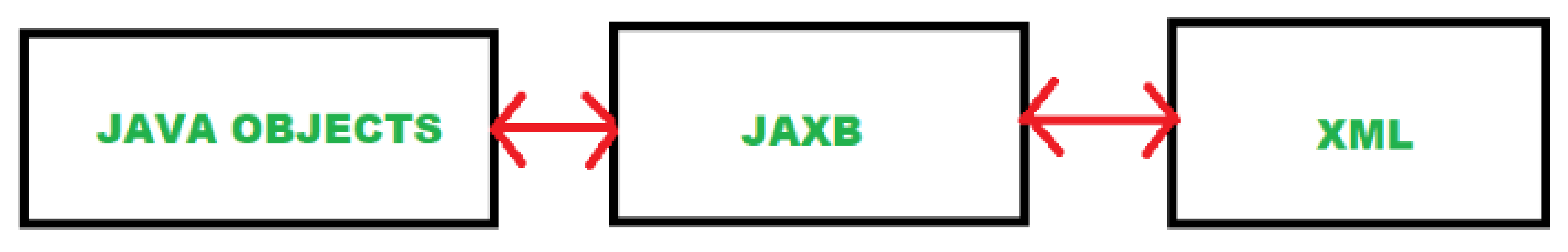
JAVA ARCHITECTURE FOR XML BINDING (JAXB)-IMPLEMENTATION AND EXAMPLES

By- BHAVYA SEHGAL (2021UCS1626)

JAXB INTRODUCTION

Java Architecture for XML Binding (JAXB) defines an API for reading and writing Java objects to and from XML documents. JAXB gives Java developers an efficient and standard way of mapping between XML and Java code. JAXB makes it easier for developers to extend their applications with XML and Web Services technologies.At First JAXB was developed as a separate project but it was finally became part of JDK in Java 6.



