

## **ЛАБОРАТОРНАЯ РАБОТА № 3 (10) «СОЗДАНИЕ СОБСТВЕННЫХ КЛАССОВ»**

### **Цель работы:**

1. Изучить основные понятия объектно-ориентированного программирования; структуру классов в Java; модификаторы доступа к данным; особенности и назначение статических методов и объектов.
2. Получить практические навыки разработки программ с использованием объектно-ориентированного подхода.

### **Теоретический материал**

Весь необходимый теоретический материал в презентации к лекции.

### **Порядок выполнения работы**

1. Создать новый пакет
2. Создать класс для хранения информации в соответствии с вариантом задания по следующим правилам:
  - a. Класс должен содержать все указанные поля (можно также добавлять свои поля)
  - b. Все поля должны быть объявлены с модификатором `private`
  - c. Для всех полей определить методы `get` и `set`
  - d. Класс должен содержать не менее двух конструкторов
  - e. Класс должен переопределять метод `toString()`
3. Создать класс для работы с массивом объектов созданного ранее класса. Класс должен включать такие методы как: заполнение, распечатка массива, сортировка по выбранному критерию, и др.
4. Реализовать задания по работе с данным массивом в соответствии с вариантом
5. Вызов отдельных заданий на выполнение оформить в виде меню в главном классе программы
6. Оформить отчет
7. Подготовить ответы на контрольные вопросы

### **Пример**

1. Разработать класс для хранения следующей информации о странах:  
Название страны, Столица, Площадь, Численность населения.
2. В программе создать и заполнить массив данных объектов, в котором

требуется:

- a. Выдать список стран, имеющих площадь больше средней
- b. Найти страну с максимальной площадью
- c. Упорядочить страны по возрастанию численности населения

Общий *алгоритм*.

1. Ввести количество стран.
2. Ввести характеристики стран.
3. Вывести характеристики стран в табличном виде.
4. Вычислить среднюю площадь стран
5. Вывести список стран с площадью больше средней
6. Найти страну с максимальной площадью.
7. Упорядочить страны по численности
8. Вывести упорядоченный список.
9. Закончить.

Проект включает пакет recordcountry, содержащий 3 класса:

- класс **Strana** – включает свойства и методы для работы с объектом-страна
- класс **RecordCountry** – содержит методы для работы с массивом стран
- класс **Main** – главный класс.

Ниже представлена программа, реализующая данный алгоритм.

### Описание класса Strana

```
package recordcountry;

import java.util.Scanner;

public class Strana {
    // поля класса
    private String name; // название страны
    private String capital; // столица
    private double square; // площадь страны
    private int population; // численность

    // конструкторы
    public Strana(String name, String capital, double square, int population)
{
```

```

        this.name = name;
        this.capital = capital;
        this.square = square;
        this.population = population;
    }

    public Strana() {
        Scanner sc = new Scanner(System.in, "cp1251");
        System.out.print("Название : ");
        this.setName(sc.nextLine());
        System.out.print("\tСтолица : ");
        this.setCapital(sc.nextLine());
        System.out.print("\tПлощадь : ");
        this.setSquare(sc.nextDouble());
        System.out.print("\tЧисленность : ");
        this.setPopulation(sc.nextInt());
    }

    // метод toString
    public String toString() {
        return String.format(" Страна: %s ==>   столица: %s \t площадь =
%5.2f млн кв.км \t численность =%d тыс.чел. ",
            name, capital, square, population);
    }

    //геттеры и сеттеры для всех полей
    public String getName() {
        return this.name;
    }

    public void setName(String value) {
        /*в сеттерах обязательно сделать проверку на допустимость
записываемого значения! */
        if (value.isEmpty())
            System.out.println("Недопустимое значение !");
        else this.name = value;
    }

    //для остальных полей аналогично

    public String getCapital() {
        return capital;
    }

    public void setCapital(String capital) {
        if (capital.isEmpty()) System.out.println("Недопустимое значение !");
        else
            this.capital = capital;
    }

    public double getSquare() {
        return square;
    }

    public void setSquare(double square) {
        if (square<=0) System.out.println("Недопустимое значение !");
        else
            this.square = square;
    }

    public int getPopulation() {
        return population;
    }

