Jackson XML e JSON

A biblioteca Jackson é uma ferramenta poderosa para serializar e desserializar dados JSON e XML em Java

Maven

```
<dependency>
    <groupId>com.fasterxml.jackson.core</groupId>
    <artifactId>jackson-databind</artifactId>
    <version>2.15.0</version>
</dependency>
<dependency>
    <groupId>com.fasterxml.jackson.dataformat</groupId>
    <artifactId>jackson-dataformat-xml</artifactId>
    <version>2.15.0</version>
</dependency>
```

JSON

Serialização

```
import com.fasterxml.jackson.databind.ObjectMapper;

public class JsonExample {
    public static void main(String[] args) {
        ObjectMapper objectMapper = new ObjectMapper();
        try {
            User user = new User("Alice", 30);
            String jsonString = objectMapper.writeValueAsString(user);
            System.out.println(jsonString);
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}

class User {
    private String name;
    private int age;

    // Construtor, getters e setters
    public User(String name, int age) {
        this.name = name;
        this.age = age;
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public int getAge() {
        return age;
    }

    public void setAge(int age) {
        this.age = age;
    }
}
```

Deserializar

```
import com.fasterxml.jackson.databind.ObjectMapper;

public class JsonExample {
    public static void main(String[] args) {
        ObjectMapper objectMapper = new ObjectMapper();
        String jsonString = "{\"name\":\"Alice\",\"age\":30}";
        try {
            User user = objectMapper.readValue(jsonString, User.class);
            System.out.println("Name: " + user.getName() + ", Age: " +
user.getAge());
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

XML

Serialização

```
import com.fasterxml.jackson.dataformat.xml.XmlMapper;

public class XmlExample {
    public static void main(String[] args) {
        XmlMapper xmlMapper = new XmlMapper();
        try {
            User user = new User("Alice", 30);
            String xmlString = xmlMapper.writeValueAsString(user);
            System.out.println(xmlString);
        } catch (Exception e) {
```

```java
            e.printStackTrace();
        }
    }
}

class User {
    private String name;
    private int age;

    // Construtor, getters e setters
    public User(String name, int age) {
        this.name = name;
        this.age = age;
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public int getAge() {
        return age;
    }

    public void setAge(int age) {
        this.age = age;
    }
}
```

Deserialização

```java
import com.fasterxml.jackson.dataformat.xml.XmlMapper;

public class XmlExample {
    public static void main(String[] args) {
        XmlMapper xmlMapper = new XmlMapper();
        String xmlString = "<User><name>Alice</name><age>30</age></User>";
        try {
            User user = xmlMapper.readValue(xmlString, User.class);
            System.out.println("Name: " + user.getName() + ", Age: " +
user.getAge());
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```