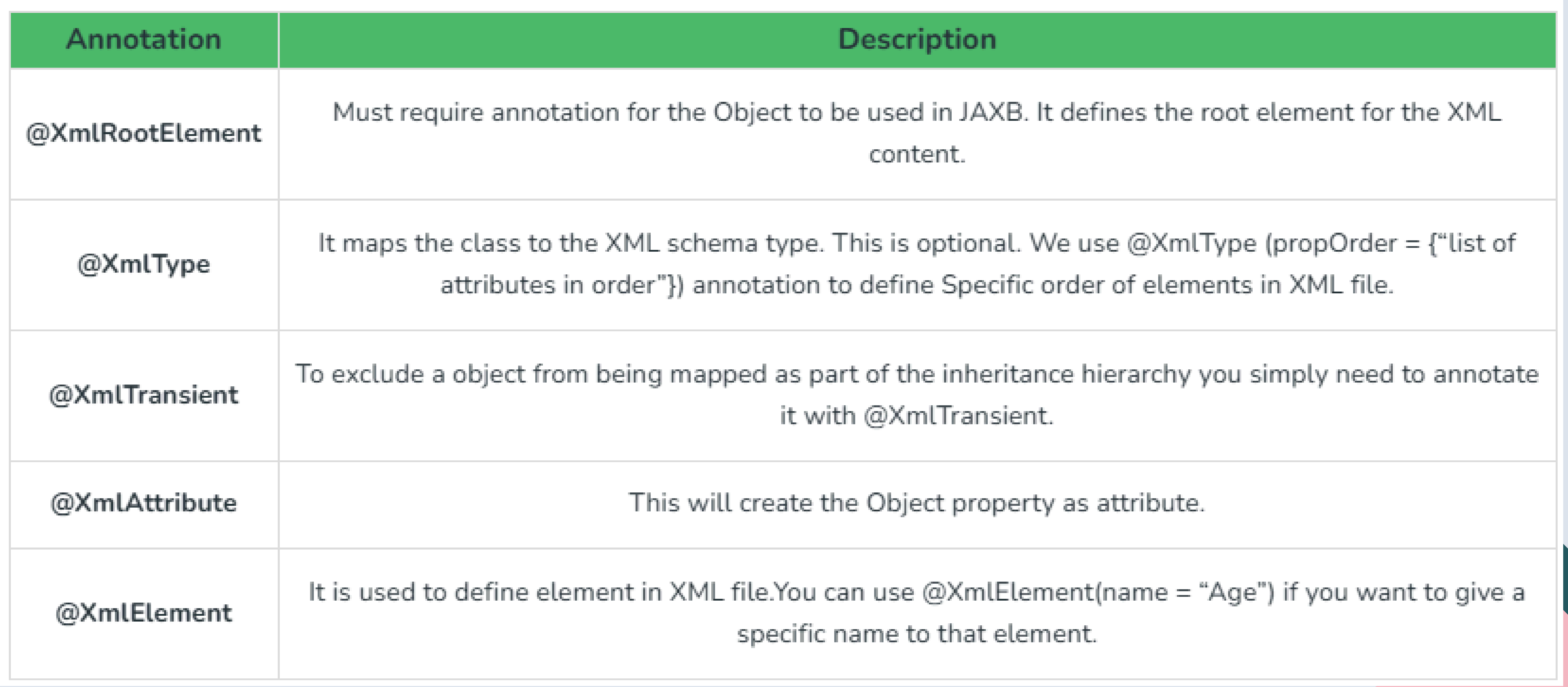
APPLICATION OF JAXB

JAXB framework is helpful to perform the following operations:

- Change XML content into a Java representation.
- Access and update the Java representation
- Change the Java representation into XML content.

JAXB ANNOTATIONS

JAXB uses annotations to indicate the central elements. Some Basics JAXB Annotations which can be used in your java class for JAXB operations are:



Annotation	Description
@XmlRootElement	Must require annotation for the Object to be used in JAXB. It defines the root element for the XML content.
