#### Абстрактные классы

Создайте несколько классов:

* абстрактный класс Человек;
* класс Клиент;
* класс Работник банка

В класс Человек добавьте:

* поля: имя, фамилия;
* метод для получение имени;
* метод для получение фамилии;
* абстрактный метод для вывода всей информации;
* конструктор для установки значений.

В класс Клиент добавьте:

* наследование класса Человек;
* реализация функции для вывода информации;
* строковое поле «название банка»;
* конструктор для установки всех значений;

В класс Работник банка добавьте:

* наследование класса Человек;
* реализация функции для вывода информации;
* строковое поле «название банка»;
* конструктор для установки всех значений;

#### Интерфейсы

Подумайте над тем, как можно использовать интерфейсы в связке с предыдущим заданием.

1. Есть класс автомобиль. Добавьте класс Грузовик, который будет наследовать все от класса Автомобиль.

В классе Грузовик создайте поля:

* количество колес;
* максимальный вес.

Также создайте метод newWheels, который устанавливает новое значение в поле «количество колес» и выводит его в консоль.

В класс Грузовик сделайте конструктор, устанавливающий все значения в конструктор главного класса и все значения в класс Грузовик.

Ниже приведен весь код для главного класса Автомобиль:

public class Car {

public int weight;

public String model;

public char color;

public float speed;

public void outPut () {

System.*out*.println("Вес " + model + " составляет " + weight + "кг.");

System.*out*.println("Цвет машины - " + color + " и её скорость - " + speed);

}

public Car (int w, String m, char c, float s) {

weight = w;

model = m;

color = c;

speed = s;

}

public Car () {}

}

1. Создайте два класса: главный и класс-наследник.

В главном классе:

* создайте поле для ввода целого числа;

В классе-наследнике:

* метод для вывода переменной из главного класса.

Создайте объект на основе класса наследника и выведите через метод переменную с главного класса.

1. Создайте главный класс. Добавьте в него:

* метод для получения возраста пользователя;

Создайте класс-наследник. Добавьте в него:

* переопределенный метод для получения имени пользователя;

Создайте объект на основе класса-наследника и используйте для него метод из класса.

Подсказка: подумайте над переменными, так как объект класса Scanner удобнее создать лишь один раз в главном классе