```

```

    public void setPopulation(int population) {
        if (population<=0) System.out.println("Недопустимое значение !");
        else
            this.population = population;
    }
}

```

## Описание класса RecordCountry

```

package recordcountry;

import java.util.Scanner;

public class RecordCountry {
    private Strana country[] = {
        new Strana("Russia", "Moscow", 17.1, 148),
        new Strana("France", "Paris", 27.1, 137),
        new Strana("Great Britain", "London", 13.1, 150),
        new Strana("Italy", "Rome", 18.1, 104)
    }; //объявляем и инициализируем массив стран

    public void fillCountry(){
        // ВВОД ИНФОРМАЦИИ О СТРАНАХ
        System.out.println("Введите количество стран=> ");
        int n=(new Scanner(System.in,"cp1251")).nextInt(); // количество
        стран
        Strana country[]=new Strana[n];
        System.out.println("Введите информацию о странах: ");

        for(int i=0;i<country.length;i++){
            System.out.println("Страна " +(i+1)+"=>");
            country[i]=new Strana();
        }
    }

    public void printCountry(){
        // ВЫВОД ПОЛУЧЕННОЙ ИНФОРМАЦИИ
        System.out.println("\nХарактеристики стран:");
        for(int i=0;i<country.length;i++){
            System.out.println(country[i]);
        }
    }

    public double avgSquare(){
        // СРЕДНЯЯ ПЛОЩАДЬ СТРАН
        double sumSquare=0;
        for(int i=0;i<country.length;i++){
            sumSquare+=country[i].getSquare();
        }
        if(country.length>0) return sumSquare/country.length;
        return -1;
    }

    public void countriesBiggerThanAvgSquareList(){
        // СТРАНЫ С ПЛОЩАДЬЮ БОЛЬШЕ СРЕДНЕЙ
        System.out.println("\nСписок стран с площадью больше средней:");
        for(Strana str:country)
            if(str.getSquare()>avgSquare()) System.out.println(str);
    }
}

```

```

        public void maxSquareCountry() {
// СТРАНА С МАКСИМАЛЬНОЙ ПЛОЩАДЬЮ
        int pos_max=0;      // номер элемента для страны с максимальной
// площадью (начальное значение)
        for(int i=0;i<country.length;i++){
            if(country[i].getSquare()>country[pos_max].getSquare()){
                pos_max=i;
            }
            System.out.println("\nСтрана с максимальной площадью :");
            System.out.println("\t"+ country[pos_max]+"млн кв. км");
        }

        public void sortByPopulation() {
// СОРТИРОВКА СТРАН ПО ЧИСЛЕННОСТИ
        for(int i=0;i<country.length-1;i++){
            for(int j=0;j<country.length-1-i;j++){
                if(country[j].getPopulation()>country[j+1].getPopulation()){
                    Страна rab=country[j];
                    country[j]=country[j+1];
                    country[j+1]=rab;
                }
            }
            System.out.println("\nОтсортированный список по численности:");//
            for(int i=0;i<country.length;i++){
                System.out.println(""+country[i]);
            }
        }

        public void sortByName() {
// СОРТИРОВКА СТРАН ПО НАЗВАНИЮ
        for(int i=0;i<country.length-1;i++){
            for(int j=0;j<country.length-i-1;j++){
                if(country[j].getName().compareTo(country[j+1].getName())>0){
                    Страна rab=country[j];
                    country[j]=country[j+1];
                    country[j+1]=rab;
                }
            }
            System.out.println("\nОтсортированный список по названиям:");
            for(int i=0;i<country.length;i++){
                System.out.println(""+country[i]);
            }
        }
    }
}

```

## Описание класса Main

```

package recordcountry;

import java.util.Scanner;

public class Main{
    public static void main(String[] args) {
//создаем экземпляр класса для работы с массивом стран
        RecordCountry rc = new RecordCountry();
        cycle: while (true) {
            System.out.println("1. Заполнить массив");
            System.out.println("2. Распечатать");
            System.out.println("3. Найти страны с площадью больше средней");
            System.out.println("4. Найти страну с максимальной площадью");
            System.out.println("5. Упорядочить страны по численности
населения");
            System.out.println("6. Упорядочить страны по названиям");

