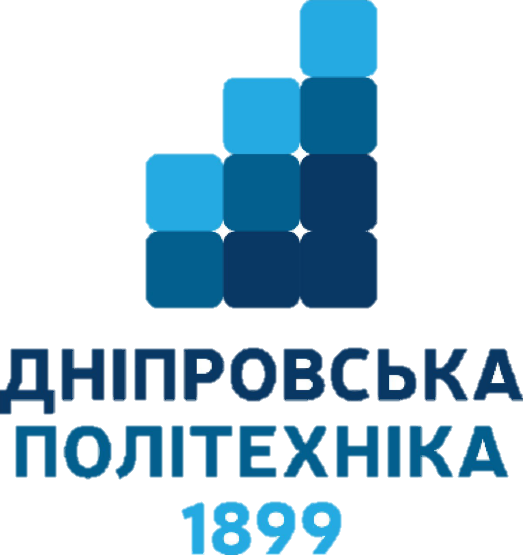
Міністерство освіти і науки України

Національний технічний університет

«Дніпровська політехніка»



Факультет інформаційних технологій

Кафедра системного аналізу і управління

*Звіт*

З дисципліни:

“ОПСJ”

**Виконав**

студент групи 124-18ск-1

Владимиров Я.Д

**GitHub**

https://github.com/gotoindex/javalabs.git

**Перевірив**

Мінєєв О.С

м. Дніпро

2020

Лабораторна робота №3

Хід роботи

Постановка задачі:

Створити програму що буде створювати та обробляти комплексний об'єкт під назвою університет(university). Програма повинна складатися з трьох частин: модель вид та контролер згідно з парадигмою mvc (Model View Controller).  Кожній з цих груп повинна відповідати package з  відповідною назвою.  В моделі повинні знаходитись усі класи що відповідають за структурні підрозділи університету. Серед них:   університет, факультет, кафедра, група, студент, людина (Human). Усі вони повинні містити  назву  типу string та голову типу Human. Студент також повинен бути породжений від  Human.  Human повинен мати поля ім'я, прізвище, побатькові та стать.  Усі поля повинні бути строковими окрім поля стать. Стать повинна використовувати спеціальний  enum  типу Sex(стать).

В цій лабораторній роботі  група View Нам не потрібна.

 Що стосується групи контроллер (controller)  то вона повинна містити менеджери що дозволяють нам створити відповідні підрозділи наприклад  StudentCreator, FacultyCreator, GroupCreator  та інші, кожен з яких повинен використовувати можливості нижчого за рівнем   створювача. Програма повинна також містити клас Run, в якому буде знаходитись точка входу та методи, що повинні дати можливість створити університет.   Процес створення університету повинен бути зроблений в методі createTypycalUniversity.

В програмі активно рекомендується використовувати абстрактні класи та інтерфейси.

Лістинг програми:

*models/Sex.java:*

**package** lr3.models;  
  
*/\*  
 \* This small enum class is used instead of a regular boolean value.  
 \* The reason for this choice is purely for the readability of the code.  
 \*/***public enum** Sex {  
 ***MALE***,  
 ***FEMALE***}

*models/CommonInterface.java:*

**package** lr3.models;  
  
*/\*\*  
 \* The purpose of this interface is to create the ability to use generic types  
 \* with classes like {****@code*** *lr3.models.AbstractGroupModel} and  
 \* {****@code*** *lr3.models.HumanModel}.  
 \* The interface has only one method definition - {****@code*** *show()}.  
 \*  
 \* <p>Please make sure you are using the correct version of Idea,  
 \* if you're not sure which version you are using please contact  
 \* the administrator. The version intended for usage is maven-14.  
 \*  
 \* <p>The documentation for the methods contained in this class includes  
 \* brief descriptions of the <i>implementations</i>. Such descriptions should  
 \* be regarded as <i>implementation notes</i>, rather than parts of the  
 \* <i>specification</i>. Implementors should feel free to substitute other  
 \* algorithms, so long as the specification itself is adhered to. (For  
 \* example, the algorithm used by {****@code*** *sort(Object[])} does not have to be  
 \* a MergeSort, but it does have to be <i>stable</i>.)  
 \*  
 \* <p>This interface is a member of the  
 \* <a href="https://github.com/gotoindex/javalabs">  
 \* Java Learning Course</a>.  
 \*  
 \** ***@author*** *Akim Vladimirov  
 \*/***public interface** CommonInterface {  
  
