JSR 5, 31, 105, 173, 206, 222, 224

XML

Intro

Problem

- XML is a software- and hardware-independent tool for storing and transporting data.
- How to work with XML in Java?

Solution

Package `java.xml`

XML

JSR for XML

- **JSR 5: XML Parsing Specification (JAXP)**
- **JSR 31: XML Data Binding Specification (JAXB)**
- **JSR 105: XML Digital Signature APIs**
- **JSR 173: Streaming API for XML (StAX)**
- **JSR 204: JavaTM API for XML Processing (JAXP) 1.3**
- **JSR 222: JavaTM Architecture for XML Binding (JAXB) 2.0**
- **JSR 224: JavaTM API for XML-Based Web Services (JAX-WS) 2.0**

Java 9+

- **JEP 320: Remove the Java EE and CORBA Modules**
- Removed: **JAXB** (JSR 31, 222) and **JAX-WS** (JSR 224)

Ways to work with XML

- Data bind
- Tree Model
- Streaming
- XPath

Implementation

- **JAXP** (**Java API for XML Processing**) - это набор API (**SAX** + **DOM** + валидация **DTD** + **XSLT**)
- **SAX** (Push Model) → **StAX** (Pull Model) - последовательное чтение из источника XML
- **DOM** (tree) → **JAXB** (mapping) - API для полного зачитывания XML и получения в приложении его готового представления в объектах Java
- **XSL**, **XSLT**
- **XPath** - язык запросов к XML

StAX

StAX

- StAX is a **pull** API. SAX is a **push** API.
- StAX can do both XML reading and writing. SAX can only do XML reading.
- StAX can use **iterator** and **cursor** readers

XMLInputFactory

- `javax.xml.stream.XMLInputFactory` - root component
- This class can create both an `XMLStreamReader` and an `XMLEventReader`
- Can set various properties on the `XMLInputFactory` instance using the `setProperty()` method

XMLEventReader

- `hasNext(): boolean`
- `nextEvent(): XMLEvent`

XMLEvent

- getEventType(): int
- asStartElement(): StartElement
- asEndElement(): EndElement
- asCharacters(): Characters
- isEndElement(): boolean
- asStartElement(): boolean

XMLStreamConstants

- ATTRIBUTE
- CDATA
- CHARACTERS
- COMMENT
- DTD
- END_DOCUMENT
- END_ELEMENT
- ENTITY_DECLARATION

XMLStreamConstants

- ENTITY_REFERENCE
- NAMESPACE
- NOTATION_DECLARATION
- PROCESSING_INSTRUCTION
- SPACE
- START_DOCUMENT
- START_ELEMENT

XMLOutputFactory

- `javax.xml.stream.XMLOutputFactory` - root component
- This class can create both an `XMLStreamWriter` and an `XMLEventWriter`
- Can set various properties on the `XMLOutputFactory` instance using the `setProperty()` method

XMLEventFactory

- `XMLEventFactory.newInstance(): XMLEventFactory`
- `createStartDocument(): StartDocument`
- `createStartElement(String prefix, String namespaceUri, String localName): StartElement`
- `createNamespace(String prefix, String namespaceUri): Namespace`

XMLEventFactory

- `createAttribute(String localName, String value): Attribute`
- `createEndElement(String prefix, String namespaceUri, String localName): EndElement`
- `createEndDocument(): EndDocument`

XPath

XPath

```
XPathFactory xpathFactory = XPathFactory.newInstance();
XPath xpath = xpathFactory.newXPath();
```

XPathExpression

```
XPathExpression xPathExpression = xpath.compile("THIS_IS_YOUR_EXPRESSI
NodeList nodes = (NodeList) xPathExpression.evaluate(doc, XPathConstar
```

XPathConstants

- STRING
- NUMBER
- BOOLEAN
- NODE
- NODESET

Example

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<developers>
    <developer id="1">
        <name>Andrew</name>
        <age>25</age>
        <position>Middle</position>
        <language>Java</language>
    </developer>
    <developer id="2">
        <name>Dima</name>
        <age>21</age>
        <position>Junior</position>
        <language>JS</language>
    </developer>
</developers>
```

Example: как получить developers какого-то уровня?

```
XPathExpression xPathExpression = xpath.compile(
    "/developers/developer[position='" + position + "']/name/text()"
);
```

Example: как получить developers младше какого-то возраста?

```
XPathExpression xPathExpression = xpath.compile(
    "/developers/developer[age<" + age + "]/name/text()"
);
```

Example: как получить имя developer по id?

```
XPathExpression xPathExpression = xpath.compile(
    "/developers/developer[@id='" + id + "']/name/text()"
);
```