```

```

        System.out.println("7. Выход");
        System.out.println("Выберете пункт меню (1..7)");
        int c = (new Scanner(System.in)).nextInt();
        switch (c) {
            case 1: rc.fillCountry(); break;
            case 2: rc.printCountry(); break;
            case 3: rc.countriesBiggerThanAvgSquareList(); break;
            case 4: rc.maxSquareCountry(); break;
            case 5: rc.sortByPopulation(); break;
            case 6: rc.sortByName(); break;
            default: break cycle;
        }
    }
}
}
}

```

## Задания

### Вариант 1

1. Разработать класс для хранения информации о людях в записной книжке (ФИО, адрес, телефон, e-mail, день рождения).
2. В программе создать и заполнить массив данных объектов, в котором требуется:
  - а. Выдать список абонентов, сгруппированный по месяцам их дней рождений.
  - б. Найти людей, у которых день рождения в текущем месяце
  - с. Упорядочить массив по фамилиям по алфавиту

### Вариант 2

1. Разработать класс для хранения информации о книгах в библиотеке.
2. В программе создать и заполнить массив данных объектов, в котором требуется:
  - а. Выдать список книг заданного автора
  - б. Найти книги, изданные более 40 лет назад
  - с. Упорядочить книги по названиям.

### Вариант 3

1. Разработать класс для хранения информации об исторических событиях
2. В программе создать и заполнить массив данных объектов, в котором требуется:
  - а. Выдать список событий русской истории 19 века
  - б. Определить, период, количество событий в котором максимально
  - с. Упорядочить события по возрастанию дат

### Вариант 4

1. Разработать класс для хранения информации об учениках класса (Фамилия и инициалы, дата рождения, пол).
2. В программе создать и заполнить массив данных объектов, в котором требуется:
  - а. Выдать список учеников, у которых день рождения в текущем месяце
  - б. Определить кого в классе больше: мальчиков или девочек
  - с. Упорядочить учеников по фамилиям

### **Вариант 5**

1. Разработать класс для хранения информации об автомобиле: марка, номер и фамилия владельца.
2. В программе создать и заполнить массив данных объектов, в котором требуется:
  - а. Найти фамилии владельцев и номера автомобилей заданной марки;
  - б. Определить наиболее популярную марку автомобиля
  - с. Упорядочить марки автомобилей по убыванию количество автомобилей.

### **Вариант 6**

1. Разработать класс для хранения информации, о результатах выборов (фамилия кандидата, дата рождения и количество набранных голосов).
2. В программе создать и заполнить массив данных объектов, в котором требуется:
  - а. Определить были ли нарушения в ходе выборов (нарушение фиксируется, если суммарное количество набранных голосов всеми кандидатами превышает общее количество избирателей)
  - б. Определить, требуется ли проведение повторных выборов (в случае, если ни один из кандидатов не набрал более 1/3 голосов от общего количества)
  - с. Упорядочить кандидатов по убыванию количества набранных голосов

### **Вариант 7**

1. Разработать класс для хранения информации об учебной нагрузке школьных учителей (ФИО учителя, предмет, класс, часы)
2. В программе создать и заполнить массив данных объектов, в котором требуется:
  - а. По каждому учителю определить его суммарную учебную нагрузку
  - б. Найти всех преподавателей-математиков, ведущих занятия в 7-х классах
  - с. Отсортировать учителей по алфавиту.

### **Вариант 8**

1. Разработать класс для хранения информации о студентах первого курса: фамилия, нуждается ли в общежитии, что окончил, какой язык изучал.
2. В программе создать и заполнить массив данных объектов, в котором требуется:
  - a. Определить, сколько человек нуждаются в общежитии;
  - b. Получить список студентов, изучавших французский язык в техникуме
  - c. Упорядочить студентов по фамилиям

### **Вариант 9**

1. Разработать класс для хранения информации о результатах соревнований: название команды, вид соревнования, количество набранных очков).
2. В программе создать и заполнить массив данных объектов, в котором требуется:
  - a. Определить победителей в каждом виде соревнований
  - b. Выбрать команду, занявшую первое место по общему зачету.
  - c. Упорядочить список команд, в зависимости от занятого места.