@XmlType	It maps the class to the XML schema type. This is optional. We use @XmlType (propOrder = {"list of attributes in order"}) annotation to define Specific order of elements in XML file.
@XmlTransient	To exclude a object from being mapped as part of the inheritance hierarchy you simply need to annotate it with @XmlTransient.
@XmlAttribute	This will create the Object property as attribute.
@XmlElement	It is used to define element in XML file.You can use @XmlElement(name = "Age") if you want to give a specific name to that element.

TWO BASIC CONCEPT RELATED TO JAXB ARE :

1. **Marshalling** : Convert a java Object to xml

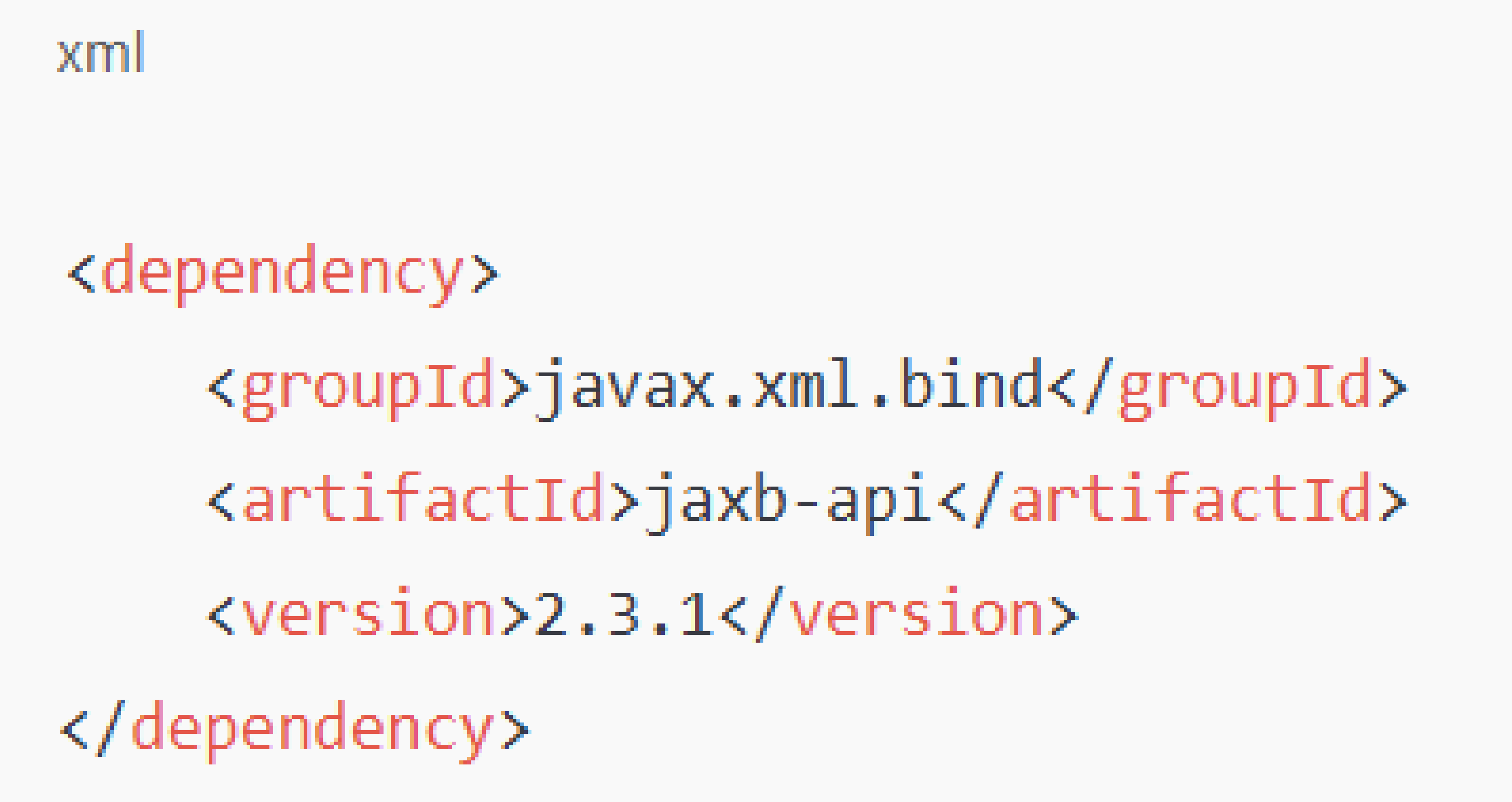
2. **UnMarshalling** : Convert xml to java object

