

JAXB

Java Collections
3 уровень, 7 лекция

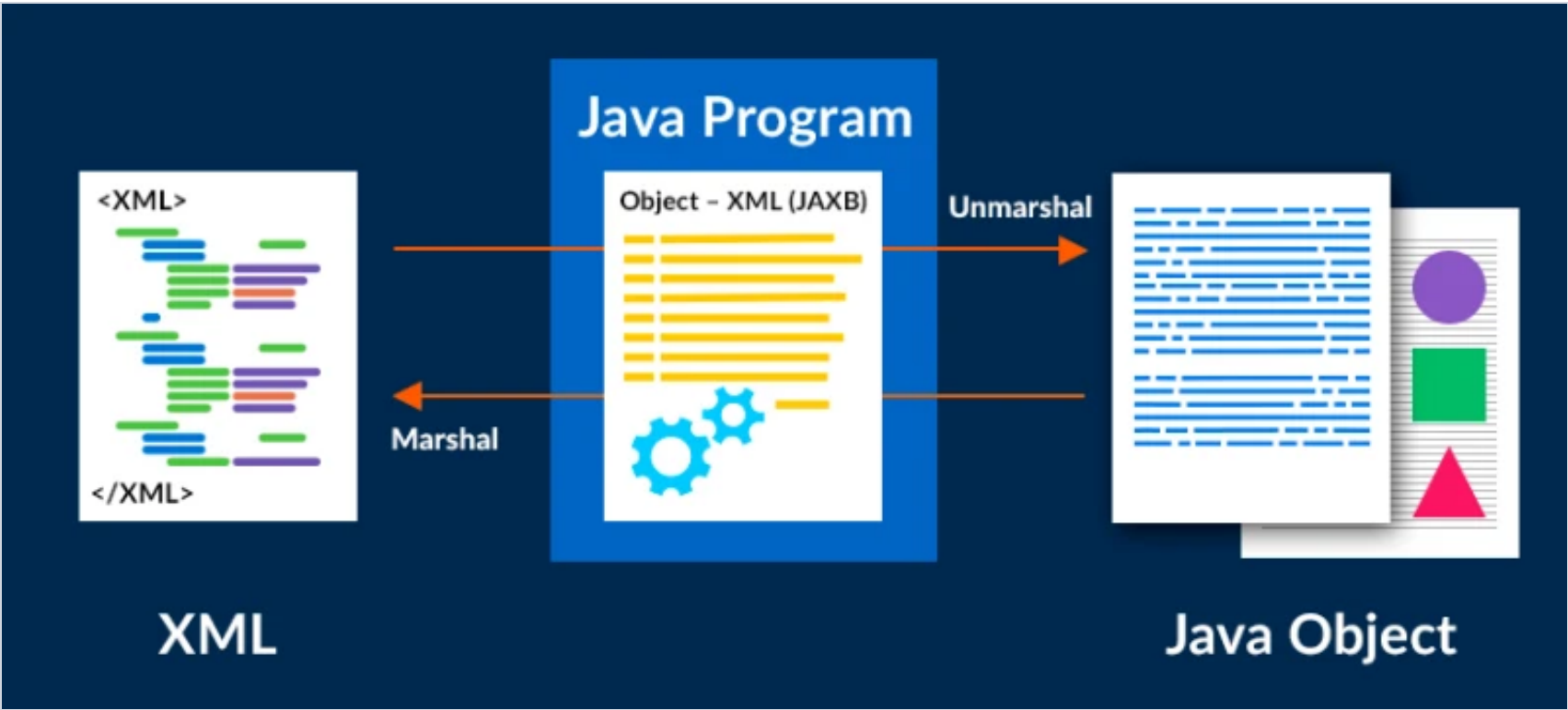
ОТКРЫТА

— Не так быстро, молодой человек! У меня для тебя еще две лекции!

— Две? Ого. Ну, ладно. Чего ни сделаешь ради собственной крутости. Я готов слушать.

— XML, как и JSON, часто применяется при пересылке данных между разными программами и компьютерами. И так же есть несколько фреймворков, которые значительно упрощают жизнь Java-программистов при работе с XML. Сегодня я познакомлю тебя с одним из них.

JAXB – это отличный многофункциональный фреймворк для работы с XML.



JAXB – это часть JDK, поэтому скачивать его отдельно не требуется.