 **void** show(String prefix);  
  
}

*models/HumanModel.java:*

**package** lr3.models;  
  
**import** java.util.Objects;  
  
*/\*\*  
 \* This class is a model that contains information about a human.  
 \* It's purpose is to be both a base for the {****@code*** *lr3.models.StudentModel}  
 \* class and to act as a head person for any group-like class in the package.  
 \*  
 \* <p>Please make sure you are using the correct version of Idea,  
 \* if you're not sure which version you are using please contact  
 \* the administrator. The version intended for usage is maven-14.  
 \*  
 \* <p>The documentation for the methods contained in this class includes  
 \* brief descriptions of the <i>implementations</i>. Such descriptions should  
 \* be regarded as <i>implementation notes</i>, rather than parts of the  
 \* <i>specification</i>. Implementors should feel free to substitute other  
 \* algorithms, so long as the specification itself is adhered to. (For  
 \* example, the algorithm used by {****@code*** *sort(Object[])} does not have to be  
 \* a MergeSort, but it does have to be <i>stable</i>.)  
 \*  
 \* <p>This class is a member of the  
 \* <a href="https://github.com/gotoindex/javalabs">  
 \* Java Learning Course</a>.  
 \*  
 \** ***@author*** *Akim Vladimirov  
 \*/***public class** HumanModel **implements** CommonInterface {  
  
 **public** String **firstName**;  
 **public** String **lastName**;  
 **public** Sex **sex**;  
  
 */\*\*  
 \* This main constructor is not used directly in the code but is necessary for  
 \* the structure of the package.  
 \*  
 \** ***@implNote*** *Since no parameters are entered, all this constructor does is calling  
 \* a parametrized constructor, giving it pre-set parameters.  
 \*/* **public** HumanModel() {  
 **this**(**"John"**, **"Smith"**, Sex.***MALE***);  
 }  
  
 */\*\*  
 \* This is the main parametrized constructor (in this class) necessary for the task  
 \* completion and thus the only parametrized constructor in the class.  
 \*/* **public** HumanModel(String firstName, String lastName, Sex sex) {  
 **this**.**firstName** = firstName;  
 **this**.**lastName** = lastName;  
 **this**.**sex** = sex;  
 }  
  
 */\*\*  
 \* The only purpose of this method is readability and shortening the length  
 \* of the code in other parts of the program.  
 \*  
 \* All it does is return all credentials of the person in question.  
 \*/* **public** String getFullName() {  
 **return this**.**firstName** + **" "** + **this**.**lastName** + **" ("** + **this**.**sex** + **")"**;  
 }  
  
 */\*\*  
 \* This method specifies the one in the interface, implemented by this class.  
 \*  
 \* All it does is display all credentials of the person in question.  
 \*/* @Override  
 **public void** show(String prefix) {  
 System.***out***.println(prefix + **"- "** + **this**.getFullName());  
 }  
  
 */\*  
 \* The following methods are overrides for the lr4 package.  
 \*/* @Override  
 **public** String toString() {  
 **return "{first\_name="** + **firstName** + **", last\_name="** + **lastName** + **", sex="** + **sex** + **"}"**;  
 }  
  
 @Override  
 **public boolean** equals(Object obj) {  
 **if** (**this** == obj) **return true**;  
 **if** (!(obj **instanceof** HumanModel)) **return false**;  
 HumanModel human = (HumanModel) obj;  
 **return** Objects.*equals*(**firstName**, human.**firstName**) &&  
 Objects.*equals*(**lastName**, human.**lastName**) &&  
 **sex** == human.**sex**;  
 }  
  
 @Override  
 **public int** hashCode() {  
 **return** Objects.*hash*(**firstName**, **lastName**, **sex**);  
 }  
  
