

КОЛЛЕКЦИИ

Создание набора данных

WWW.JAVABEGIN.RU

БЛОК: ОСНОВЫ JAVA



План урока

- ✓ Коллекции в Java
- ✓ Сравнение с массивом
- ✓ Java Collection Framework (JCF)
- ✓ Интерфейсы JCF

Коллекции в Java 8

- **Java Collection Framework (JCF)** – часто называют набор классов и интерфейсов в Java для работы с коллекциями:
 - Коллекция - группировка объектов определенного типа, например коллекция товаров магазина
 - JCF имеет продуманную иерархию интерфейсов и классов (можно даже изучить их, для понимания ООП)
 - Существуют готовые коллекции под различные задачи (очередь, связанный список, индексированный список и т.д.)
 - Чаще всего коллекция используется для хранения объектов
 - Многие фреймворки ORM (Object Relational Mapping) основаны на коллекциях (Hibernate, JPA)
- **Официальная документация**
- <http://docs.oracle.com/javase/8/docs/technotes/guides/collections/changes8.html>

Сравнение с массивом

- **Сходства:**

- Коллекция – это набор данных определенного типа.
- Коллекция имеет имя (ссылка на участок памяти, где хранится коллекция)
- В основе многих коллекций лежат массивы (можно изучить исходный код для понимания)

- **Различия:**

- **Динамическое изменение размера** по необходимости (массив имеет строгий размер, который нельзя превышать)
- Коллекция не может хранить примитивные типы (int, char...) – вместо них нужно использовать wrappers («**wrapper**») – классы обертки: Integer, Character и т.д.
- Более удобная работа с набором данных
- Готовые коллекции для решения множества задач



Java Collection Framework

- Набор системных классов и интерфейсов по работе с коллекциями, построены по принципам ООП
 - Интерфейсы → Абстрактный классы → Классы реализации
- Разделяют:
 - абстрактный уровень (описание поведения разных типов коллекций), задает API
 - уровень реализации – конкретные классы (имплементации) для различных типов коллекций



Интерфейсы JCF

Set – неупорядоченный набор

- не содержит одинаковых значений (дубликатов)
- аналог множества в математике

List – каждый объект имеет свой индекс, по которому его можно быстро получить

- может содержать дубликаты

Queue – односторонняя очередь

- Может использовать принцип FIFO

Deque – двусторонняя очередь

- FIFO
- LIFO

Map – коллекция типа «ключ-значение»

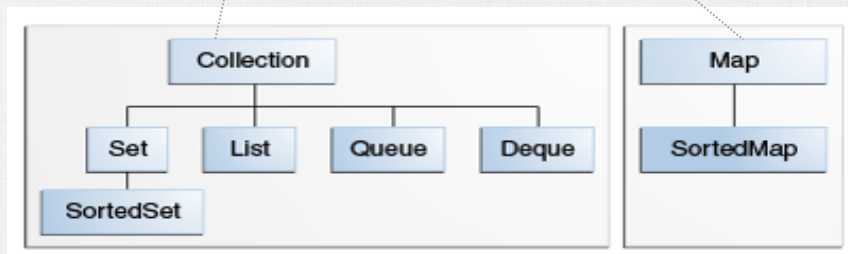
- Уникальность ключей

SortedSet – отсортированный Set

SortedMap – отсортированный Map

Два базовых несвязанных интерфейса JCF:

Collection (набор значений) и **Map (ключ-значение)**



* Подробности:

<http://docs.oracle.com/javase/tutorial/collections/interfaces/index.html>



Что нужно запомнить

- Готовые интерфейсы для разных типов коллекций
- Большое количество реализаций (можно посмотреть через дерево наследования в IDE)
- В каждой конкретной задаче нужно применять коллекцию, которая больше всего подходит под требования
- Сначала нужно понять структуру JCF – потом уже укреплять знания API на практике
- Системные классы по работе с коллекциями (как и многие другие) построены по принципам ООП
 - Интерфейсы → Абстрактный классы → Классы реализации

....

Домашнее задание



- ✓ Прочитать все разделы последовательно:
<http://docs.oracle.com/javase/8/docs/technotes/guides/collections/changes8.html>
Разобрать примеры кодов
- ✓ Изучить дерево зависимостей интерфейсов и классов в IDEA