Міністерство освіти інауки України

Національний технічний університет України

“Київський політехнічний інститут”

Кафедра АСОІУ

ЗВІТ

про виконання комп’ютерного практикуму № 5

з дисципліни

“Основи програмної інженерії”

Тема: Наслідування та поліморфізм.

Виконав:

студент групи ІП-51

Субцельний Олександр Володимирович

«5»квітня 2016 р.

Перевірив:

Радченко Костянтин Олександрович

«12»квітня 2016 р.

**Мета**: Ознайомлення з механізмом наслідування та принципом поліморфізму.

Використання механізму наслідування та принципу поліморфізму в мові

Java. Здобуття навичок у використанні механізму наслідування та принципу

поліморфізму.

**Завдання**

1. Визначити C 17 як остачу від ділення номера залікової книжки студента на 17.

2. Визначити ієрархію музичних композицій. Записати на диск альбом.

Порахувати тривалість альбому. Провести перестановку композицій диска на

основі приналежності до стилю. Знайти композицію, що відповідає заданому

діапазону довжини треків.

3.Створити узагальнений клас та не менше 3 класів-нащадків, що описують задану

варіантом (п.2) область знань. Створити клас, що складається з масиву об’єктів, з

яким можна виконати вказані варіантом дії. Необхідно обробити всі виключні

ситуації, що можуть виникнути під час виконання програмного коду. Всі змінні

повинні бути описані та значення їх задані у виконавчому методі. Код повинен

відповідати стандартам JCC та бути детально задокументований.

**Код програми**

import java.util.Scanner;

public class main {

public static void main(String[] arg) {

boolean l,p,d;

Album create = new Album();

System.out.println("do you wish to calc length of comps[false/true]");

Scanner sc = new Scanner(System.in);

l = sc.nextBoolean();

System.out.println("do you wish to sort comps by style[false/true]");

Scanner sc1 = new Scanner(System.in);

p = sc1.nextBoolean();

System.out.println("do you wish to find comps by length[false/true]");

Scanner sc2 = new Scanner(System.in);

d = sc2.nextBoolean();

create.start(l,p,d); }}

import java.util.Scanner;

public class Album {

public static int n = 7;

Composition obj = new Composition();

public void start(boolean l,boolean p,boolean d){

Rock[] r = new Rock[5];

for(int i=0;i<2;i++)

r[i]=new Rock();

r[0].rocksong("Liebe",4.08,"rock");

r[1].rocksong("Lust",3.24,"rock") ;

Jazz[] j = new Jazz[5];

for(int i=0;i<1;i++)

j[i]=new Jazz();

j[0].jazzsong("BlueSuedeShoes",5.26,"jazz");

Narodniy[] nar = new Narodniy[5];

for(int i=0;i<2;i++)

nar[i]=new Narodniy();

nar[0].narodniysong("narodnaya",3.42,"folk");

nar[1].narodniysong("RusskieBabki",3.42,"folk");

Composition[] album = new Composition[10];

album[0]=r[0]; album[1]=r[1];

album[2]=j[0]; album[3]=nar[0];

System.out.println("add songs?[true/false]");

Scanner sc = new Scanner(System.in);

boolean y = sc.nextBoolean();

if (y) { System.out.println("Enter n");

int n = sc.nextInt();

Scanner sc1 = new Scanner(System.in);

Scanner sc2 = new Scanner(System.in);

Scanner sc3 = new Scanner(System.in);

String temp;

for (int i = 0; i < n; i++) {

int rindex=2; int jindex=1; int narindex = 2;

System.out.println("Enter style of comp " + (i + 1));

temp = sc2.nextLine();

if (temp.equals("rock")) {

r[rindex] = new Rock();

System.out.println("Enter title of comp " + (i + 1));

r[rindex].title = sc3.nextLine();