}

*models/StudentModel.java:*

**package** lr3.models;  
  
*/\*\*  
 \* This class is a model that contains information about a student.  
 \* For obvious reasons, it is based off of the {****@code*** *lr3.models.HumanModel}  
 \* class and and has no additional functionality, as it is not required by the task.  
 \*  
 \* <p>Please make sure you are using the correct version of Idea,  
 \* if you're not sure which version you are using please contact  
 \* the administrator. The version intended for usage is maven-14.  
 \*  
 \* <p>The documentation for the methods contained in this class includes  
 \* brief descriptions of the <i>implementations</i>. Such descriptions should  
 \* be regarded as <i>implementation notes</i>, rather than parts of the  
 \* <i>specification</i>. Implementors should feel free to substitute other  
 \* algorithms, so long as the specification itself is adhered to. (For  
 \* example, the algorithm used by {****@code*** *sort(Object[])} does not have to be  
 \* a MergeSort, but it does have to be <i>stable</i>.)  
 \*  
 \* <p>This class is a member of the  
 \* <a href="https://github.com/gotoindex/javalabs">  
 \* Java Learning Course</a>.  
 \*  
 \** ***@author*** *Akim Vladimirov  
 \*/***public class** StudentModel **extends** HumanModel {  
}

*models/AbstractGroupModel.java:*

**package** lr3.models;  
  
**import** lr3.controllers.HumanCreator;  
  
**import** java.util.ArrayList;  
**import** java.util.Objects;  
  
*/\*\*  
 \* This class is a base model that is used to define common properties of:  
 \* a university, a faculty, a chair, and a group.  
 \*  
 \* <p>Please make sure you are using the correct version of Idea,  
 \* if you're not sure which version you are using please contact  
 \* the administrator. The version intended for usage is maven-14.  
 \*  
 \* <p>The documentation for the methods contained in this class includes  
 \* brief descriptions of the <i>implementations</i>. Such descriptions should  
 \* be regarded as <i>implementation notes</i>, rather than parts of the  
 \* <i>specification</i>. Implementors should feel free to substitute other  
 \* algorithms, so long as the specification itself is adhered to. (For  
 \* example, the algorithm used by {****@code*** *sort(Object[])} does not have to be  
 \* a MergeSort, but it does have to be <i>stable</i>.)  
 \*  
 \* <p>This class is a member of the  
 \* <a href="https://github.com/gotoindex/javalabs">  
 \* Java Learning Course</a>.  
 \*  
 \** ***@author*** *Akim Vladimirov  
 \*/***public abstract class** AbstractGroupModel<T **extends** CommonInterface> **implements** CommonInterface {  
  
 */\*  
 \* This is what each structure in the university (including the university  
 \* itself) have in common: a name (like the University of Technology),  
 \* a head (a leader or a responsible person in command), a list of  
 \* sub-elements (for a university its faculties, for group its students).  
 \*/* **public** String **name**;  
 **public** HumanModel **head**;  
 **public** ArrayList<T> **list**;  
  
 */\*  
 \* This parameter is only used in the show() method and is used to properly  
 \* display a plural name of sub-elements of the element (like  
 \* "students" in a group model).  
 \*/* **public** String **listPlural**;  
  
 */\*\*  
 \* This main constructor is not used directly in the code but is necessary for  
 \* the structure of the package.  
 \*  
 \** ***@implNote*** *Since no parameters are entered, all this constructor does is calling  
 \* a parametrized constructor, giving it pre-set parameters.  
 \*/* **public** AbstractGroupModel() {  
 **this**(**"Unknown Group"**, **new** HumanCreator(1).**object**, **new** ArrayList<>());  
 }  
  
 */\*\*  
 \* This is the main parametrized constructor (in this class) necessary for the task  
 \* completion and thus the only parametrized constructor in the class.  
 \*/* **public** AbstractGroupModel(String name, HumanModel head, ArrayList<T> list) {  
 **this**.**name** = name;  
 **this**.**head** = head;  
 **this**.**list** = list;  
 }  
  
 */\*\*  
 \* This method specifies the one in the interface, implemented by this class.  
 \*  
 \* All it does is display all information about a group and then lists all of  
 \* its sub-elements.  
 \*/* **public void** show(String prefix) {  
 System.***out***.println(prefix + **"- "** + **this**.**name** + **":"**);  
 System.***out***.println(prefix + **" head: "** + **this**.**head**.getFullName());  
 System.***out***.println(prefix + **" list of "** + **this**.**listPlural** + **":"**);  
 **for** (T t : **this**.**list**) t.show(prefix + **"\t"**);  
 }  
  
