

- Наследование должно применяться только тогда, когда это имеет смысл.
 - При кодировании должны быть использованы соглашения об оформлении кода java code convention.
 - Классы должны быть грамотно разложены по пакетам.
 - Консольное меню должно быть минимальным.
 - Для хранения параметров инициализации можно использовать файлы.
1. **Цветочница.** Определить иерархию цветов. Создать несколько объектов-цветов. Собрать букет (используя аксессуары) с определением его стоимости. Провести сортировку цветов в букете на основе уровня свежести. Найти цветок в букете, соответствующий заданному диапазону длин стеблей.
 2. **Новогодний подарок.** Определить иерархию конфет и прочих сладостей. Создать несколько объектов-конфет. Собрать детский подарок с определением его веса. Провести сортировку конфет в подарке на основе одного из параметров. Найти конфету в подарке, соответствующую заданному диапазону содержания сахара.
 3. **Домашние электроприборы.** Определить иерархию электроприборов. Включить некоторые в розетку. Подсчитать потребляемую мощность. Провести сортировку приборов в квартире на основе мощности. Найти прибор в квартире, соответствующий заданному диапазону параметров.
 4. **Шеф-повар.** Определить иерархию овощей. Сделать салат. Подсчитать калорийность. Провести сортировку овощей для салата на основе одного из параметров. Найти овощи в салате, соответствующие заданному диапазону калорийности.
 5. **Звукозапись.** Определить иерархию музыкальных композиций. Записать на диск сборку. Подсчитать продолжительность. Провести перестановку композиций диска на основе принадлежности к стилю. Найти композицию, соответствующую заданному диапазону длины треков.
 6. **Камни.** Определить иерархию драгоценных и полудрагоценных камней. Отобрать камни для ожерелья. Подсчитать общий вес (в каратах) и стоимость. Провести сортировку камней ожерелья на основе ценности. Найти камни в ожерелье, соответствующие заданному диапазону параметров прозрачности.
 7. **Мотоциклист.** Определить иерархию амуниции. Экипировать мотоциклиста. Подсчитать стоимость. Провести сортировку амуниции на основе веса. Найти элементы амуниции, соответствующие заданному диапазону параметров цены.
 8. **Транспорт.** Определить иерархию подвижного состава железнодорожного транспорта. Создать пассажирский поезд. Подсчитать общую численность пассажиров и багажа. Провести сортировку вагонов поезда на основе уровня комфортности. Найти в поезде вагоны, соответствующие заданному диапазону параметров числа пассажиров.
 9. **Авиакомпания.** Определить иерархию самолетов. Создать авиакомпанию. Посчитать общую вместимость и грузоподъемность. Провести сортировку

- самолетов компании по дальности полета. Найти самолет в компании, соответствующий заданному диапазону параметров потребления горючего.
10. **Таксопарк.** Определить иерархию легковых автомобилей. Создать таксопарк. Подсчитать стоимость автопарка. Провести сортировку автомобилей парка по расходу топлива. Найти автомобиль в компании, соответствующий заданному диапазону параметров скорости.
11. **Страхование.** Определить иерархию страховых обязательств. Собрать из обязательств дериватив. Подсчитать стоимость. Провести сортировку обязательств в деривативе на основе уменьшения степени риска. Найти обязательство в деривативе, соответствующее заданному диапазону параметров.
12. **Мобильная связь.** Определить иерархию тарифов мобильной компании. Создать список тарифов компании. Подсчитать общую численность клиентов. Провести сортировку тарифов на основе размера абонентской платы. Найти тариф в компании, соответствующий заданному диапазону параметров.
13. **Фургон кофе.** Загрузить фургон определенного объема грузом на определенную сумму из различных сортов кофе, находящихся к тому же в разных физических состояниях (зерно, молотый, растворимый в банках и пакетиках). Учитывать объем кофе вместе с упаковкой. Провести сортировку товаров на основе соотношения цены и веса. Найти в фургоне товар, соответствующий заданному диапазону параметров качества.
14. **Игровая комната.** Подготовить игровую комнату для детей разных возрастных групп. Игрушек должно быть фиксированное количество в пределах выделенной суммы денег. Должны встречаться игрушки родственных групп: маленькие, средние и большие машины, куклы, мячи, кубики. Провести сортировку игрушек в комнате по одному из параметров. Найти игрушки в комнате, соответствующие заданному диапазону параметров.
15. **Налоги.** Определить множество и сумму налоговых выплат физического лица за год с учетом доходов с основного и дополнительного места работы, авторских вознаграждений, продажи имущества, получения в подарок денежных сумм и имущества, переводов из-за границы, льгот на детей и материальной помощи. Провести сортировку налогов по сумме.
16. **Счета.** Клиент может иметь несколько счетов в банке. Учитывать возможность блокировки/разблокировки счета. Реализовать поиск и сортировку счетов. Вычисление общей суммы по счетам. Вычисление суммы по всем счетам, имеющим положительный и отрицательный балансы отдельно.
17. **Туристические путевки.** Сформировать набор предложений клиенту по выбору туристической путевки различного типа (отдых, экскурсии, лечение, шопинг, круиз и т.д.) для оптимального выбора. Учитывать возможность выбора транспорта, питания и числа дней. Реализовать выбор и сортировку путевок.
18. **Кредиты.** Сформировать набор предложений клиенту по целевым кредитам различных банков для оптимального выбора. Учитывать возможность

досрочного погашения кредита и/или увеличения кредитной линии.
Реализовать выбор и поиск кредита.

Тестовые задания к главе 4

Вопрос 4.1.

Объявлены классы X и Y. Заполнить пустое место в методе **action()** так, чтобы был корректно вызван метод из класса X (выбрать один).

```
class X {  
    void action(){}
}  
class Y extends X {  
    static void action(int i){  
        _____.action();  
    }
}  
  
a) super  
b) this  
c) X  
d) new X()
```

Вопрос 4.2.

Объявлены два класса в иерархии наследования:

```
class Base {  
    public void method() {  
        System.out.println("base");  
    }
}  
class Derived extends Base {  
    public static void main(String[] args) {  
        Base base = (Base) new Derived();  
        base.method();  
    }
    public void method() {  
        System.out.print("derived");  
    }
}
```

Что будет выведено в результате компиляции и выполнения кода? (выбрать один)

- a) compilation fails
- b) runtime exception
- c) derived
- d) base