

## Курс: «Программирование с использованием технологии Java и СУБД Oracle»

### Тема: Объектно-ориентированное программирование. Часть 2

#### Задание 1

Создайте класс **Human**, который будет содержать информацию о человеке.

С помощью механизма наследования, реализуйте класс **Builder** (содержит информацию о строителе), класс **Sailor** (содержит информацию о моряке), класс **Pilot** (содержит информацию о летчике).

Каждый из классов должен содержать необходимые для работы методы.

#### Задание 2

Создать базовый класс «**Животное**» и производные классы «**Тигр**», «**Крокодил**», «**Кенгуру**». С помощью конструктора установить имя каждого животного и его характеристики.

Создайте для каждого класса необходимые методы и поля.

### Задание 3

Запрограммируйте класс **Money** (объект класса оперирует одной валютой) для работы с деньгами.

В классе должны быть предусмотрены поле для хранения целой части денег (доллары, евро, гривны и т.д.) и поле для хранения копеек (центы, евроценты, копейки и т.д.).

Реализовать методы для вывода суммы на экран, задания значений для частей.

На базе класса **Money** создать класс **Product** для работы с продуктом или товаром. Реализовать метод, позволяющий уменьшить цену на заданное число.

Для каждого из классов реализовать необходимые методы и поля.

### Задание 4

Создать базовый класс «Устройство» и производные классы «Чайник», «Микроволновка», «Автомобиль», «Пароход». С помощью конструктора установить имя каждого устройства и его характеристики.

Реализуйте для каждого из классов методы:

- **Sound** — издает звук устройства (пишем текстом в консоль);
- **Show** — отображает название устройства;
- **Desc** — отображает описание устройства.

### Задание 5

Создать базовый класс «Музыкальный инструмент» и производные классы «Скрипка», «Тромбон», «Укулеле», «Виолончель». С помощью конструктора установить имя каждого музыкального инструмента и его характеристики.

Реализуйте для каждого из классов методы:

- **Sound** — издает звук музыкального инструмента (пишем текстом в консоль);
- **Show** — отображает название музыкального инструмента;
- **Desc** — отображает описание музыкального инструмента;
- **History** — отображает историю создания музыкального инструмента.

### Задание 6

Создайте интерфейс **IMath**. В нём должно быть три метода:

- **int Max()** — возвращает максимум;
- **int Min()** — возвращает минимум;
- **float Avg()** — возвращает среднеарифметическое.

Класс **Array**, созданный ранее, должен имплементировать интерфейс **IMath**.

Метод **Max** — возвращает максимум среди элементов массива.

Метод **Min** — возвращает минимум среди элементов массива.

Метод `Avg` — возвращает среднеарифметическое среди элементов массива.

Напишите код для тестирования полученной функциональности.

### Задание 7

Создайте интерфейс `ISort`. В нём должно быть два метода:

- `void SortAsc()` — сортировка по возрастанию;
- `void SortDesc()` — сортировка по убыванию.

Класс `Array`, созданный ранее, должен имплементировать интерфейс `ISort`.

Метод `SortAsc` — сортирует массив по возрастанию.

Метод `SortDesc` — сортирует массив по убыванию.

Напишите код для тестирования полученной функциональности.