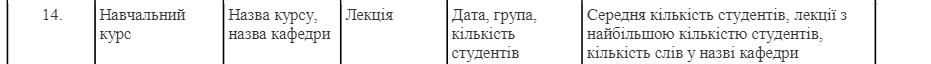
**Мета:** Вивчити особливості роботи з класами та їх спадкуванням у Java.

**Обладнання:** ПК, програмне забезпечення Intellij IDEA, методичні вказівки та завдання до лабораторної роботи.

**Завдання.**

1. Розробити консольний застосунок для роботи з базою даних, що зберігається у текстовому файлі (початковий масив не менше 5записів). Структура бази даних описується ієрархією класівзгідно вашого варіанта. Для ідентифікації спроби введення з клавіатури некоректних даних описати виключення. Реалізувати методи у базовому класідля:–додавання записів; –редагування записів; –знищення записів; –виведення інформації з файла на екран; –обчислення та виведення на екран результатів згідно свого варіанта індивідуального завдання.Меню програми реалізувати по натисненню на певні клавіші: наприклад, Enter –вихід, п -пошук, р –редагування тощо.Один з методівіндивідуального завдання зробити віртуальним.(7балів)

2. Модифікуватиконсольний застосунок з попереднього завдання таким чином, щоб:А) Базовий клас бувабстрактним.Б) Реалізація методів індивідуального завдання була перенесена у похідний клас.(3балa)



**[Код на GitHub](https://github.com/ev9en4ik/java/tree/main/lb4.1)**

**[Код на GitHub](https://github.com/ev9en4ik/java/tree/main/lb4.2)**

**Реалізовані класи**

public abstract class Course {  
 String Ncourse;  
 String Ncathedra;  
  
  
  
 public Course () {  
 Ncourse = "";  
 Ncathedra ="";  
 }  
  
 public Course (String Ncourse , String Ncathedra) {  
 this.Ncourse = Ncourse;  
 this.Ncathedra = Ncathedra;  
 }  
 public abstract void input() ;  
  
 public abstract void output() ;  
  
 public String getNcourse() {  
 return Ncourse;  
 }  
  
 public String getNcathedra() {  
 return Ncathedra;  
 }  
  
 public void setNcourse(String Ncourse) {  
 this.Ncourse = Ncourse;  
 }  
  
 public void setNcathedra(String Ncathedra) {  
 this.Ncathedra = Ncathedra;  
 }  
}

public class Lesson extends Course {  
 Dat date;  
 int group;  
 int kilkist;  
  
  
 public Lesson () {  
 date = new Dat();  
 group = 0;  
 kilkist = 0;  
 }  
 public Lesson (int d , int m , int y , int group , int kilkist) {  
 this.date = new Dat(d,m,y);  
 this.group = group;  
 this.kilkist = kilkist;  
 }  
 public Lesson (String Ncouse, String Ncathedra , int d , int m , int y , int group , int kilkist){  
 super(Ncouse , Ncathedra);  
 this.date = new Dat(d,m,y);  
 this.group = group;  
 this.kilkist = kilkist;  
 }

}

**1) Код :**

public void add() throws IOException{  
 Kassa new\_kassa = new Kassa();  
 System.*out*.println("\nВведіть інформацію :");  
 new\_kassa.input();  
 kassa.add(new\_kassa);  
  
 System.*out*.println("\nІнформація додана!");  
  
 this.write\_to\_file("Kassa.txt");  
}

public void edit() throws IOException {  
 count\_note();  
 Scanner scanner = new Scanner(System.*in*);  
 System.*out*.print("\nВведіть порядковий номер квитка, який ви хочете редагувати : ");  
 String line = scanner.nextLine();  
 if(!line.matches("\\d+") || Integer.*parseInt*(line) <= 0 || Integer.*parseInt*(line) > kassa.size()){  
 System.*err*.println("Не вірно введений номер!");  
 this.edit();  
 return;  
 }  
 int nomer = Integer.*parseInt*(line);  
 nomer--;  
  