MARSHALLING

Below is the step by step algorithm for converting Java Objects to XML(Marshalling):

1. First Create Java Objects to be Marshalled.
2. Create JAXBContext Object and initializing Marshaller Object.
3. To get the formatted xml output one can set JAXB_FORMATTTED_OUTPUT to True(this Step is optional).
4. Create xml file object by providing location of file as parameter to File Class
5. Call marshal method on Marshaller Object and pass created XML File object to marshal method.
6. Now the XMl file is created.

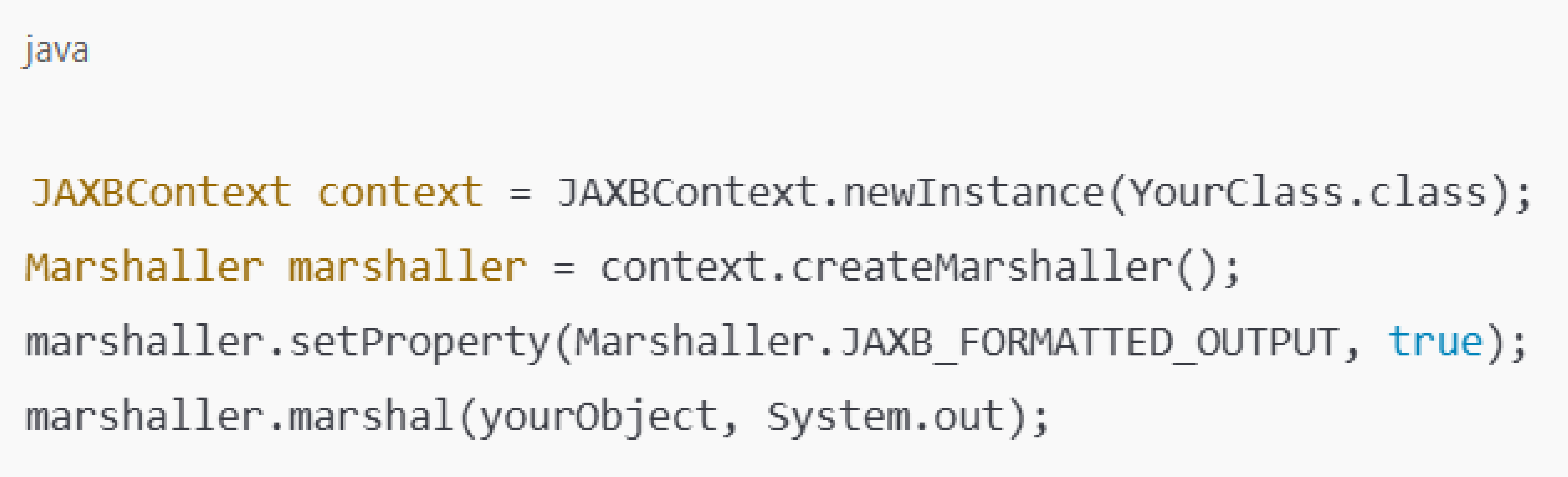
Setting Up JAXB in a Java Project



xml

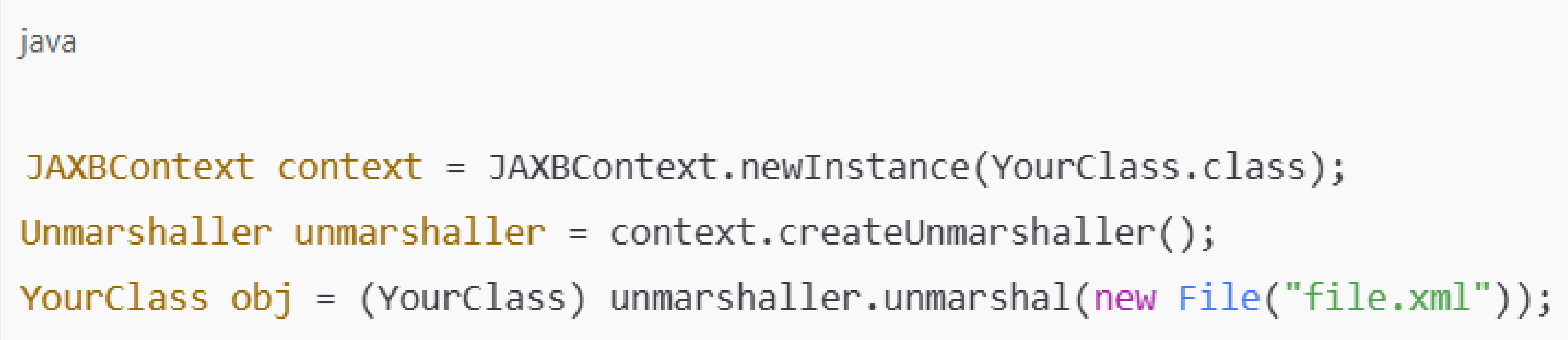
```xml
<dependency>
    <groupId>javax.xml.bind</groupId>
    <artifactId>jaxb-api</artifactId>
    <version>2.3.1</version>
</dependency>
```

MARSHALLING (JAVA OBJECT TO XML)



```java
JAXBContext context = JAXBContext.newInstance(YourClass.class);
Marshaller marshaller = context.createMarshaller();
marshaller.setProperty(Marshaller.JAXB_FORMATTED_OUTPUT, true);
marshaller.marshal(yourObject, System.out);
```

