Министерство образования Калининградской области

государственное бюджетное учреждение Калининградской области

профессиональная образовательная организация

«Колледж информационных технологий и строительства»

(ГБУ КО ПОО «КИТиС»)

**Отчет по учебной практике**

УП.01 Разработка модулей программного обеспечения для компьютерных систем

по ПМ.01 Разработка модулей программного обеспечения для компьютерных систем

Специальность 09.02.07 «Информационные системы и программирование»

Сроки прохождения практики:

с «12» октября 2022 г. по «01» ноября 2022 г.

Место практики ГБУ КО ПОО «КИТиС»

|  |  |
| --- | --- |
| Выполнил: | студент 4 курса,  группы ИСп 19-2к  Попков Савелий Михайлович  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись) |
| Проверила: | Большакова-Стрекалова Анна Викторовна  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (оценка)  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись, дата) |

Калининград, 2022

Содержание

Введение

Основная часть………………………………………………………………...4

1. Задача 1…………………………………………………………………….4
2. Задача 2…………………………………………………………………….6
3. Задача 3…………………………………………………………………….9
4. Задача 4…………………………………………………………………….10
5. Задача 5…………………………………………………………………….12
6. Задача 6…………………………………………………………………….
7. Задача 7…………………………………………………………………….13
8. Задача 8…………………………………………………………………….16

2. Разработка АИС…………………………………………………………….21

3. Разработка сайта……………………………………………………………30

Введение

Данный отчет составлен по учебной практике УП.01 Разработка модулей программного обеспечения для компьютерных систем.

   Цели:

1. Написать консольные задачи в среде разработки InteliJ IDEA.
2. Разработка АИС.
3. Разработка ER-диаграммы.
4. Сформировать «Техническое задание».
5. Проектирование и реализация интерфейса.
6. Разработать встроенную систему справочного руководства и программного документа «Руководство пользователя».
7. Проведение тестирования.
8. Обеспечить механизм авторизации и работы как минимум двух пользователей, разграничить права.

 Основная часть

Задача 1.1

Условие:

17. Найти число локальных минимумов(Соседями элемента матрецы назовём элементы, имеющие с ним общую сторону или угол.)

Решение:

import java.util.Random;

public class Main {

    public static void main(String[] args){

        int[][] mat1 = new int[5][5];

        int result = 0;

        Random r = new Random();

        for (int i = 0; i < mat1.length; i++){

            for (int j = 0; j < mat1[0].length; j++){

                mat1[i][j] = r.nextInt(10);

                System.out.print(mat1[i][j] + " ");

            }

            System.out.println();

        }

        for (int i = 0; i < mat1.length; i++){

            for (int j = 0; j < mat1[0].length; j++){

                if (i + 1 < mat1.length)

                    if (!(mat1[i][j] < mat1[i + 1][j]))

                        continue;

                if (i - 1 >= 0)

                    if (!(mat1[i][j] < mat1[i - 1][j]))

                        continue;

                if (j + 1 < mat1[0].length)

                    if (!(mat1[i][j] < mat1[i][j + 1]))

                        continue;

                if (j - 1 >= 0)

                    if (!(mat1[i][j] < mat1[i][j - 1]))

                        continue;

                if (i + 1 < mat1.length && j + 1 < mat1[0].length)

                    if (!(mat1[i][j] < mat1[i + 1][j + 1]))

                        continue;

                if (i - 1 >= 0 && j - 1 >= 0)

                    if (!(mat1[i][j] < mat1[i - 1][j - 1]))

                        continue;

                if (i + 1 < mat1.length && j - 1 >= 0)

                    if (!(mat1[i][j] < mat1[i + 1][j - 1]))

                        continue;

                if (i - 1 >= 0 && j + 1 < mat1[0].length)

                    if (!(mat1[i][j] < mat1[i - 1][j + 1]))

                        continue;

                result++;

                System.out.println(mat1[i][j] + ": [" + i + "|" + j + "]");

