  
        System.out.println("Поточна інформація: " +  
workerToUpdate.getInfo());  
  
        // Запит на нову кількість відпрацьованих годин  
        System.out.print("Введіть нову кількість відпрацьованих годин:  
");  
  
        double newHours = scanner.nextDouble();  
        workerToUpdate.setHoursWorked(newHours);  
  
        // Запит на нову почасову ставку  
        System.out.print("Введіть нову почасову ставку: ");  
        double newHourlyRate = scanner.nextDouble();  
        workerToUpdate.setHourlyRate(newHourlyRate);  
  
        // Вивід оновленої інформації  
        System.out.println("Оновлена інформація: " +  
workerToUpdate.getInfo());
```

```
        System.out.println("Оновлена зарплата: " +  
workerToUpdate.calculateSalary());  
    } else {  
        System.out.println("Неправильний номер робітника.");  
    }  
    scanner.nextLine(); // Очистити буфер  
}  
} while (userChoice.equalsIgnoreCase("так"));  
  
scanner.close();  
}  
}
```

```
Worker.java x Main.java
1 public class Worker { 4 usages new *
2     private final String name; 2 usages
3     private final String position; 2 usages
4     private double hourlyRate; 4 usages
5     private double hoursWorked; 5 usages
6
7     // Конструктор
8     public Worker(String name, String position, double hourlyRate) { 1 usage new *
9         this.name = name;
10        this.position = position;
11        this.hourlyRate = hourlyRate;
12        this.hoursWorked = 0;
13    }
14
15    // Метод для додавання відпрацьованих годин
16    public void addHours(double hours) { 1 usage new *
17        if (hours > 0) {
18            this.hoursWorked += hours;
19        } else {
20            System.out.println("Час повинен бути позитивним числом.");
21        }
22    }
23
24    // Метод для зміни кількості відпрацьованих годин
25    public void setHoursWorked(double hours) { 1 usage new *
26        if (hours >= 0) {
27            this.hoursWorked = hours;
28        } else {
```

```
Worker.java x Main.java
1 public class Worker { 4 usages new *
25    public void setHoursWorked(double hours) { 1 usage new *
26        this.hoursWorked = hours;
27    } else {
28        System.out.println("Кількість годин не може бути негативною.");
29    }
30    }
31
32
33    // Метод для зміни почасової ставки
34    public void setHourlyRate(double hourlyRate) { 1 usage new *
35        if (hourlyRate >= 0) {
36            this.hourlyRate = hourlyRate;
37        } else {
38            System.out.println("Почасова ставка не може бути негативною.");
39        }
40    }
41
42    // Метод для розрахунку заробітної плати
43    public double calculateSalary() { 2 usages new *
44        return hourlyRate * hoursWorked;
45    }
46
47    // Метод для отримання інформації про робітника
48    public String getInfo() { 3 usages new *
49        return "Ім'я: " + name + ", Посада: " + position + ", Почасова ставка: " + hourlyRate + ", Відпрацьовані годин
50    }
51 }
52
```



```
Worker.java Main.java x
1 import java.util.ArrayList;
2 import java.util.List;
3 import java.util.Scanner;
4
5 public class Main { new *
6     public static void main(String[] args) { new *
7         Scanner scanner = new Scanner(System.in);
8         List<Worker> workers = new ArrayList<>();
9         String userChoice;
10
11         // Додавання робітників
12         do {
13             // Введення даних про робітника
14             System.out.print("Введіть ім'я робітника: ");
15             String name = scanner.nextLine();
16
17             System.out.print("Введіть посаду робітника: ");
18             String position = scanner.nextLine();
19
20             System.out.print("Введіть почасову ставку: ");
21             double hourlyRate = scanner.nextDouble();
22
23             // Створення об'єкта Worker та додавання до списку
24             Worker worker = new Worker(name, position, hourlyRate);
25             workers.add(worker);
26
27             // Введення відпрацьованих годин
28             System.out.print("Введіть кількість відпрацьованих годин: ");
```

```
Worker.java Main.java x
5     public class Main { new *
6         public static void main(String[] args) { new *
7             // Введення відпрацьованих годин
8             System.out.print("Введіть кількість відпрацьованих годин: ");
9             double hours = scanner.nextDouble();
10            worker.addHours(hours);
11
12            // Вивід інформації про робітника
13            System.out.println(worker.getInfo());
14            System.out.println("Зарплата: " + worker.calculateSalary());
15
16            // Запит на повторне введення
17            System.out.print("Бажаєте додати ще одного робітника? (так/ні): ");
18            scanner.nextLine(); // Очистити буфер
19            userChoice = scanner.nextLine();
20
21        } while (userChoice.equalsIgnoreCase( anotherString: "так"));
22
23        // Цикл для зміни даних про робітників
24        do {
25            System.out.print("Бажаєте змінити інформацію про робітника? (так/ні): ");
26            userChoice = scanner.nextLine();
27
28            if (userChoice.equalsIgnoreCase( anotherString: "так")) {
29                System.out.print("Введіть номер робітника (1-" + workers.size() + "): ");
30                int workerIndex = scanner.nextInt() - 1;
31
32                if (workerIndex >= 0 && workerIndex < workers.size()) {
```

```

C:\Users\Nelia\IdeaProjects\untitled\src\Worker.java
6      public static void main(String[] args) { new *
51
52          if (workerIndex >= 0 && workerIndex < workers.size()) {
53              Worker workerToUpdate = workers.get(workerIndex);
54              System.out.println("Поточна інформація: " + workerToUpdate.getInfo());
55
56              // Запит на нову кількість відпрацьованих годин
57              System.out.print("Введіть нову кількість відпрацьованих годин: ");
58              double newHours = scanner.nextDouble();
59              workerToUpdate.setHoursWorked(newHours);
60
61              // Запит на нову почасову ставку
62              System.out.print("Введіть нову почасову ставку: ");
63              double newHourlyRate = scanner.nextDouble();
64              workerToUpdate.setHourlyRate(newHourlyRate);
65
66              // Вивід оновленої інформації
67              System.out.println("Оновлена інформація: " + workerToUpdate.getInfo());
68              System.out.println("Оновлена зарплата: " + workerToUpdate.calculateSalary());
69          } else {
70              System.out.println("Неправильний номер робітника.");
71          }
72          scanner.nextLine(); // Очистити буфер
73      }
74      } while (userChoice.equalsIgnoreCase( anotherString: "так"));
75
76      scanner.close();

```

- **Клас Worker:** зберігає інформацію про робітника, включаючи ім'я, позицію, почасову ставку та відпрацьовані години. Дозволяє змінювати години та зарплату.
- **Клас Main:** забезпечує взаємодію з користувачем для введення даних про робітників, додавання нових записів і зміни інформації про існуючих робітників.

В кодї використовується клас **Scanner** для зчитування вводу з консолі.