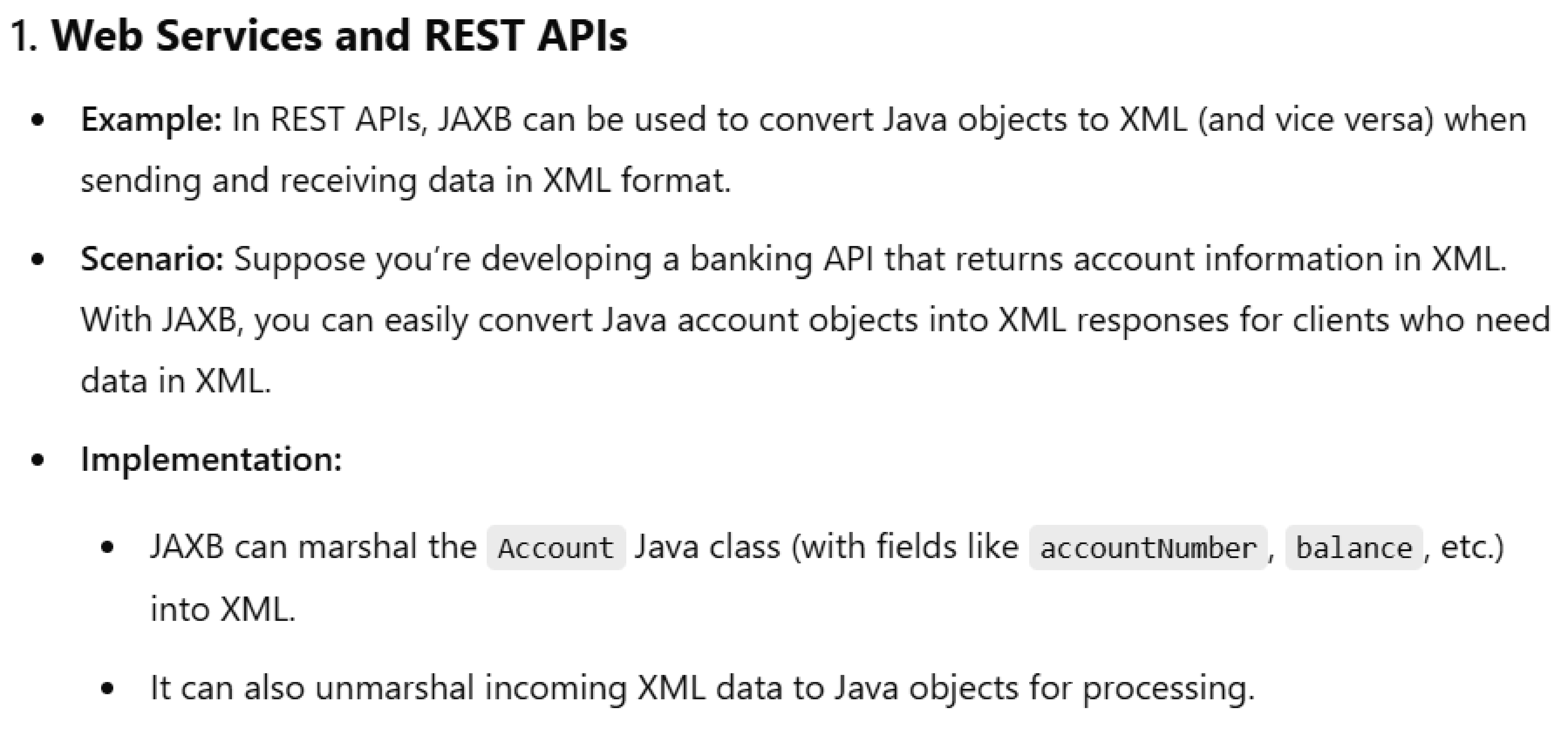
UNMARSHALLING (XML TO JAVA OBJECT)



java

```java
JAXBContext context = JAXBContext.newInstance(YourClass.class);
Unmarshaller unmarshaller = context.createUnmarshaller();
YourClass obj = (YourClass) unmarshaller.unmarshal(new File("file.xml"));
```