### **Вариант 10**

1. Разработать класс для хранения информации о результатах сдачи экзаменов абитуриентами (фамилия, экзамен\_1, экзамен\_2, экзамен\_3, индивидуальные\_достижения)
2. В программе создать и заполнить массив данных объектов, в котором требуется:
  - a. Определить списки абитуриентов, зачисленных в институт на 90% мест, если общее число мест задано
  - b. Определить списки студентов, не попавших в первый список, но имеющих индивидуальные достижения
  - c. Отсортировать список студентов по убыванию суммарного балла по экзаменам.

### **Вариант 11**

1. Разработать класс для хранения информации о заказах книг читателями: фамилия, книга, дата заказа, дата выдачи книги.
2. В программе создать и заполнить массив данных объектов, в котором требуется:
  - a. Определить самый маленький срок, за который нашли книгу;
  - b. Определить сколько заказов не было удовлетворено;
  - c. Упорядочить читателей по количеству заказанных книг.



### **Вариант 12**

1. Разработать класс для хранения информации о посылках: город, улица, дом, квартира, кому, ценность.
2. В программе создать и заполнить массив данных объектов, в котором требуется:
  - a. Определить сколько посылок отправлено в г. Самару;
  - b. Определить сколько и куда (список городов) отправлено посылок ценностью выше 10 рублей;
  - c. Упорядочить список посылки по городам назначения.

### **Вариант 13**

1. Разработать класс для хранения информации по читательским билетам: фамилия читателя, информация о книге (автор, название) дата выдачи книги).
2. В программе создать и заполнить массив данных объектов, в котором требуется:
  - a. Определить, кто брал книгу Ефремова "Таис Афинская" 15.05.19
  - b. Найти количество книг заданного автора
  - c. Упорядочить читателей по убыванию количества выданных им книг.

### **Вариант 14**

1. Разработать класс для хранения информации о результатах сдачи экзаменов: предмет, номер группы, дата экзамена, фамилия студента, оценка.
2. В программе создать и заполнить массив данных объектов, в котором требуется:
  - a. Определить сколько человек не сдали информатику, выдать их списки: фамилия, номер группы.
  - b. Определить сколько групп сдавали экзамены 11.06.18 и какие
  - c. Упорядочить список предметов по убыванию среднего балла за экзамен.

### **Вариант 15**

1. Разработать класс для хранения информации о расписании занятий: предмет, преподаватель, номер группы, день недели, номер пары, аудитория).
2. В программе создать и заполнить массив данных объектов, в котором требуется:
  - a. Определить у каких групп совпадают аудитории на занятиях;
  - b. Определить какая нагрузка у заданного преподавателя на этой неделе.

- с. Упорядочить аудитории по убыванию общего количества пар в неделю

### **Вариант 16**

1. Разработать класс для хранения информации о результатах олимпиады: фамилия участника, номер школы, класс, занятое место.
2. В программе создать и заполнить массив данных объектов, в котором требуется:
  - а. Вывести списки школ, занявших призовые места;
  - б. Определить, какая из школ заняла больше всех призовых мест;
  - с. Упорядочить учеников, занявших первые места, и вывести с указанием их класса.

### **Вариант 17**

1. Разработать класс для хранения информации о контактах в записной книжке: фамилия, рабочий телефон, домашний телефон, место работы
2. В программе создать и заполнить массив данных объектов, в котором требуется:
  - а. Определить, есть ли в записной книжке сведения о знакомом с фамилией на букву "Ф", если есть напечатать его фамилию и телефон;
  - б. Определить количество контактов, по которым известен только рабочий номер
  - с. Упорядочить список контактов по фамилиям.

### **Вариант 18**

1. Разработать класс для хранения информации о результатах сдачи экзаменов на первой сессии: фамилия, номер группы, оценки (по геометрии, алгебре, физике и информатике).
2. В программе создать и заполнить массив данных объектов, в котором требуется:
  - а. Получить список студентов, имеющих задолженность хотя бы по одному предмету,
  - б. Определить качественную успеваемость студентов группы (процент студентов, сдавших экзамены на 4 и 5),
  - с. Упорядочить список предметов по убыванию среднего балла по нему.

