**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
НАЦІОНАЛЬНОМУ УНІВЕРСИТЕТІ “ЛЬВІВСЬКА  
ПОЛІТЕХНІКА”**

**Кафедра систем штучного інтелекту**

**Лабораторна робота №1  
з дисципліни  
«Об’єктно-орієнтоване програмування»**

**Виконав:  
студент групи КН-109**

**Качмар Олексій  
Викладач:  
Мочурад Л.І.**

**Львів – 2018 р.**

Лабораторна робота №3. Об’єктно-орієнтована декомпозиція. Основи введення/виведення Java SE.

Мета

Використання об’єктно-орієнтованого підходу для розробки об’єкта предметної (прикладної) галузі. Оволодіння навичками управління введенням/виведенням даних з використанням класів Java SE.

Вимоги 1. Використовуючи об’єктно-орієнтований аналіз, реалізувати класи для представлення сутностей відповідно списку прикладних задач - domain-об’єктів (Прикладні задачі. Список №2. 20 варіантів) 2. Забезпечити та продемонструвати коректне введення та відображення кирилиці. 3. Продемонструвати можливість управління масивом domain-об’єктів. 4. Забезпечити можливість збереження і відновлення масива об’єктів рішення завдання з Прикладні задачі. Список №2. 20 варіантів. 5. Забороняється використання ​ стандартного протокола серіалізації​ . 6. Продемонструвати використання моделі ​ Long Term Persistence​ . 7. Забезпечити діалог з користувачем у вигляді текстового меню. 8. При збереженні та відновленні даних забезпечити діалоговий режим вибору директорії з відображенням вмісту і можливістю переміщення по підкаталогах.

Прикладна задача:

Бібліотека. Дані про книгу: ​ ISBN​ ; назва; автори (кількість не обмежена); видавництво; жанр; дата видання.

Код програми:

Клас -сутність:

**import** java.io.Serializable;

**import** java.util.ArrayList;

**public** **class** Book **implements** Serializable

{

**private** ArrayList<String> authors;

**public** ArrayList<String> getAuthors()

{ **return** authors; }

**public** **void** setAuthors(ArrayList<String> Authors)

{ authors = (ArrayList<String>)Authors.clone(); }

**private** String widawniztwo;

**public** String getWidawniztwo()

{ **return** widawniztwo; }

**public** **void** setWidawniztwo(String Widawniztwo)

{ widawniztwo = Widawniztwo; }

**private** String isbn;

**public** String getIsbn()

{ **return** isbn; }

**public** **void** setIsbn(String Isbn)

{ isbn = Isbn; }

**private** String name;

**public** String getName()

{ **return** name; }

**public** **void** setName(String Name)

{name = Name;}

**private** String genre;

**public** String getGenre()

{ **return** genre; }

**public** **void** setGenre(String Genre)

{ genre = Genre; }

**private** String date;

**public** String getDate()

{ **return** date; }

**public** **void** setDate(String Date)

{ date = Date; }

}

Робота з класом сутністю:

import java.beans.XMLDecoder;

import java.beans.XMLEncoder;

import java.io.BufferedInputStream;

import java.io.BufferedOutputStream;

import java.io.File;

import java.io.FileInputStream;

import java.io.FileOutputStream;

import java.io.IOException;

import java.util.ArrayList;

import java.util.List;

import java.util.Locale;

import java.util.Scanner;

//Locale.setDefault(new Locale(language: "uk",country: " UA");

public class MainClass

{

public static String removeCharAt(String s, int pos)

{

return s.substring(0, pos) + s.substring(pos + 1);

}

public static void write(Book[] a, String path) throws Exception

{

XMLEncoder encoder = new XMLEncoder(new BufferedOutputStream(new FileOutputStream(path)));

encoder.writeObject(a.length);

for(int i = 0;i<a.length;i++)

{

encoder.writeObject(a[i]);

}

encoder.close();

}

public static Book[] read(String filename) throws Exception

{

XMLDecoder decoder = new XMLDecoder(new BufferedInputStream(new FileInputStream(filename)));

int size = (int)decoder.readObject();

Book[] books = new Book[size];

for(int i = 0; i < size; i++)

{

books[i] = (Book)decoder.readObject();

}

decoder.close();

return books;