            }

        }

        System.out.println("Кол-во локальных минимумов: " + result);

    }

}

Результат:

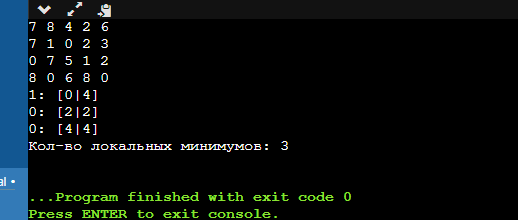


Рисунок 1 – Результат задания 1

Задача 1.2

Условие:

1. Определить класс Дробь (Рациональная Дробь) в виде пары чисел m и n. Объявить и инициализировать массив из k дробей, ввести/вывести значения для массива дробей. Создать массив/список/множество объектов и передать его в метод, который изменяет каждый элемент массива с четным индексом путем добавления следующего за ним элемента.

import java.util.ArrayList;

import java.util.Scanner;

public class Main {

    int x;

    int y;

    Main(){

    }

    Main(int x, int y){

        this.x = x;

        this.y = y;

    }

    public void setX(int x){

        this.x = x;

    }

    public int getX(){

        return x;

    }

    public void  setY(int y){

        this.y = y;

    }

    public int getY(){

        return y;

    }

    public String toString(){

        return x + "/" + y;

    }

    public static void main(String[] args){

        Scanner scanner = new Scanner(System.in);

        System.out.print("Введите количество дробей, которое нужно напечатать: ");

        int k = scanner.nextInt();

        Main[] array = new Main[k];

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

            System.out.print("Введите значение N для дроби " + (i+1) + ": ");

            int n = scanner.nextInt();

            System.out.print("Введите значение M для дроби " + (i+1) + ": ");

            int m = scanner.nextInt();

            array[i] = new Main(n, m);

        }

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

            System.out.println(array[i].toString());

        }

        System.out.println();

        Zadanie(array);

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

            System.out.println(array[i].toString());

        }

    }

    public static void Zadanie(Main[] array){

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

            if(i % 2 == 0 && i+1 < array.length){

                array[i] = array[i+1];

            }

        }

    }

}

Результат:

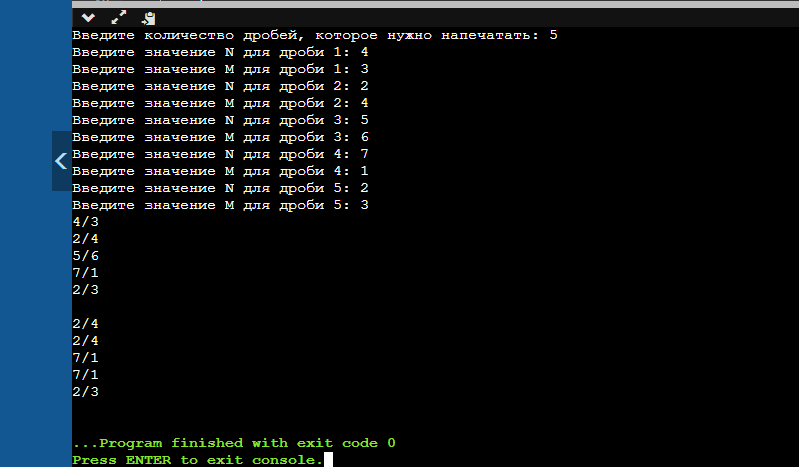


Рисунок 2 – Результат задания 2

Задача 1.3

Условие:

Создать объект класса Щенок, используя классы Животное, Собака.

Методы: вывести на консоль имя, подать голос, прыгать, бегать, кусать.