System.out.println("Enter length of comp " + (i + 1));

r[rindex].length = sc1.nextDouble();

r[rindex].style = temp;

album[5+i] = r[rindex];

rindex++; }

if (temp.equals("jazz")) {

j[jindex] = new Jazz();

System.out.println("Enter title of comp " + (i + 1));

j[jindex].title = sc3.nextLine();

System.out.println("Enter length of comp " + (i + 1));

j[jindex].length = sc1.nextDouble();

j[jindex].style = temp;

album[5+i] = j[jindex];

jindex++; }

if (temp.equals("folk")) {

nar[narindex] = new Narodniy();

System.out.println("Enter title of comp " + (i + 1));

nar[narindex].title = sc3.nextLine();

System.out.println("Enter length of comp " + (i + 1));

nar[narindex].length = sc1.nextDouble();

nar[narindex].style = temp;

album[5+i] = nar[narindex];

narindex++; } }}

if (l) { obj.whatLength(album); }

if (p){ obj.sortByStyle(album); }

if (d){ obj.findSongs(album); } }

public void album() {

Scanner sc = new Scanner(System.in);

Scanner sc1 = new Scanner(System.in);

Scanner sc2 = new Scanner(System.in);

for (int i = 0; i < n; i++) {

System.out.println("Enter title of comp "+ (i+1));

album[i].title = sc.nextLine();

System.out.println("Enter length of comp "+ (i+1));

album[i].length = sc1.nextInt();

System.out.println("Enter style of comp " + (i+1));

album[i].style = sc2.nextLine(); } }}

import java.util.Scanner;

public class Composition {

public String title;

public double length;

public String style;

public void settitle(String title){

this.title = title; }

public void setlength(double length){

this.length = length; }

public double whatLength(Composition[] m){

double len = 0;

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

try { len = len + m[i].length;

} catch (Exception e) { break; } }

System.out.println("The total length of compositions : " + len);

System.out.println();

return len; }

public void findSongs(Composition[] m){

System.out.println("enter the min length");

Scanner sc1 = new Scanner(System.in);

Scanner sc2 = new Scanner(System.in);

int len1 = sc1.nextInt();

System.out.println("enter the max length");

int len2 = sc2.nextInt();

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

try {if ((m[i].length >= len1) && (m[i].length <= len2)) {

System.out.println(m[i].title); } }

catch (Exception e){ break; } }

System.out.println(); }

public void sortByStyle(Composition[] m){

System.out.println("Rock compositions: ");

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

try { if (m[i].style.equals("rock")) {

System.out.println("---" + m[i].title); } }

catch (Exception e){ break; } }

System.out.println("Jazz compositions: ");

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

try {if (m[i].style.equals("jazz")) {

System.out.println("---" + m[i].title); } }

catch(Exception e){ break; } }

System.out.println("Folk compositions: ");

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

try { if (m[i].style.equals("folk")) {

System.out.println("---" + m[i].title); } }

catch (Exception e){ break; } }

System.out.println(); }}

public class Jazz extends Composition{

public void setstyle(String style){

this.style = style; }

public void jazzsong(String title,double length,String style){

settitle(title);

setlength(length);

setstyle(style); }}

public class Rock extends Composition{

public void setstyle(String style){

this.style = style; }

public void rocksong(String title,double length,String style){

settitle(title);

setlength(length);

setstyle(style); }}

public class Narodniy extends Composition {

public void setstyle(String style){

this.style = style; }

public void narodniysong(String title,double length,String style){

settitle(title);

setlength(length);

setstyle(style); }}

**Висновок**

Ознайомився з механізмом наслідування та принципом поліморфізму.

Використав механізм наслідування та принцип поліморфізму в мові Java. Здобув навички у використанні механізму наслідування та принципуполіморфізму. В ході лабораторної роботи була написана програма, яка визначає їєрархію музичних композицій та записує на диск альбом. Створено методи для рахування тривалості альбому, проводить перестановку композицій диска на основі приналежності до стилю, знаходить композицію, що відповідає заданому діапазону довжини треків.