}

@SuppressWarnings("unlikely-arg-type")

public static void main(String[] args) throws Exception

{

Scanner ob = new Scanner(System.in);

int numberOfBooks = ob.nextInt();

Book[] arr = new Book[numberOfBooks];

Locale.setDefault(new Locale("uk", "UA"));

while(true)

{

System.out.println("\n1 - Заповніть масив для домейн-об'єкту");

System.out.println("2 - Вкажіть шлях до файлу,та запишіть дані в файл");

System.out.println("3 - зчитування з xml-файлу");

System.out.println("4 - вихід");

int num = ob.nextInt();

switch(num)

{

case 1:

{

String name,widawniztwo,genre,date,ISBN;

String authors;

ArrayList<String> listOfAuthors = new ArrayList();

for(int i = 0;i<numberOfBooks;i++)

{

ob.nextLine();

System.out.println("Enter book's ISBN:\n");

ISBN = ob.nextLine();

System.out.println("Enter name:\n");

name = ob.nextLine();

System.out.println("Enter widawniztwo:\n");

widawniztwo = ob.nextLine();

System.out.println("Enter genre of Book:\n");

genre = ob.nextLine();

System.out.println("Enter date of widannja:\n");

date = ob.nextLine();

System.out.println("Enter authors,that wrote thee book:\n");

authors = ob.nextLine();

for(String author:authors.split(" "))

{

listOfAuthors.add(author);

}

arr[i] = new Book();

arr[i].setAuthors(listOfAuthors);

arr[i].setDate(date);

arr[i].setGenre(genre);

arr[i].setIsbn(ISBN);

arr[i].setName(name);

arr[i].setWidawniztwo(widawniztwo);

listOfAuthors.clear();

}

/\* MainClass.write(arr, "file.xml");

MainClass.read("file.xml");\*/

break;

}

case 2:

{

boolean result = false;

ob.nextLine();

String index = null;

String fileName = ob.nextLine();

boolean fileFound = false;

String path = "D:\\Універ";

String next;

File dir = new File(path);

while(!dir.getName().equals("kachmar-olexij3"))

{

System.out.println("Виберіть директорію");

for(File item : dir.listFiles())

{

if(item.isDirectory())

{

System.out.println(item.getName() + " folder\t");

}

else

{

System.out.println(item.getName() + " file\t");

}

}

next = '\\' + ob.nextLine();

if(!next.equals("\\"))

{

path+=next;

}

do

{

dir = new File(path);

for(File item : dir.listFiles())

{

if(item.isDirectory())

{

System.out.println(item.getName() + " folder\t");

}

else

{

System.out.println(item.getName() + " file\t");

}

}

System.out.println("Продовжити?");

index = ob.nextLine();

if(index.equals("Записати"))

{

MainClass.write(arr, path+"\\"+fileName);

break;

}

System.out.println("Якщо хочете повернутись введіть : back.Інакше введіть назву директорії");

index = ob.nextLine();

if(index.equals("back"))

{

do

{

path = removeCharAt(path,path.length() - 1);

} while(path.charAt(path.length() - 1) != '\\');

}

else

{

path+="\\";

path+=index;

}

}while(true);

}

break;

}

case 3:

{

Book[] new\_b = MainClass.read("file.xml").clone();

for(int i = 0;i<new\_b.length;i++)

{

System.out.println("Унікальний номер книги:\n");

System.out.println(new\_b[i].getIsbn());

System.out.println("Назва книги:\n");

System.out.println(new\_b[i].getName());

System.out.println("Дата видання книги:\n");

System.out.println(new\_b[i].getDate());

System.out.println("Жанр книги:\n");

System.out.println(new\_b[i].getGenre());

System.out.println("Видавництво:\n");

System.out.println(new\_b[i].getWidawniztwo());

System.out.println("Автори:\n");

ArrayList<String> str = (ArrayList<String>) new\_b[i].getAuthors().clone();

for(String s : str)

{

System.out.println(s);

}

}

break;

}

case 4:

{

System.out.println("Bye!");

System.exit(0);

}

default:

{

System.out.println("Unknown command");

}

}

}

}

}

Висновок: Я використав об’єктно-орієнтований підхід для розробки об’єкта предметної (прикладної) галузі. Оволодів навичками управління введенням/виведенням даних з використанням класів Java SE.