Давай я сначала покажу тебе пример работы с ним, а после мы его разберем. Пример:

Конвертация объекта в XML

```
1
2 public static void main(String[] args) throws JAXBException
3 {
4     //создание объекта для сериализации в XML
5     Cat cat = new Cat();
6     cat.name = "Murka";
7     cat.age = 5;
8     cat.weight = 4;
9
10    //писать результат сериализации будем в Writer(StringWriter)
11    StringWriter writer = new StringWriter();
12
13    //создание объекта Marshaller, который выполняет сериализацию
14    JAXBContext context = JAXBContext.newInstance(Cat.class);
15    Marshaller marshaller = context.createMarshaller();
16    marshaller.setProperty(Marshaller.JAXB_FORMATTED_OUTPUT, Boolean.TRUE);
```

НАЧАТЬ ОБУЧЕНИЕ

```
18     marshall.marshall(cat, writer);
19
20     //преобразовываем в строку все записанное в StringWriter
21     String result = writer.toString();
22     System.out.println(result);
    }
```

Класс, объект которого конвертирует в XML

```
@XmlType(name = "cat")
@XmlRootElement
public class Cat
{
    public String name;
    public int age;
    public int weight;

    public Cat()
    {
    }
}
```

Результат сериализации и вывода на экран:

```
<cat>
<name>Murka</name>
<age>5</age>
<weight>4</weight>
</cat>
```

— Что-то мне этот код один в один напоминает сериализацию в JSON. Тоже аннотации, но там был ObjectMapper, тут Context & Marshaller.

— Ага. Действительно они очень похожи. Jackson писался по образцу JAXB. Но и JAXB то же с кого-то списали. Нельзя выдумать нечто гениальное на пустом месте.

— Похоже на то.

— Ок, вот что там происходит:

В строках 5-8 мы создаем объект класса Cat и заполняем его данными.

Строка 11 – создаем объект Writer для записи результата.

Стока 14 – создаем «контекст». Это аналог ObjectMapper, но от него потом создаются еще два дочерних объекта Marshaller – для сериализации, и Unmarshaller для десериализации. Небольшие отличия от Jackson, но – не принципиально.

Строка 15 – создаем объект Marshaller. Маршалинг – это синоним слова сериализация.

Строка 16 – устанавливает свойство FORMATTED_OUTPUT в TRUE. В результат будут добавлены переносы строки и пробелы, чтобы код был читабельным для человека, а не весь текст в одну строку.

Строка 18 – сериализация объекта.

Строки 21-22 – вывод результата сериализации на экран.

— А что это еще за аннотации @XmlType(name = «cat») и @XmlRootElement?

— @XmlRootElement указывает на то, что этот объект может быть «корнем дерева» элементов в XML. Т.е. быть элементом

@XmlType(name = «cat») указывает на то, что класс участвует в JAXB сериализации, в ней же задано имя, которое будет у XML-тега для этого класса.

Ладно, давай теперь покажу пример десериализации из XML:

Конвертация объекта из XML

```
1 public static void main(String[] args) throws JAXBException
2 {
3     String xmldata = "<cat><name>Murka</name><age>5</age><weight>4</weight></cat>";
4     StringReader reader = new StringReader(xmldata);
5
6     JAXBContext context = JAXBContext.newInstance(Cat.class);
7     Unmarshaller unmarshaller = context.createUnmarshaller();
8
9     Cat cat = (Cat) unmarshaller.unmarshal(reader);
10 }
```

Класс, объект которого десериализуется из XML

```
@XmlType(name = "cat")
@XmlRootElement
public class Cat
{
    public String name;
    public int age;
    public int weight;

    public Cat()
    {
    }
}
```

Все практически аналогично случаю с Jackson. Но на всякий случай объясню все, что тут происходит.

Строка 3 – задаем **строку**, которая хранит xml для десериализации.

Строка 4 – оборачиваем xml-строку в **StringReader**.

Строка 6 – создаем **JAXB-контекст**, куда передаем список классов.

Строка 7 — создаем **Unmarshaller** – объект, который будет выполнять десериализацию.

Строка 9 – десериализуем xml из объекта reader и получаем объект cat типа Cat.

— Теперь это все выглядит чуть ли не очевидно. А еще пару часов назад, я ломал голову — как же это работает.