Public class Animal {

protected String name;

public Animal(String name) {

this.name = name;

}

public String getName() {

return name;

}

}

public class Animal {

protected String name;

public Animal(String name) {

this.name = name;

}

public String getName() {

return name;

}

}

public class Puppy extends Dog {

public Puppy(String name) {

super(name);

}

}

public class PuppyRunner {

public static void main(String[] args) {

Puppy puppy = new Puppy(“Ralf”);

System.out.println(“Name: “+puppy.getName());

System.out.println(“Votes Cast: “ + puppy.votesCast());

System.out.println(“Jump: “ + puppy.jump());

System.out.println(“Run: “+ puppy.run());

System.out.println(“Bite: “+ puppy.bite());

}

}

Результат:

Рисунок 3 – Результат задания 3

Задача 1.4

Условие:

Создать класс Account с внутренним классом, с помощью объектов которого можно хранить информацию обо всех операциях со счетом (снятие, платежи, поступления).

        }  
  
  
    }  
  
    class Payment extends Transaction {  
  
        public Payment(int amount) {  
            this.balance = accountBalance -= amount;  
            this.amount = amount;  
  
        }  
  
  
    }  
  
    class Receipt extends Transaction {  
  
        public Receipt(int amount) {  
            this.balance = accountBalance += amount;  
            this.amount = amount;  
        }  
  
  
    }  
  
    public void newTransaction(typeTransaction type, int amount)  
            throws UnsupportedOperationException {  
  
        Transaction newTransaction;  
  
        switch (type) {  
            case WITHDRAWAL:  
                newTransaction = new Withdrawal(amount);  
                break;  
  
            case PAYMENT:  
                newTransaction = new Payment(amount);  
                break;  
  
            case RECEIPT:  
                newTransaction = new Receipt(amount);  
                break;  
  
            default:  
                throw new UnsupportedOperationException("Not supported yet.");  
  
        }  
  
        newTransaction.type = type;  
        history.add(newTransaction);  
    }  
  
    public String print() {  
        String string = "";  
  
        for( Transaction t : history)  
        {  
            string += t.toString();  
        }  
  
  
        return string;  
    }

}

Результат:

Рисунок 4 – Результат задания 4

Задача 1.5

Условие:

  13. Напечатать без повторения слова текста, у которых первая и последняя буквы совпадают.

import java.util.Scanner;

public class Main {

        public static void main(String[] args) {

                Scanner sc = new Scanner(System.in);

                String text = sc.nextLine();

                String text2 = removeConsecutiveWords(text);

                String[] words = text2.split(" ");

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

                        if (words[i].toCharArray()[0] == words[i].toCharArray()[words[i].length()-1]){

                                System.out.print(words[i]+" ");

                        }

                }

        }

        public static String removeConsecutiveWords(String str) {

                return str

                        .replaceAll("(?i)(\\b\\w+\\W+)(\\1)+", "$1")

                        .replaceAll("(?i)(\\W\*\\w+\\b)(\\1)+", "$1");

        }

}

Результат:

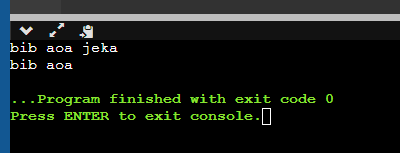


Рисунок 5 – Результат задания 5

Задача 1.6

Условие:

Вариант A

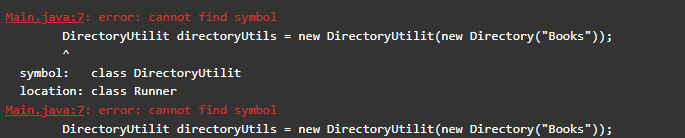
Выполнить задания на основе варианта А задачи 3, контролируя состояние потоков ввода/вывода. При возникновении ошибок, связанных с корректностью выполнения математических операций, генерировать и обрабатывать исключительные ситуации. Предусмотреть обработку исключений, возникающих при нехватке памяти, отсутствии требуемой записи (объекта) в файле, недопустимом значении поля и т. д.