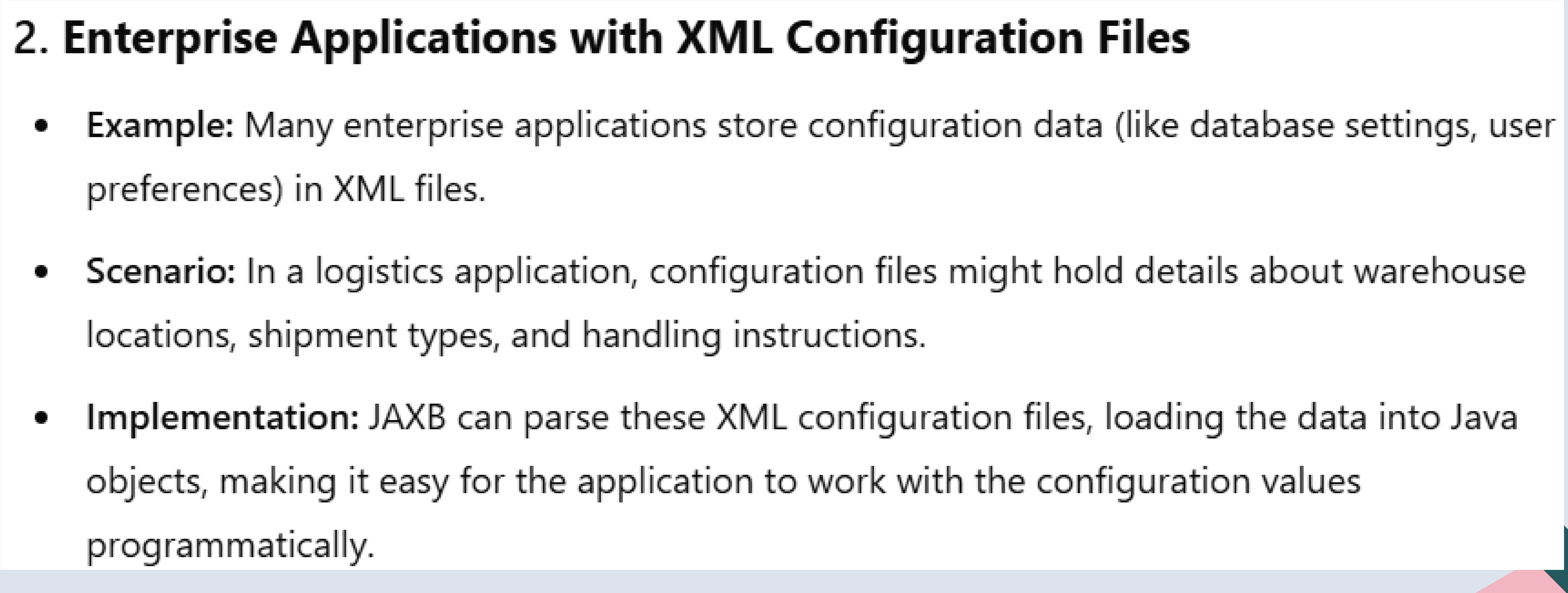
SOME REAL LIFE EXAMPLES OF JAXB



1. Web Services and REST APIs

- **Example:** In REST APIs, JAXB can be used to convert Java objects to XML (and vice versa) when sending and receiving data in XML format.
- **Scenario:** Suppose you're developing a banking API that returns account information in XML. With JAXB, you can easily convert Java account objects into XML responses for clients who need data in XML.
- **Implementation:**
    - JAXB can marshal the `Account` Java class (with fields like `accountNumber`, `balance`, etc.) into XML.
    - It can also unmarshal incoming XML data to Java objects for processing.