— Так всегда бывает, когда становишься умнее – сложные вещи становятся простыми.

— Я становлюсь умнее? Это не может не радовать.

— Отлично. Тогда вот тебе список аннотаций, которые ты можешь использовать, чтобы управлять результатом JAXB сериализации:

JAXB-аннотации	Описание
@XmlElement(name)	Ставится около поля. Поле будет представлено в XML-элементом.

НАЧАТЬ ОБУЧЕНИЕ

@XmlAttribute(name)	Ставится около поля. Поле будет представлено в XML-атрибутом! Позволяет задать имя для атрибута.
@XmlElementWrapper(nillable = true)	Ставится около поля. Позволяет задать «обрамляющий тег» для группы элементов.
@XmlType	Ставится около класса. Позволяет задать метод для создания объекта, если конструктор по умолчанию private.
@XmlJavaTypeAdapter	Ставится около поля. Позволяет задать класс, который будет преобразовывать данные поля в строку.

— Как интересно. А можно примеры с этими аннотациями, а то написанное – это одно, а живой пример – это совсем другое.

— Ок. Будет тебе пример. Хотел только добавить, то JAXB позволяет ставить аннотации у методов getter/setter вместо полей.

Обещанный пример:

Конвертация объекта в XML

```
1 public static void main(String[] args) throws JAXBException
2 {
3     //создание объектов Cat&Zoo для сериализации в XML
4     Cat cat = new Cat();
5     cat.name = "Murka";
6     cat.age = 5;
7     cat.weight = 4;
8
9     Zoo zoo = new Zoo();
10    zoo.animals.add(cat);
11    zoo.animals.add(cat);
12
13    //писать результат сериализации будем во Writer(StringWriter)
14    StringWriter writer = new StringWriter();
15
16    //создание объекта Marshaller, который выполняет сериализацию
17    JAXBContext context = JAXBContext.newInstance(Cat.class, Zoo.class);
18    Marshaller marshaller = context.createMarshaller();
19    marshaller.setProperty(Marshaller.JAXB_FORMATTED_OUTPUT, Boolean.TRUE);
20    // самосериализация
21    marshaller.marshal(zoo, writer);
22
23    //преобразовываем все записанное в StringWriter в строку
24    System.out.println(writer.toString());
25 }
```

Класс, объект которого конвертирует в XML

```
1 @XmlType(name = "zoo")
2 @XmlRootElement
3 public class Zoo
```

```
6      @XmlElement(name = "tiger")
7      public List animals = new ArrayList<>();
8  }
9
10 public class Cat
11 {
12     @XmlElement(name = "catname")
13     public String name;
14     @XmlAttribute(name = "age")
15     public int age;
16     @XmlAttribute(name = "w")
17     public int weight;
18
19     public Cat()
20     {
21     }
22 }
```

Результат сериализации и вывода на экран:

```
<zoo>
  <wild-animals>
    <tiger age="5" w="4">
      <catname>Murka</catname>
    </tiger>
    <tiger age="5" w="4">
      <catname>Murka</catname>
    </tiger>
  </wild-animals>
</zoo>
```

Обрати внимание: в этот раз мы сериализуем не объект Cat, а объект типа Zoo, которых хранит коллекцию объектов Cat.

Объект cat в коллекцию был добавлен дважды, поэтому он 2 раза в XML.

У коллекции есть свой тег – «wild-animals» , который обрамляет все элементы коллекции.

Элементы age & weight стали атрибутами age & w.

С помощью атрибута @XmlElement мы поменяли тэг cat на tiger.

Обрати еще внимание на строку 17, теперь мы передаем в JAXB-контекст два класса – Zoo & Cat, т.к. они оба участвуют в сериализации.

— Сегодня очень интересный день – столько нового.

— Ага. Рад за тебя. Сейчас сделаем небольшой перерыв и продолжим.

< Предыдущая лекция

 x23

JavaCoder

Введите текст комментария

Бостон Уровень 26, Russian Federation 5 июня, 15:57 ...

Лишние закрывающие кавычки в десериализации:
String xmldata = "<cat><name>Murka</name><age>5</age><weight>4</weight></cat>"";

Ответить

-

0

+

Бостон Уровень 26, Russian Federation 5 июня, 15:53 ...

