

Часть II

Задание на классы

2.1. Напишите консольное приложение, которое:

- описывает интерфейс **Drawable** с методом построения фигуры *draw()*;
- описывает абстрактный класс **Shape**, реализующий интерфейс **Drawable** и содержащий поле *shapeColor* типа **String** для цвета фигуры и конструктор для его инициализации, абстрактный метод вычисления площади фигуры *calcArea()* и переопределенный метод *toString()*;
- описывает классы **Rectangle**, **Triangle**, **Circle**, которые наследуются от класса **Shape** и реализуют метод *calcArea()*, а также переопределяют метод *toString()*;
- создает набор данных типа **Shape** (массив размерностью не меньше 10 элементов);
- обрабатывает массив:
 - отображает набор данных;
 - вычисляет суммарную площадь всех фигур набора данных;
 - вычисляет суммарную площадь фигур заданного вида;
 - упорядочивает набор данных по увеличению площади фигур, используя объект интерфейса **Comparator**;
 - упорядочивает набор данных по цвету фигур, используя объект интерфейса **Comparator**.

ТРЕБОВАНИЯ.

1. Приложение должно быть написано на языке Java.
2. Использовать только стандартные компиляторы и библиотеки.
3. При кодировании должны быть использованы соглашения об оформлении кода для языка Java.
4. Значения для инициализации объектов выбираются из заранее подготовленных данных (выбранных случайным образом или по порядку следования).
5. При построении приложения использовать архитектурный шаблон MVC.

2.2. Напишите консольное приложение, которое:

- описывает тип данных согласно таблице;
- создает набор данных описанного типа (массив размерностью не меньше 10 элементов);
- использует работу с меню для обработки созданного массива данных согласно таблице;
- использует интерфейс **Comparator** для упорядочивания элементов массива данных согласно таблице.

ТРЕБОВАНИЯ.

1. Приложение должно быть написано на языке Java.
2. Использовать только стандартные компиляторы и библиотеки.
3. При кодировании должны быть использованы соглашения об оформлении кода для языка Java.
4. Значения для инициализации объектов выбираются из заранее подготовленных данных (выбранных случайным образом или по порядку следования).
5. Данные для обработки массива (поиск) выбираются из заранее подготовленных данных случайным образом.
6. Если в результате обработки массива данных не найдено, то необходимо вывести соответствующее сообщение.
7. Для проверки результата работы нужно вывести исходный массив после создания.
8. При построении приложения использовать архитектурный шаблон MVC..

Таблица

<i>Класс</i>	<i>Обработка массива данных</i>
Книжка: Название, Автор, Издательство, Год издания, Количество страниц, Цена	1. Получить список книг указанного автора; 2. Получить список книг, которые изданы указанным издательством; 3. Получить список книг, изданных позже указанного года. 4. Упорядочить книги по издательствам