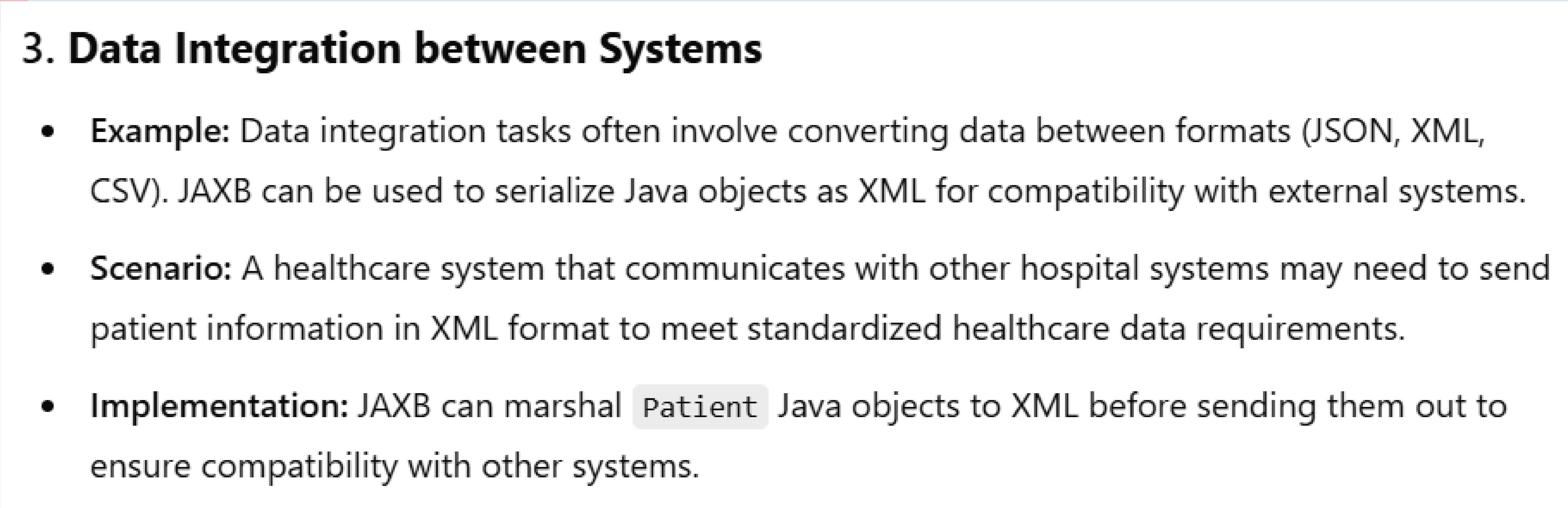
SOME REAL LIFE EXAMPLES OF JAXB



2. Enterprise Applications with XML Configuration Files

- **Example:** Many enterprise applications store configuration data (like database settings, user preferences) in XML files.
- **Scenario:** In a logistics application, configuration files might hold details about warehouse locations, shipment types, and handling instructions.
- **Implementation:** JAXB can parse these XML configuration files, loading the data into Java objects, making it easy for the application to work with the configuration values programmatically.

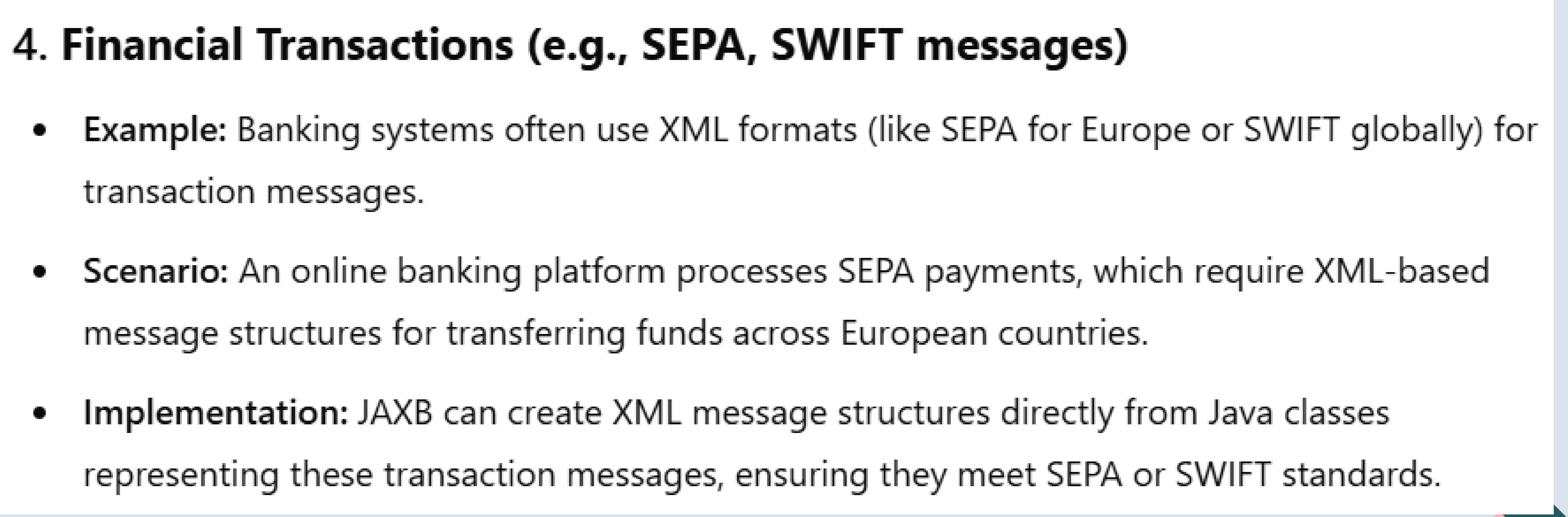
SOME REAL LIFE EXAMPLES OF JAXB



3. Data Integration between Systems

- **Example:** Data integration tasks often involve converting data between formats (JSON, XML, CSV). JAXB can be used to serialize Java objects as XML for compatibility with external systems.
- **Scenario:** A healthcare system that communicates with other hospital systems may need to send patient information in XML format to meet standardized healthcare data requirements.
- **Implementation:** JAXB can marshal `Patient` Java objects to XML before sending them out to ensure compatibility with other systems.

SOME REAL LIFE EXAMPLES OF JAXB



4. Financial Transactions (e.g., SEPA, SWIFT messages)

- **Example:** Banking systems often use XML formats (like SEPA for Europe or SWIFT globally) for transaction messages.
- **Scenario:** An online banking platform processes SEPA payments, which require XML-based message structures for transferring funds across European countries.
- **Implementation:** JAXB can create XML message structures directly from Java classes representing these transaction messages, ensuring they meet SEPA or SWIFT standards.

THANK YOU