@XmlType(name = «cat») указывает на то, что класс участвует в JAXB сериализации, в ней же задано имя, которое будет у XML-тега для этого класса.
Указываю имя name = "dog" - вывод не меняется

Ответить

-

0

+

Jerry Backend Developer 9 января, 21:39 ...

Можно сразу выводить на экран, без Writer(StringWriter):

1

marshaller.marshall(cat, System.out);

Ответить

-

+3

+

SchlechtGut Уровень 51, Москва 25 ноября 2021, 16:12 ...

занятно, тако вывод для последнего примера:
<zoo>
 <wild-animals>
 <tiger xsi:type="cat" age="5" w="4" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
 <catname>Murka</catname>
 </tiger>
 <tiger xsi:type="cat" age="5" w="4" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
 <catname>Murka</catname>
 </tiger>
 </wild-animals>
</zoo>

и откуда там взялись эти ссылки?

Ответить

-

0

+

Дмитрий Уровень 36, Москва 19 декабря 2021, 14:48 ...

пространство имен же, есть у каждого xml

Ответить

-

0

+

Grigory Bryanskiy Уровень 41, Москва, Россия 21 ноября 2021, 04:11 ...

Я целый день бился с ошибками после подключения JAXB.
Все заработало когда я скачал jar отсюда. Версия 2.3.3.
<https://jar-download.com/artifacts/org.glassfish.jaxb/jaxb-runtime/2.3.3/source-code>

Ответить

-

+4

+

Griseo Уровень 43 2 февраля, 16:10 ...

Спасибо дружище! Выручил

Ответить

-

0

+

Griseo Уровень 43 2 февраля, 16:17 ...

я подключал так, скачал файл, разархивировал
чтобы подключить все это, нажимаете в Idea File -> Project Structure - > Libraries ->
там выбираете слева сверху плюсиик и дальше выбираете скаченный файл, затем справа где 3
плюса, самый левый кликаете и выбираете оставшиеся 4 файла

Ответить

-

+1

+

AlinaAlina Уровень 35, Санкт-Петербург 1 сентября 2021, 21:32 ...

В последнем примере у меня вывод получился как в лекции, после того, как я перенесла аннотацию
 @XmlElement(name = "tiger")
из класса Cat в класс Zoo, указав её перед листом animals. До этого вывод был другим)

Ответить

-

0

+

N1NT3ND0___

Уровень 51

26 августа 2021, 23:13

нужно читать и читать...

Ответить

+1

Баба-Яга , молодая, красивая, 153 года

Уровень 40, Москва

14 июня 2021, 08:32

Вот, если у вас JAXB не часть JDK, и вы пользуетесь Maven, можно записать в pom файл такие зависимости:

```
<dependency>
  <groupId>com.sun.xml.bind</groupId>
  <artifactId>jaxb-core</artifactId>
  <version>2.3.0.1</version>
</dependency>
<dependency>
  <groupId>javax.xml.bind</groupId>
  <artifactId>jaxb-api</artifactId>
  <version>2.3.1</version>
</dependency>
<dependency>
  <groupId>com.sun.xml.bind</groupId>
  <artifactId>jaxb-impl</artifactId>
  <version>2.3.1</version>
</dependency>
<dependency>
  <groupId>org.javassist</groupId>
  <artifactId>javassist</artifactId>
  <version>3.25.0-GA</version>
</dependency>
```

Ответить

+2

Артур

Системный администратор

14 февраля, 09:25

Лучшая!

Ответить

0

Иван Сибиряков

Уровень 35, Санкт-Петербург, Россия

20 апреля 2021, 18:47

Галопом по Европам, что называется. Каждая лекция - какая-то новая и весьма обширная тема, которая практически никак толком не раскрывается, а только пугает и заставляет чувствовать себя тупым. Вообще это уже в многопоточности было, но я только сейчас осознал, что курс перестал быть самодостаточным от слова совсем. Комментарии с полезными ссылками спасают. Открыл для себя несколько хороших сайтов. Приходится после каждой лекции углубляться в поднятые этой же лекцией темы.

Ответить

+10

DZ

Уровень 35, Россия

10 декабря 2020, 02:23

Чем больше я узнаю, тем больше понимаю, что ничего не знаю

Ответить

+43

ViktorSlon

QAD testing engineer

24 мая 2021, 14:56

Но другие не знают и этого...