 */\*  
 \* The following methods are overrides for the lr4 package.  
 \*/* @Override  
 **public** String toString() {  
 **return "{head="** + **head** + **", name="** + **name** + **", list="** + **list** + **", list\_plural="** + **listPlural** + **"}"**;  
 }  
  
 @Override  
 **public boolean** equals(Object obj) {  
 **if** (**this** == obj) **return true**;  
 **if** (!(obj **instanceof** AbstractGroupModel)) **return false**;  
 AbstractGroupModel<?> other = (AbstractGroupModel<?>) obj;  
 **return** Objects.*equals*(**head**, other.**head**) &&  
 Objects.*equals*(**name**, other.**name**) &&  
 Objects.*equals*(**list**, other.**list**) &&  
 Objects.*equals*(**listPlural**, other.**listPlural**);  
 }  
  
 @Override  
 **public int** hashCode() {  
 **return** Objects.*hash*(**head**, **name**, **list**, **listPlural**);  
 }  
}

*models/GroupModel.java:*

**package** lr3.models;  
  
*/\*\*  
 \* This class is a model that contains information about a typical group of students.  
 \* It is based off of the {****@code*** *lr3.models.AbstractGroupModel}  
 \* class and and has no additional functionality, as it is not required by the task.  
 \*  
 \* <p>Please make sure you are using the correct version of Idea,  
 \* if you're not sure which version you are using please contact  
 \* the administrator. The version intended for usage is maven-14.  
 \*  
 \* <p>The documentation for the methods contained in this class includes  
 \* brief descriptions of the <i>implementations</i>. Such descriptions should  
 \* be regarded as <i>implementation notes</i>, rather than parts of the  
 \* <i>specification</i>. Implementors should feel free to substitute other  
 \* algorithms, so long as the specification itself is adhered to. (For  
 \* example, the algorithm used by {****@code*** *sort(Object[])} does not have to be  
 \* a MergeSort, but it does have to be <i>stable</i>.)  
 \*  
 \* <p>This class is a member of the  
 \* <a href="https://github.com/gotoindex/javalabs">  
 \* Java Learning Course</a>.  
 \*  
 \** ***@author*** *Akim Vladimirov  
 \*/***public class** GroupModel **extends** AbstractGroupModel<StudentModel> {  
}

*models/ChairModel.java:*

**package** lr3.models;  
  
*/\*\*  
 \* This class is a model that contains information about a typical chair in the university.  
 \* It is based off of the {****@code*** *lr3.models.AbstractGroupModel}  
 \* class and and has no additional functionality, as it is not required by the task.  
 \*  
 \* <p>Please make sure you are using the correct version of Idea,  
 \* if you're not sure which version you are using please contact  
 \* the administrator. The version intended for usage is maven-14.  
 \*  
 \* <p>The documentation for the methods contained in this class includes  
 \* brief descriptions of the <i>implementations</i>. Such descriptions should  
 \* be regarded as <i>implementation notes</i>, rather than parts of the  
 \* <i>specification</i>. Implementors should feel free to substitute other  
 \* algorithms, so long as the specification itself is adhered to. (For  
 \* example, the algorithm used by {****@code*** *sort(Object[])} does not have to be  
 \* a MergeSort, but it does have to be <i>stable</i>.)  
 \*  
 \* <p>This class is a member of the  
 \* <a href="https://github.com/gotoindex/javalabs">  
 \* Java Learning Course</a>.  
 \*  
 \** ***@author*** *Akim Vladimirov  
 \*/***public class** ChairModel **extends** AbstractGroupModel<GroupModel> {  
}