Вариант B

Выполнить задания из варианта В задачи 3, реализуя собственные обработчики исключений и исключения ввода/вывода.

package zadanie.utilit;  
  
import zadanie.utilit.entity.Directory;  
import zadanie.utilit.entity.File;  
import zadanie.utilit.entity.TextFile;  
import zadanie.utilit.utilits.DirectoryUtilit;  
  
  
  
public class Runner {  
  
    public static void main(String[] args) {  
  
        DirectoryUtilit directoryUtils = new DirectoryUtilit(new Directory("Books"));  
  
        File book1 = new TextFile("Играть");  
        File book2 = new TextFile("Изучение Java");  
        File book3 = new TextFile("Алгоритмы");  
        File book4 = new TextFile("Изучение MySql");  
        File book5 = new TextFile("Пока");  
  
        directoryUtils.addFile(book1);  
        directoryUtils.addFile(book2);  
        directoryUtils.addFile(book3);  
        directoryUtils.addFile(book4);  
        directoryUtils.addFile(book5);  
  
        directoryUtils.printFiles();  
  
        System.out.println(book1.getCreated());  
  
        book1.read();  
        System.out.println(book1.getOpened());  
  
        book1.printContent();  
  
        book1.addContent("Привет");  
        System.out.println(book1.getModified());  
  
        book1.getSize();  
  
        book1.printContent();  
  
    }  
}

Результат:

Рисунок 6 – Результат задания 6

Задача 2

Техническое задание.

1) Анализ предметной области. Разработка ER-диаграммы «сущность-связь» (ErWin). Разработка диаграммы вариантов использования, диаграммы классов, диаграммы последовательностей, диаграммы кооперации, диаграммы состояний, диаграммы деятельности (MS Visio, Rational Rose и другие).

2) Разработать базу данных в СУБД MySQL «Поликлиника».

Клиентская часть должна быть разработана на языке Java. Каждый запрос должен быть реализован отдельным методом. Все методы могут принадлежать одному классу. Отдельно должен быть реализован класс, содержащий метод main().

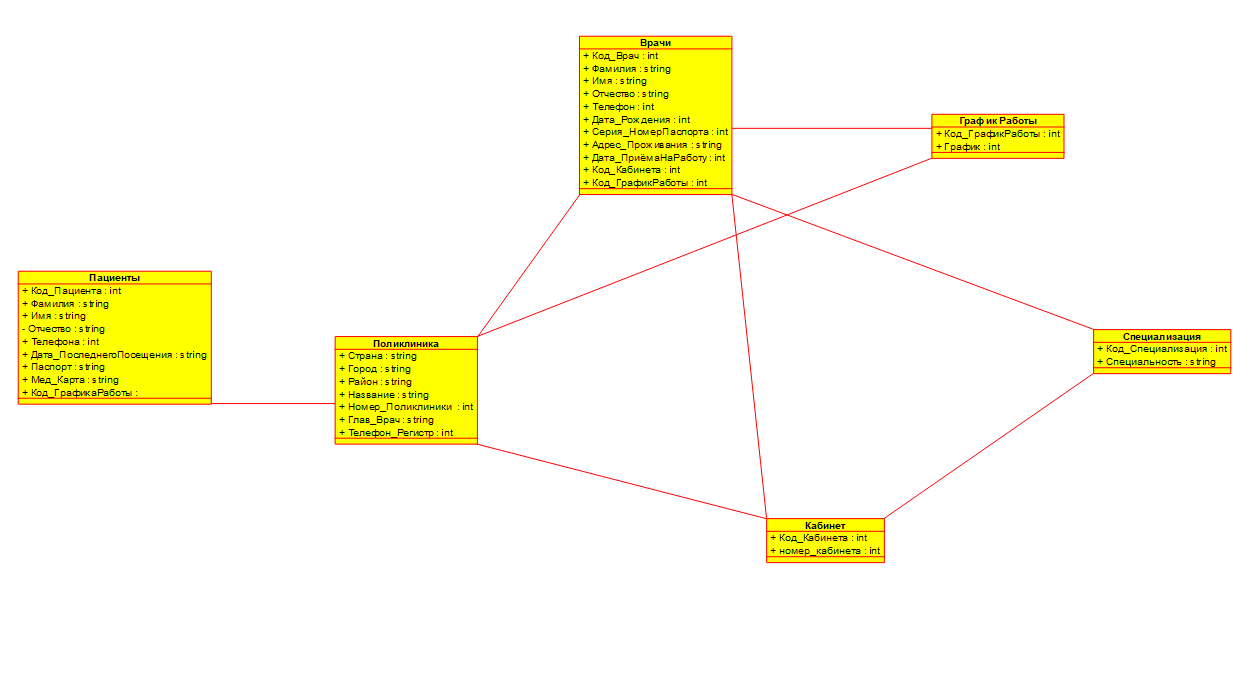
3) Разработка структурной и функциональной схемы АИС.