 System.*out*.println("\nВідредагуйте :");  
 kassa.get(nomer).input();  
 System.*out*.println("\nРедагування пройшло успішно!");  
  
 this.write\_to\_file("Kassa.txt");  
}

public void print() throws IOException {  
 count\_note();  
  
 int i = 0;  
 System.*out*.println("\nІнформація про квитки :");  
 for (var p:  
 kassa) {  
 System.*out*.println("\nКвитки №" + (i + 1) + " :");  
 p.print();  
 i++;  
 }  
}

public void search() throws IOException {  
 count\_note();  
 Scanner scanner = new Scanner(System.*in*);  
 System.*out*.println("\nВведіть назву пункту для пошуку : ");  
 String line = scanner.nextLine();  
 int i = 0;  
 for (var p:  
 kassa) {  
 if(p.getName().equals(line)) {  
 System.*out*.println("Квиток №" + (i + 1) + " :");  
 p.print();  
 System.*out*.println();  
 i++;  
 }  
 }

if(i == 0){

System.*out*.println("Такого квитка немає !");  
 }  
}

public void sort() throws IOException {  
 count\_note();  
 ArrayList<Kassa> sortkassa = new ArrayList<Kassa>(kassa);  
 sortkassa.sort(Comparator.*comparing*(Kassa::getHours));  
  
 int i = 0;  
 System.*out*.println("\nВідсортований список :");  
 for (var p:  
 sortkassa) {  
 System.*out*.println("\nКвиток №" + (i + 1) + " :");  
 p.print();  
 i++;  
 }  
}

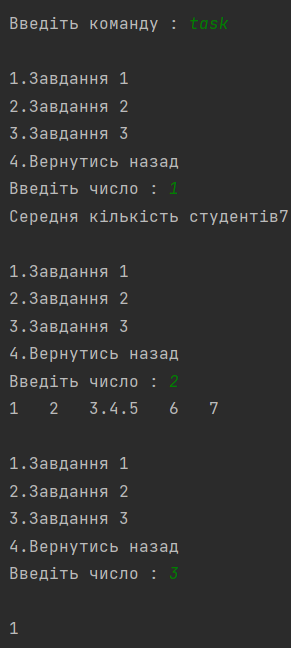
public void delete() throws IOException {  
 count\_note();  
  
 Scanner scanner = new Scanner(System.*in*);  
 System.*out*.print("\nВведіть порядковий номер квитка, який ви хочете видалити : ");  
 String line = scanner.nextLine();  
 if(line.isEmpty() || Integer.*parseInt*(line) <= 0 || Integer.*parseInt*(line) > kassa.size()){  
 System.*err*.println("Не вірно введений номер!");  
 this.delete();  
 return;  
 }  
 int nomer = Integer.*parseInt*(line);  
 nomer--;  
  
 kassa.remove(nomer);  
 System.*out*.println("\nВидалення пройшло успішно!");  
  
 this.write\_to\_file("Kassa.txt");  
}

**Результат виконання :**

1. **Код :**
2. private void task\_1() {  
    try {  
    int count\_student = 0;  
    int count = 0;  
    for (Lesson lesson : lesson) {  
    count\_student += lesson.kilkist;  
    count++;  
    }  
    if (count == 0)  
    throw new IOException("Студентів немає!");  
     
    double result = (double) count\_student / count;  
     
    System.*out*.println("Середня кількість студентів"+ result);  
    } catch (IOException ex) {  
    System.*out*.println(ex.getMessage());  
    }  
   }

private void task\_2() {  
 int temp = 0;  
 Lesson[] arr = sort\_date();  
 for (int i = 1; i < arr.length; i++)  
 temp = arr[i].kilkist;  
  
 for (Lesson lesson : lesson) {  
 if (lesson.kilkist > temp){  
 temp = lesson.kilkist;  
 }  
 }  
 for (Lesson lesson : lesson) {  
 if(lesson.kilkist>=temp){  
 System.*out*.println(lesson);  
 }  
 }  
}  
  
private void task\_3() {  
 System.*out*.println();  
 int k = 0;  
 for (Lesson p : lesson) {  
 String[] words = p.Ncathedra.split("[\\s]+");  
 System.*out*.println(words.length);  
 }  
}

**Результат :**



**Висновок:** Вивчив особливості роботи з класами та їх спадкуванням у Java.