*models/FacultyModel.java:*

**package** lr3.models;  
  
*/\*\*  
 \* This class is a model that contains information about a typical faculty.  
 \* It is based off of the {****@code*** *lr3.models.AbstractGroupModel}  
 \* class and and has no additional functionality, as it is not required by the task.  
 \*  
 \* <p>Please make sure you are using the correct version of Idea,  
 \* if you're not sure which version you are using please contact  
 \* the administrator. The version intended for usage is maven-14.  
 \*  
 \* <p>The documentation for the methods contained in this class includes  
 \* brief descriptions of the <i>implementations</i>. Such descriptions should  
 \* be regarded as <i>implementation notes</i>, rather than parts of the  
 \* <i>specification</i>. Implementors should feel free to substitute other  
 \* algorithms, so long as the specification itself is adhered to. (For  
 \* example, the algorithm used by {****@code*** *sort(Object[])} does not have to be  
 \* a MergeSort, but it does have to be <i>stable</i>.)  
 \*  
 \* <p>This class is a member of the  
 \* <a href="https://github.com/gotoindex/javalabs">  
 \* Java Learning Course</a>.  
 \*  
 \** ***@author*** *Akim Vladimirov  
 \*/***public class** FacultyModel **extends** AbstractGroupModel<ChairModel> {  
}

*models/UniversityModel.java:*

**package** lr3.models;  
  
*/\*\*  
 \* This class is a model that contains information about a typical university.  
 \* It is based off of the {****@code*** *lr3.models.AbstractGroupModel}  
 \* class and and has no additional functionality, as it is not required by the task.  
 \*  
 \* <p>Please make sure you are using the correct version of Idea,  
 \* if you're not sure which version you are using please contact  
 \* the administrator. The version intended for usage is maven-14.  
 \*  
 \* <p>The documentation for the methods contained in this class includes  
 \* brief descriptions of the <i>implementations</i>. Such descriptions should  
 \* be regarded as <i>implementation notes</i>, rather than parts of the  
 \* <i>specification</i>. Implementors should feel free to substitute other  
 \* algorithms, so long as the specification itself is adhered to. (For  
 \* example, the algorithm used by {****@code*** *sort(Object[])} does not have to be  
 \* a MergeSort, but it does have to be <i>stable</i>.)  
 \*  
 \* <p>This class is a member of the  
 \* <a href="https://github.com/gotoindex/javalabs">  
 \* Java Learning Course</a>.  
 \*  
 \** ***@author*** *Akim Vladimirov  
 \*/***public class** UniversityModel **extends** AbstractGroupModel<FacultyModel> {  
}

*controllers/HumanCreator.java:*

**package** lr3.controllers;  
  
**import** lr3.models.HumanModel;  
**import** lr3.models.Sex;  
**import** lr3.models.StudentModel;  
  
*/\*\*  
 \* This class is a controller for the human model.  
 \*  
 \* <p>Please make sure you are using the correct version of Idea,  
 \* if you're not sure which version you are using please contact  
 \* the administrator. The version intended for usage is maven-14.  
 \*  
 \* <p>The documentation for the methods contained in this class includes  
 \* brief descriptions of the <i>implementations</i>. Such descriptions should  
 \* be regarded as <i>implementation notes</i>, rather than parts of the  
 \* <i>specification</i>. Implementors should feel free to substitute other  
 \* algorithms, so long as the specification itself is adhered to. (For  
 \* example, the algorithm used by {****@code*** *sort(Object[])} does not have to be  
 \* a MergeSort, but it does have to be <i>stable</i>.)  
 \*  
 \* <p>This class is a member of the  
 \* <a href="https://github.com/gotoindex/javalabs">  
 \* Java Learning Course</a>.  
 \*  
 \** ***@author*** *Akim Vladimirov  
 \*/***public class** HumanCreator {  
  
 **public** HumanModel **object**;  
  
 **public** HumanCreator(**int** index) {  
  
 StudentModel student = **new** StudentModel();  
 student.**firstName** += index;  
  
 *//In this case it has been decided to give female sex to every odd student.* student.**sex** = index % 2 == 0 ? Sex.***MALE*** : Sex.***FEMALE***;  
 **this**.**object** = student;  
  
 }  
  
}

*controllers/StudentCreator.java:*

**package** lr3.controllers;  
  
**import** lr3.models.Sex;  
**import** lr3.models.StudentModel;  
  
*/\*\*  
 \* This class is a controller for the student model.  
 \*  
 \* <p>Please make sure you are using the correct version of Idea,  
 \* if you're not sure which version you are using please contact  
 \* the administrator. The version intended for usage is maven-14.  
 \*  
 \* <p>The documentation for the methods contained in this class includes  
 \* brief descriptions of the <i>implementations</i>. Such descriptions should  
 \* be regarded as <i>implementation notes</i>, rather than parts of the  
 \* <i>specification</i>. Implementors should feel free to substitute other  
 \* algorithms, so long as the specification itself is adhered to. (For  
 \* example, the algorithm used by {****@code*** *sort(Object[])} does not have to be  
 \* a MergeSort, but it does have to be <i>stable</i>.)  
 \*  
 \* <p>This class is a member of the  
 \* <a href="https://github.com/gotoindex/javalabs">  
 \* Java Learning Course</a>.  
 \*  
 \** ***@author*** *Akim Vladimirov  
 \*/***public class** StudentCreator {  
  
