Государственный Университет Молдовы

Факультет Математики и Информатики

Департамент Информатики

“Tehnologii Java”

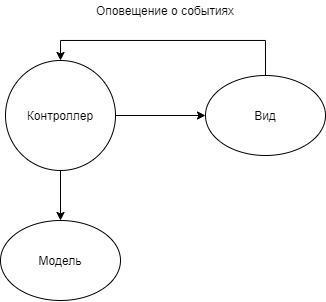
Лабораторная работа 1

Проверил: Латул Георгий

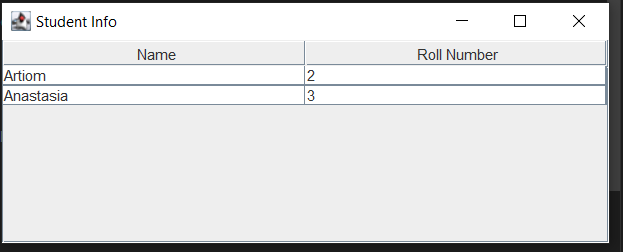
Выполнил: Чобану Артём

Кишинев 2021

Архитектура в моём проекте выглядит следующим образом:



Скриншот:



**Код программы**

Метод main в классе Lab1:

public class Lab1  
{  
 public static void main(String[] args)  
 {  
 //fetch student record based on his roll no from the database  
 Student model = retrieveStudentFromDatabase();  
  
 //Create a view : to write student details  
 StudentView view = new StudentView();  
  
 StudentController controller = new StudentController(view);  
  
 controller.addStudent(new Student("2", "Artiom"));  
 controller.addStudent(new Student("3", "Anastasia"));  
 controller.setStudentRollNo("Artiom", "1");  
 }  
  
 private static Student retrieveStudentFromDatabase()  
 {  
 Student student = new Student();  
 student.setName("Robert");  
 student.setRollNo("10");  
 return student;  
 }  
}

Класс Student:

public class Student  
{  
 private String rollNo;  
 private String name;  
  
 public Student()  
 {  
 }  
  
 public Student(String rollNo, String name)  
 {  
 this.rollNo = rollNo;  
 this.name = name;  
 }

public String getRollNo()  
 {  
 return rollNo;  
 }  
  
 public void setRollNo(String rollNo)  
 {  
 this.rollNo = rollNo;  
 }  
  
 public String getName()  
 {  
 return name;  
 }  
  
 public void setName(String name)  
 {  
 this.name = name;  
 }  
}

Класс StudentController:

public class StudentController  
{  
 private final StudentView view;  
 private final List<Student> students;  
 public StudentController(Student[] defaultStudents)  
 {  
 this.students = Arrays.asList(defaultStudents);  
 view = null;  
 }  
 public StudentController(StudentView view)  
 {  
 this.view = view;  
 students = new ArrayList<>();  
 }  
 public void setStudentRollNo(String studentName, String rollNo)  
 {  
 students.stream()  
 .filter(s -> Objects.equals(s.getName(), studentName))  
 .findFirst()  
 .ifPresent(student ->  
 {  
 student.setRollNo(rollNo);  
 view.SetStudentRollNo(rollNo, students.indexOf(student));  
 });  
 }  
 public String getStudentRollNo(String studentName)  
 {  
 var student = students.stream()  
 .filter(s -> Objects.equals(s.getName(), studentName))  
 .findFirst()  
 .orElse(null);  
 return student != null  
 ? student.getRollNo()  
 : null;  
 }  
 public void addStudent(Student student)  
 {  
 if (student != null)  
 {  
 view.AddStudent(student);  
 }  
 }  
 public void setStudents(Student[] students)  
 {  
 if (students != null && students.length > 0)  
 {  
 view.SetStudents(students);  
 }  
 }  
}

Класс StudentView:

import javax.swing.JFrame;  
import javax.swing.JScrollPane;  
import javax.swing.JTable;  
import javax.swing.table.DefaultTableModel;  
  
public class StudentView  
{  
 private final InfoTable infoTable;  
  
 public StudentView()  
 {  
 infoTable = new InfoTable();  
 }  
  
 public void AddStudent(Student student)  
 {  
 var dataRow = new String[]  
 {  
 student.getName(),  
 student.getRollNo()  
 };  
 infoTable.AddRow(dataRow);  
 }  
  
 public void SetStudents(Student[] students)  
 {  
 var studentTableData = StudentsToStringTableData(students);  
 infoTable.SetTableData(studentTableData);  
 }  
  
 private String[][] StudentsToStringTableData(Student[] students)  
 {  
 String[][] result = new String[students.length][2];  
  
 for (int i = 0; i < students.length; i++)  
 {  
 var currentStudent = students[i];  
 result[i][0] = currentStudent.getName();  
 result[i][1] = currentStudent.getRollNo();  
 }  
  
 return result;  
 }  
 public void SetStudentRollNo(String studentRollNo, int index)  
 {  
 var row = infoTable.GetRowAt(index, 1);  
 System.out.println(row.length);  
 }  
}

Класс InfoTable:

class InfoTable  
{  
 private final Object[] columnNames =  
 {  
 "Name",  
 "Roll Number"  
 };  
  
 private final JFrame frame;  
 private JScrollPane scrollPane;  
 private JTable studentInfoTable;  
  
 InfoTable()  
 {  
 frame = new JFrame();  
 frame.setTitle("Student Info");  
  
 DefaultTableModel tableModel = new DefaultTableModel();  
 tableModel.addColumn("Name");  
 tableModel.addColumn("Roll Number");  
  
 studentInfoTable = new JTable(tableModel);  
 studentInfoTable.setBounds(30, 40, 200, 300);  
  
 JScrollPane scrollPane = new JScrollPane(studentInfoTable);  
 frame.add(scrollPane);  
 frame.setSize(500, 200);  
 frame.setVisible(true);  
 }  
  
 public void SetTableData(String[][] tableData)  
 {  
 studentInfoTable = new JTable(tableData, columnNames);  
 studentInfoTable.setBounds(30, 40, 200, 300);  
  
 frame.remove(scrollPane);  
 scrollPane = new JScrollPane(studentInfoTable);  
 frame.add(scrollPane);  
  
 frame.update(frame.getGraphics());  
 }  
  
 public void AddRow(String[] row)  
 {  
 var model = (DefaultTableModel) studentInfoTable.getModel();  
  
 var rowAmount = model.getDataVector().size();  
 model.addRow(row);  
 model.fireTableRowsUpdated(rowAmount - 2, rowAmount - 1);  
 }  
  
 public String[] GetRowAt(int index, int column)  
 {  
 var model = (DefaultTableModel) studentInfoTable.getModel();  
 var result = model.getValueAt(index, column);  
  
 return (String[]) result;  
 }  
}