Ответить

+7

Е К

Уровень 41, Краснодар, Россия

11 июня 2021, 15:10

некоторое [пояснение](#) этого эффекта

Ответить

+3

Vadim Zakirov

Java Developer

17 июля 2021, 14:22

фига ты быстро идешь, уже 40 уровень

Ответить

0

Е К

Уровень 41, Краснодар, Россия

17 июля 2021, 21:27

Фига се ты анализируешь)))

Пока решаю большую задачу, на мобиле в свободные минутки читаю следующие лекции - вот и весь фокус)

Ответить

0

Vadim Zakirov

Java Developer

20 июля 2021, 21:56

дааа) наблюдаю
ну успехов нам)

Ответить

0

Ответить

Е К

Уровень 41, Краснодар, Россия

20 июля 2021, 22:38

...

Ответить

0

+

Е К

Уровень 41, Краснодар, Россия

20 июля 2021, 22:40

...

Что за курс, если не секрет? Что стоит? Сколько по времени? Каков формат? Есть ли строгая привязка ко времени?

Покопался там - для java всего два курса - один "оптимизация", второй для "профессионалов", не густо как-то.. или не там копал

Ответить

0

+

Vadim Zakirov

Java Developer

21 июля 2021, 15:54

...

Ну да, для профессионалов 100 тыс, скорее в рассрочку возьму, самый начальный уж не буду брать, сам подготовлюсь Формат онлайн, 4 часа в неделю, строго или нет не скажу, в сентябре когда начну обучаться отпишусь как там вообще).

Есть комплексный, типа java разработчик туда входит этот базовый, прфессиональный и спринг, но стоит 200 тыс. на 15 месяцев.

Ответить

0

+

Е К

Уровень 41, Краснодар, Россия

21 июля 2021, 21:56

...

Красава! Если есть возможность - то лучшая инвестиция это в себя))

Успехов в учёбе!

Ответить

0

+

Vadim Zakirov

Java Developer

31 января, 15:59

...

ура, устроился младшим разрабом

Ответить

+2

+

Е К

Уровень 41, Краснодар, Россия

7 февраля, 12:39

...

Принимай сердечные конгратуляции !! Это кайф, когда есть позитивный результат собственной деятельности!

А я вот чего-то подгорел походу... Не тянет больше кодить хоть ты тресни :(

Ответить

+2

+

Руслан

Уровень 42

24 мая, 22:01

...

Так и не тянет?

Ответить

0

+

Е К

Уровень 41, Краснодар, Россия

10 июня, 15:58

...

Тянет

Пишу свой пэт проджект. Точне совсем недавно "вернулся"...

Но осадок остался. Сейчас всё ещё не определился для себя куда всё-таки двигаться. JavaEE или Kotlin. Пока пишу на Java простенькое взб приложение, а там видно будет

Ответить

0

+

Руслан

Уровень 42

10 июня, 17:16

...

Отлично. Тоже недавно задумался, мб стоит пойти в андроид-разработку. Интересно, где порог входа будет ниже, в JavaEE или android?

Ответить

0

+

Е К

Уровень 41, Краснодар, Россия

11 июня, 15:16

...

Думаю в android порог будет ниже. Там стек поменьше в сравнении с тем, как раздуло жаву. Буквально вчера делал скрининг вакансий и выписывал стек по обоим направлениям. Так вот Kotlin проигрывает в количестве чуть ли не 4 к 1

Ответить

0

+

Показать еще комментарии

ОБУЧЕНИЕ

- Курсы программирования
- Курс Java
- Помощь по задачам

СООБЩЕСТВО

- Пользователи
- Статьи
- Форум

КОМПАНИЯ

- О нас
- Контакты
- Отзывы

НАЧАТЬ ОБУЧЕНИЕ



JavaRush — это интерактивный онлайн-курс по изучению Java-программирования с нуля. Он содержит 1200 практических задач с проверкой решения в один клик, необходимый минимум теории по основам Java и мотивирующие фишки, которые помогут пройти курс до конца: игры, опросы, интересные проекты и статьи об эффективном обучении и карьере Java-девелопера.

ПОДПИСЫВАЙТЕСЬ

ЯЗЫК ИНТЕРФЕЙСА

 Русский

▼

СКАЧИВАЙТЕ НАШИ ПРИЛОЖЕНИЯ

 **ДОСТУПНО В**
Google Play

 Загрузите в
App Store