 **public** StudentModel **object**;  
  
 **public** StudentCreator(**int** index) {  
  
 StudentModel student = **new** StudentModel();  
 student.**firstName** += index;  
  
 *//In this case it has been decided to give female sex to every odd student.* student.**sex** = index % 2 == 0 ? Sex.***MALE*** : Sex.***FEMALE***;  
 **this**.**object** = student;  
  
 }  
  
}

*controllers/AbstractGroupCreator.java:*

**package** lr3.controllers;  
  
**import** lr3.models.CommonInterface;  
  
**import** java.util.Random;  
  
*/\*\*  
 \* This class is a base controller that is used to define common properties of  
 \* of most controllers in the main package.  
 \*  
 \* <p>Please make sure you are using the correct version of Idea,  
 \* if you're not sure which version you are using please contact  
 \* the administrator. The version intended for usage is maven-14.  
 \*  
 \* <p>The documentation for the methods contained in this class includes  
 \* brief descriptions of the <i>implementations</i>. Such descriptions should  
 \* be regarded as <i>implementation notes</i>, rather than parts of the  
 \* <i>specification</i>. Implementors should feel free to substitute other  
 \* algorithms, so long as the specification itself is adhered to. (For  
 \* example, the algorithm used by {****@code*** *sort(Object[])} does not have to be  
 \* a MergeSort, but it does have to be <i>stable</i>.)  
 \*  
 \* <p>This class is a member of the  
 \* <a href="https://github.com/gotoindex/javalabs">  
 \* Java Learning Course</a>.  
 \*  
 \** ***@author*** *Akim Vladimirov  
 \*/***public abstract class** AbstractGroupCreator<T **extends** CommonInterface> {  
  
 *// This generator is only used in the gen\_size() method.* **public static final** Random ***generator*** = **new** Random();  
  
 **public** T **object**;  
  
 */\*\*  
 \* This method is used to generate a size of a group (for example, a faculty  
 \* can contain 1-3 chairs).  
 \*  
 \** ***@param from*** *an included minimum.  
 \** ***@param to*** *an included maximum.  
 \*/* **public static int** gen\_size(**int** from, **int** to) {  
  
 **return *generator***.nextInt(to - from + 1) + from;  
  
 }  
  
}

*controllers/GroupCreator.java:*

**package** lr3.controllers;  
  
**import** lr3.models.GroupModel;  
  
*/\*\*  
 \* This class is a controller for the group model.  
 \*  
 \* <p>Please make sure you are using the correct version of Idea,  
 \* if you're not sure which version you are using please contact  
 \* the administrator. The version intended for usage is maven-14.  
 \*  
 \* <p>The documentation for the methods contained in this class includes  
 \* brief descriptions of the <i>implementations</i>. Such descriptions should  
 \* be regarded as <i>implementation notes</i>, rather than parts of the  
 \* <i>specification</i>. Implementors should feel free to substitute other  
 \* algorithms, so long as the specification itself is adhered to. (For  
 \* example, the algorithm used by {****@code*** *sort(Object[])} does not have to be  
 \* a MergeSort, but it does have to be <i>stable</i>.)  
 \*  
 \* <p>This class is a member of the  
 \* <a href="https://github.com/gotoindex/javalabs">  
 \* Java Learning Course</a>.  
 \*  
 \** ***@author*** *Akim Vladimirov  
 \*/***public class** GroupCreator **extends** AbstractGroupCreator<GroupModel> {  
  
