

1. Прочитать текст из файла и построить частотный словарь слов.  
Выведете список слов (с частотами) упорядоченный по алфавиту.  
Для решения этой задачи пригодится Map.
2. Написать метод, который на входе получает коллекцию объектов, а возвращает коллекцию уже без дубликатов.  
Для решения этой задачи пригодится Set.
3. Напишите метод, который добавляет 1млн элементов в ArrayList и LinkedList.  
Напишите метод, который выбирает из заполненного списка элемент наугад 100000 раз. Замерьте время, которое потрачено на это. Сравните результаты, предположите, почему они именно такие.
4. Опишите класс User с одним полем name. Добавьте конструктор, сеттер и геттер.

```
class User {  
    private String name;  
  
    ...  
}
```

Создайте Map, в котором для каждого пользователя хранится количество очков, заработанных в какой-то игре (Map<User, Integer>)

Напишите программу, которая считывает с консоли имя и показывает, сколько очков у такого пользователя.

Сами данные можно добавить в Map при создании или сгенерировать случайно.

5. Написать итератор по массиву (одномерному, потом обобщить до двумерного)
6. Метод получает на вход массив элементов типа K.

Вернуть надо объект Map<K, Integer>, где K — Значение из массива, а Integer количество вхождений в массив.

```
<K> Map<K, Integer> arrayToMap(K[] ks);
```

7. Метод получает на вход массив Map<K, V>

Вернут надо объект Map<V, List<K>>.

То есть поменять ключи и значения в HashMap.

Так как значения могут совпадать, то в итоговой Map они уже пакуются в List

```
<K, V> Map<V, List<K>> mapToMap(Map<K, V> map);
```