### **Вариант 19**

1. Разработать класс для хранения информации о багаже пассажира: ФИО пассажира, количество вещей и общий вес вещей.

2. В программе создать и заполнить массив данных объектов, в котором требуется:
  - a. Найти число пассажиров, имеющих более двух вещей
  - b. Определить список пассажиров, количество вещей которых превосходит среднее число вещей
  - c. Упорядочить список пассажиров по ФИО

### **Вариант 20**

1. Разработать класс для хранения информации о студента: фамилия, имя, отчество, пол, возраст, курс.
2. В программе создать и заполнить массив данных объектов, в котором требуется:
  - a. Определить номер курса, на котором наибольший процент мужчин,
  - b. Найти самые распространенные мужские и женские имена;
  - c. Упорядочить студентов по возрасту, отдельно по каждому полу.

### **Вариант 21**

1. Разработать класс для хранения информации о кубиках: размер каждого кубика (длина ребра в сантиметрах), его цвет (красный, зеленый, желтый или синий) и материал (деревянный, металлический, картонный).
2. В программе создать и заполнить массив данных объектов, в котором требуется:
  - a. Найти количество деревянных кубиков с ребром 3 см
  - b. Определить максимальный размер отдельно по каждому виду кубика
  - c. и количество металлических кубиков с ребром, большим 5 см.

### **Вариант 22**

1. Разработать класс для хранения информации о веществах: указывается название вещества, его удельный вес и проводимость (проводник, полупроводник, изолятор).
2. В программе создать и заполнить массив данных объектов, в котором требуется:
  - a. Найти удельные веса и названия всех полупроводников.
  - b. Определить средний вес по каждому типу проводимости
  - c. Упорядочить вещества по возрастанию удельного веса

### **Вариант 23**

1. Разработать класс для хранения информации об экспортируемых товарах: указывается наименование товара, страна, импортирующая товар, и объем поставляемой партии в штуках.
2. В программе создать и заполнить массив данных объектов, в котором требуется:
  - a. Найти страны, в которые экспортируется данный товар,
  - b. Определить по каждому товару общий объем его экспорта.
  - c. Упорядочить страны по убыванию суммарного объема экспорта товаров в них

#### **Вариант 24**

1. Разработать класс для хранения информации об автомобилях: Модель, рег\_номер, год\_выпуска, пробег, стоимость.
2. В программе создать и заполнить массив данных объектов, в котором требуется:
  - a. Определить самый дешевый автомобиль;
  - b. Вывести информацию по автомобилю по регистрационному номеру
  - c. Упорядочить модели автомобилей по убыванию их средней стоимости;

#### **Вариант 25**

1. Разработать класс для хранения информации об фильмах: название\_фильма, год\_выпуска, страна, жанр, стоимость\_проката.
2. В программе создать и заполнить массив данных объектов, в котором требуется:
  - a. Определить фильмы, стоимость проката которых выше средней
  - b. Найти информацию и фильму с минимальным годом выпуска
  - c. Упорядочить жанры фильмов по убыванию суммарной стоимости проката

-

### **Содержание отчета**

Готовится один отчет на бригаду в печатном виде для заданий, указанных в лабораторной работе. Он должен содержать следующие разделы:

- титульный лист;
- задание согласно варианту;

- словесное описание алгоритма;
- текст программы.

### **Контрольные вопросы**

1. Принципы объектно-ориентированного программирования
2. Методы и свойства объектов. Назначение геттеров и сеттеров в классе
3. Что делает конструктор класса? Должен ли он обязательно явно присутствовать в объявлении класса?
4. Назначение и правила использования ключевого слова `this`.
5. Статическое содержимое класса: назначение, правила работы?
6. Ограничения статических методов
7. Для чего используются модификаторы доступа? Какие существуют модификаторы доступа, как они ограничивают доступ к элементам?
8. Состояние, индивидуальность и поведение объектов
9. Правила описания класса