 **public** GroupCreator(**int** index) {  
  
 GroupModel group = **new** GroupModel();  
 group.**head**.**lastName** += **" (Curator)"**;  
 group.**name** = **"Group "** + index;  
 group.**listPlural** = **"students"**;  
 **for** (**int** i = 0; i < *gen\_size*(5, 8); i++) group.**list**.add((**new** StudentCreator(i + 1)).**object**);  
 **this**.**object** = group;  
  
 }  
  
}

*controllers/ChairCreator.java:*

**package** lr3.controllers;  
  
**import** lr3.models.ChairModel;  
  
*/\*\*  
 \* This class is a controller for the chair model.  
 \*  
 \* <p>Please make sure you are using the correct version of Idea,  
 \* if you're not sure which version you are using please contact  
 \* the administrator. The version intended for usage is maven-14.  
 \*  
 \* <p>The documentation for the methods contained in this class includes  
 \* brief descriptions of the <i>implementations</i>. Such descriptions should  
 \* be regarded as <i>implementation notes</i>, rather than parts of the  
 \* <i>specification</i>. Implementors should feel free to substitute other  
 \* algorithms, so long as the specification itself is adhered to. (For  
 \* example, the algorithm used by {****@code*** *sort(Object[])} does not have to be  
 \* a MergeSort, but it does have to be <i>stable</i>.)  
 \*  
 \* <p>This class is a member of the  
 \* <a href="https://github.com/gotoindex/javalabs">  
 \* Java Learning Course</a>.  
 \*  
 \** ***@author*** *Akim Vladimirov  
 \*/***public class** ChairCreator **extends** AbstractGroupCreator<ChairModel> {  
  
 **public** ChairCreator(**int** index) {  
  
 ChairModel chair = **new** ChairModel();  
 chair.**head**.**lastName** += **" (Chairman)"**;  
 chair.**name** = **"Chair "** + index;  
 chair.**listPlural** = **"groups"**;  
 **for** (**int** i = 0; i < *gen\_size*(3, 6); i++) chair.**list**.add((**new** GroupCreator(i + 1)).**object**);  
 **this**.**object** = chair;  
  
 }  
  
}

*controllers/FacultyCreator.java:*

**package** lr3.controllers;  
  
**import** lr3.models.FacultyModel;  
  
*/\*\*  
 \* This class is a controller for the faculty model.  
 \*  
 \* <p>Please make sure you are using the correct version of Idea,  
 \* if you're not sure which version you are using please contact  
 \* the administrator. The version intended for usage is maven-14.  
 \*  
 \* <p>The documentation for the methods contained in this class includes  
 \* brief descriptions of the <i>implementations</i>. Such descriptions should  
 \* be regarded as <i>implementation notes</i>, rather than parts of the  
 \* <i>specification</i>. Implementors should feel free to substitute other  
 \* algorithms, so long as the specification itself is adhered to. (For  
 \* example, the algorithm used by {****@code*** *sort(Object[])} does not have to be  
 \* a MergeSort, but it does have to be <i>stable</i>.)  
 \*  
 \* <p>This class is a member of the  
 \* <a href="https://github.com/gotoindex/javalabs">  
 \* Java Learning Course</a>.  
 \*  
 \** ***@author*** *Akim Vladimirov  
 \*/***public class** FacultyCreator **extends** AbstractGroupCreator<FacultyModel> {  
  
 **public** FacultyCreator(**int** index) {  
  
 FacultyModel faculty = **new** FacultyModel();  
 faculty.**head**.**lastName** += **" (Faculty Man)"**;  
 faculty.**name** = **"Faculty "** + index;  
 faculty.**listPlural** = **"chairs"**;  
 **for** (**int** i = 0; i < *gen\_size*(1, 2); i++) faculty.**list**.add((**new** ChairCreator(i + 1)).**object**);  
 **this**.**object** = faculty;  
  