4) Проектирование и реализация интерфейса.

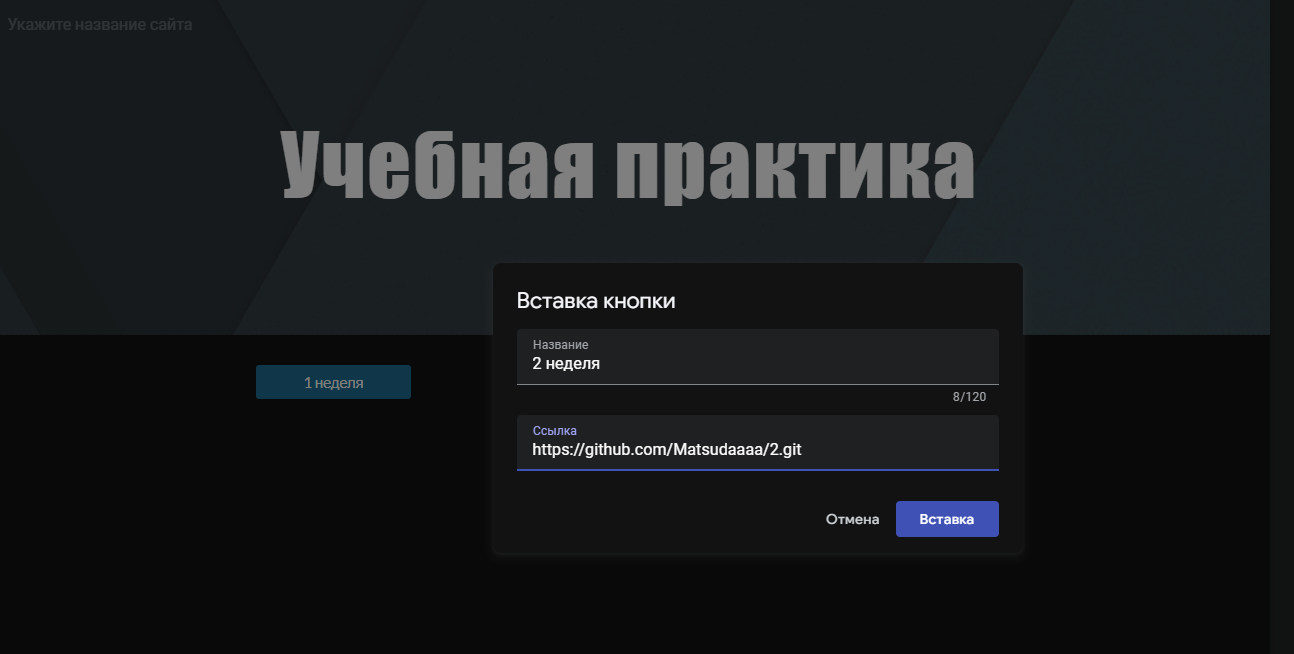
5) Разработка встроенной системы справочного руководства и программного документа «Руководство пользователя».

6) Проведение тестирования.

ER-диаграмма:

Рисунок 7 – ER-диаграмма

Задача 3 – Создание сайта

Рисунок 8 – Добавление кнопок на сайт

Ссылка на сайт:

https://sites.google.com/view/pchelics/главная-страница