Билет 3

# 1. Что такое исключения?

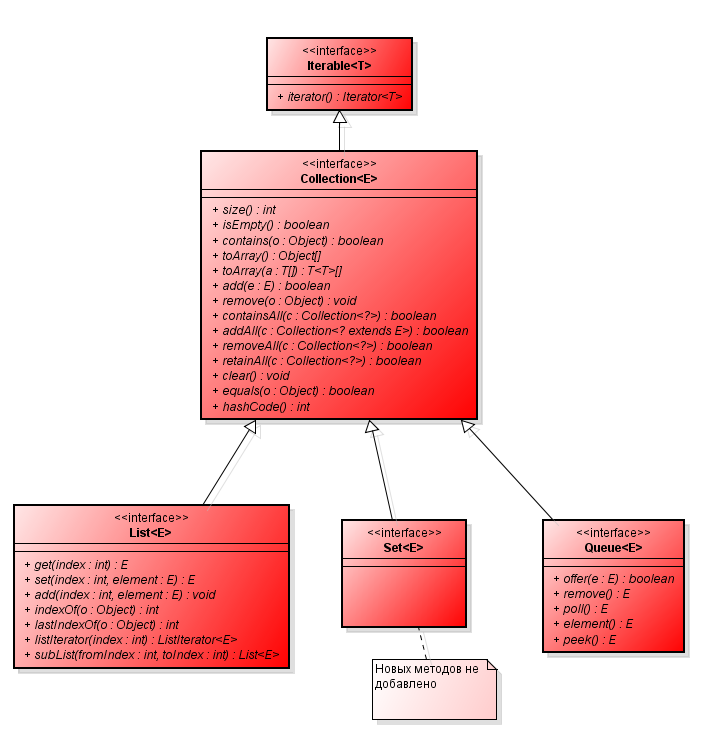
Нередко в процессе выполнения программы могут возникать ошибки, при том необязательно по вине разработчика. Некоторые из них трудно предусмотреть или предвидеть, а иногда и вовсе невозможно. Так, например, может неожиданно оборваться сетевое подключение при передаче файла. Подобные ситуации называются **исключениями.**

# 2. Расскажите про интерфейс Collection и реализацию его расширяющих интерфейсов.

Как видно с диаграммы, интерфейс **Collection**не является базовым. Интерфейс **Collection**расширяет интерфейс **Iterable**, у которого есть только один метод iterator(). Это значит что любая коллекция, которая есть наследником **Iterable**должна возвращать итератор.

[**Итератор**](http://ru.wikipedia.org/wiki/%D0%98%D1%82%D0%B5%D1%80%D0%B0%D1%82%D0%BE%D1%80) -   объект, абстрагирующийся за единым интерфейсом доступ к элементам коллекции. Итератор это паттерн позволяющий получить доступ к элементам любой коллекции без вникания в суть ее реализации.

**Collection**расширяют интерфейсы **List, Set**и **Queue**. Давайте рассмотрим, зачем нужен каждый.  
    1. **List**- Представляет собой неупорядоченную коллекцию, в которой допустимы дублирующие значения. Иногда их называют последовательностями (sequence ). Элементы такой коллекции пронумерованы, начиная от нуля, к ним можно обратиться по индексу.  
    2. **Set**- описывает неупорядоченную коллекцию, не содержащую повторяющихся элементов. Это соответствует математическому понятию множества (set).  
    3. **Queue**- очередь. Это коллекция, предназначенная для хранения элементов в порядке, нужном для их обработки. В дополнение к базовым операциям интерфейса Collection, очередь предоставляет дополнительные операции вставки, получения и контроля.

**