 }  
  
}

*controllers/UniversityCreator.java:*

**package** lr3.controllers;  
  
**import** lr3.models.FacultyModel;  
  
*/\*\*  
 \* This class is a controller for the faculty model.  
 \*  
 \* <p>Please make sure you are using the correct version of Idea,  
 \* if you're not sure which version you are using please contact  
 \* the administrator. The version intended for usage is maven-14.  
 \*  
 \* <p>The documentation for the methods contained in this class includes  
 \* brief descriptions of the <i>implementations</i>. Such descriptions should  
 \* be regarded as <i>implementation notes</i>, rather than parts of the  
 \* <i>specification</i>. Implementors should feel free to substitute other  
 \* algorithms, so long as the specification itself is adhered to. (For  
 \* example, the algorithm used by {****@code*** *sort(Object[])} does not have to be  
 \* a MergeSort, but it does have to be <i>stable</i>.)  
 \*  
 \* <p>This class is a member of the  
 \* <a href="https://github.com/gotoindex/javalabs">  
 \* Java Learning Course</a>.  
 \*  
 \** ***@author*** *Akim Vladimirov  
 \*/***public class** FacultyCreator **extends** AbstractGroupCreator<FacultyModel> {  
  
 **public** FacultyCreator(**int** index) {  
  
 FacultyModel faculty = **new** FacultyModel();  
 faculty.**head**.**lastName** += **" (Faculty Man)"**;  
 faculty.**name** = **"Faculty "** + index;  
 faculty.**listPlural** = **"chairs"**;  
 **for** (**int** i = 0; i < *gen\_size*(1, 2); i++) faculty.**list**.add((**new** ChairCreator(i + 1)).**object**);  
 **this**.**object** = faculty;  
  
 }  
  
}

*Run.java:*

**package** lr3;  
  
**import** lr3.controllers.UniversityCreator;  
  
*/\*\*  
 \* This class is basically a main executable class, meaning this is the only  
 \* class in the whole {****@code*** *lr3} package to implement the {****@code*** *main()} method.  
 \* The class uses only standard java methods and objects, and a custom class  
 \* from package {****@code*** *university} inside {****@code*** *lr3}.  
 \*  
 \* <p>Please make sure you are using the correct version of Idea,  
 \* if you're not sure which version you are using please contact  
 \* the administrator. The version intended for usage is maven-14.  
 \*  
 \* <p>The documentation for the methods contained in this class includes  
 \* brief descriptions of the <i>implementations</i>. Such descriptions should  
 \* be regarded as <i>implementation notes</i>, rather than parts of the  
 \* <i>specification</i>. Implementors should feel free to substitute other  
 \* algorithms, so long as the specification itself is adhered to. (For  
 \* example, the algorithm used by {****@code*** *sort(Object[])} does not have to be  
 \* a MergeSort, but it does have to be <i>stable</i>.)  
 \*  
 \* <p>This class is a member of the  
 \* <a href="https://github.com/gotoindex/javalabs">  
 \* Java Learning Course</a>.  
 \*  
 \** ***@author*** *Akim Vladimirov  
 \*/***public class** Run {  
  
 */\*\*  
 \* Initializes a {****@code*** *lr3.controllers.UniversityCreator} object,  
 \* generates a university and displays its contents to the user.  
 \*  
 \** ***@implNote*** *This is the main function. It executes everytime the build is called.  
 \*  
 \** ***@param args*** *regular string-type arguments. Note, that all arguments in  
 \* this case will be ignored.  
 \*/* **public static void** main(String[] args) {  
  
 UniversityCreator university = **new** UniversityCreator(1);  
 university.print();  
  
 }  
  
}

Результати виконання:

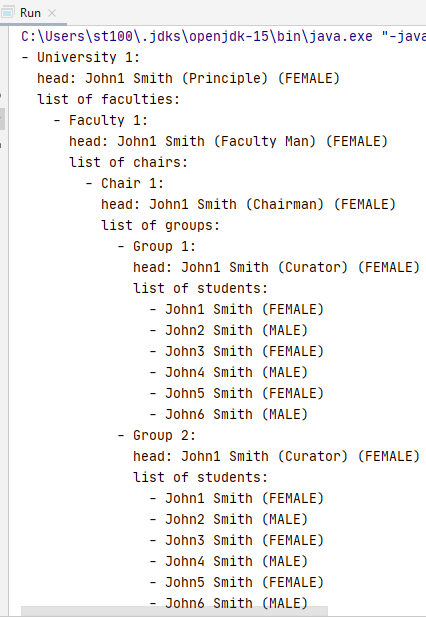


Рисунок 3.1 – Результат виконання програми

Висновок: Під час виконання лабораторної роботи №3 було розроблено програму, що буде створювати та обробляти комплексний об'єкт під назвою університет.