**Solution: Reading XML data in Groovy**

Groovy script that meets the requirements and extracts the information from the provided XML document

import groovy.xml.XmlSlurper

// Function to parse the XML document and extract information

def parseCatalogXML(String xmlContent) {

def xml = new XmlSlurper().parseText(xmlContent)

xml.book.eachWithIndex { book, index ->

println "Book ${index + 1}:"

println "Author: ${book.author.text()}"

println "Title: ${book.title.text()}"

println "Genre: ${book.genre.text()}"

println "Price: ${book.price.text()}"

println "Publish Date: ${book.publish\_date.text()}"

println "Description: ${book.description.text()}"

println()

}

}

// Main script

def xmlContent = '''

<?xml version="1.0" encoding="UTF-8"?>

<catalog>

<book id="bk001">

<author>Writer</author>

<title>The First Book</title>

<genre>Fiction</genre>

<price>44.95</price>

<publish\_date>2000-10-01</publish\_date>

<description>A great book about something or other.</description>

</book>

<book id="bk002">

<author>Writer2</author>

<title>The Second Book</title>

<genre>Non-Fiction</genre>

<price>29.99</price>

<publish\_date>2001-10-01</publish\_date>

<description>Another great book about something or other.</description>

</book>

</catalog>

'''

// Call the parseCatalogXML function and display the data

parseCatalogXML(xmlContent)

This script defines a function called **parseCatalogXML** that takes an XML string as input. It then uses the **XmlSlurper** class to parse the XML content and iterates over each **<book>** element. For each book, it extracts and prints the required information (author, title, genre, price, publish date, and description) in a clear and organized format. The main script provides the sample XML content, and the **parseCatalogXML** function is called with this content to display the extracted information.