Коллекции. Итератор и Компаратор

№ урока: 1 **Курс:** Java Professional

Средства обучения: Компьютер с установленной IntelliJ IDEA.

Обзор, цель и назначение урока

Рассмотрение понятий «Итератор», «Компаратор». Интерфейс «Collection», «Map», «Comparable».

Изучив материал данного занятия, учащийся сможет:

- Понимать какие коллекции и карты содержаний существуют
- Понимать работу цикла foreach
- Использовать Iterator, ListIterator.
- Понимать различие между Comparator, Comparable.
- Использовать основные методы Collection, Map.

Содержание урока

1. Коллекции
2. Интерфейсы Collection, Map.
3. Работа цикла foreach.
4. Интерфейсы Iterator, ListIterator.
5. Интерфейсы Comparator, Comparable

Резюме

Collection - коллекция содержит набор объектов (элементов). Здесь определены основные методы для манипуляции с данными, такие как вставка (**add**, **addAll**), удаление (**remove**, **removeAll**, **clear**), поиск (**contains**).

Интерфейс Map соотносит уникальные ключи со значениями.

Ключ — это объект, который вы используете для последующего извлечения данных. Задавая ключ и значение, вы можете помещать значения в объект карты. После того как это значение сохранено, вы можете получить его по ключу.

Итератор - 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.

ListIterator – расширяет интерфейс Iterator, содержит большее количество методов. Использует преимущества реализации списка.

В интерфейсе **Collection<E>** определены методы, которые работают на всех коллекциях:

- **boolean add (E obj)** – добавляет **obj** к вызывающей коллекции и возвращает **true**, если объект добавлен, и **false**, если **obj** уже элемент коллекции;
- **boolean addAll (Collection<? Extends E> c)** – добавляет все элементы коллекции к вызывающей коллекции;
- **void clear()** – удаляет все элементы из коллекции;
- **boolean contains (Object obj**) – возвращает **true**, если вызывающая коллекция содержит элемент **obj**;
- **boolean equals (Object obj)** – возвращает **true**, если соответствующие элементы коллекции эквивалентны;
- **boolean isEmpty ()** – возвращает **true**, если коллекция пуста;
- **Iterator<E> iterator ()** – извлекает итератор;
- **boolean remove (Object obj)** – удаляет **obj** из коллекции;

CyberBionic systematics

CyberBionic Systematics ® 2018
19 Eugene Sverstyuk Str., 5 floor
Kyiv, Ukraine

Tel. 0 800 750 312
E-mail: edu@cbsystematics.com
Site: www.edu.cbsystematics.com | itvdn.com

- **int size ()** – возвращает количество элементов в коллекции;
- **Object[ ] toArray ()** – копирует элементы коллекции в массив объектов;
- **<T> T[ ] toArray (T a[ ])** – копирует элементы коллекции в массив объектов определенного типа.

В интерфейсе **Comparable** объявлен всего один метод - **compareTo(Object obj**), предназначенный для реализации упорядочивания объектов класса. **compareTo()** удобно использовать при сортировке упорядоченных списков или массивов объектов.

Данный метод сравнивает вызываемый объект с **obj**. В отличие от метода equals(), который возвращает **true** или **false**, **compareTo()** возвращает:

- 0, если значения равны;
- Отрицательное значение, если вызываемый объект меньше параметра;
- Положительное значение, если вызываемый объект больше параметра.

В интерфейсе Comparator объявлено два метода **compareTo (Object obj1, Object obj2)** и **equals (Object obj)**.

compareTo (Object obj1, Object obj2) – так же, как и метод **compareTo** интерфейса Comparable, упорядочивает объекты класса.
Точно так же на выходе получает 0, положительное значение и отрицательное значение.

Метод может выбросить исключение **ClassCastException**, если типы объектов не совместимы при сравнении.

Основным отличием интерфейса **Comparator** от **Comparable** является то, что вы можете создавать несколько видов независимых сортировок.

Закрепление материала

- Что такое Collection?
- Что такое Map?
- Что такое Comparable?
- Что такое Iterator?
- Какие методы в Iterator содержаться?
- В чем различие между Iterator и ListIterator?

Дополнительное задание

Задание
В папке с примерами, ex_004_comparable.
Дописать логику, чтобы метод compareTo() осуществил поиск по скорости(если же скорость у 2-х объектов равна, то ищем по цене) -> цене - > моделе -> цвету машины.

Самостоятельная деятельность учащегося

Задание 1
В любой из профильных **книг (Хорстман, Эккель)** найти соответствующие темы и закрепить материал. Использование **YouTube, Quizful** приветствуется.

Задание 2
Вывод на экран элементов **List**:
Создать список, заполнить его на 10 элементов и вывести на экран содержимое используя **iterator**.

CyberBionic systematics
CyberBionic Systematics ® 2018
19 Eugene Sverstyuk Str., 5 floor
Kyiv, Ukraine
Tel. 0 800 750 312
E-mail: edu@cbsystematics.com
Site: www.edu.cbsystematics.com | itvdn.com

Рекомендуемые ресурсы

Oracle: Копаратор
https://docs.oracle.com/javase/tutorial/collections/algorithms/index.html

Oracle: Итератор
https://docs.oracle.com/javase/tutorial/collections/interfaces/collection.html

Oracle: Коллекции
https://docs.oracle.com/javase/tutorial/collections/index.html

CyberBionic systematics

CyberBionic Systematics ® 2018
19 Eugene Sverstyuk Str., 5 floor
Kyiv, Ukraine

Tel. 0 800 750 312
E-mail: edu@cbsystematics.com
Site: www.edu.cbsystematics.com | itvdn.com