Webservices

Topics

* Webservices overview
* Introduction to REST
* Introduction to Jax-RS
* Resources & requests
* HTTP entities with XML & JSON
* Handling responses

Pre-requisites

* Java
  + OOPS
  + Collections
  + Exception Handling

Software requirements

* Java 17 or later
* Eclipse IDE for Web & Enterprise Developers
* Tomcat 10
* Visual Studio Code

Webservices

These are online resources or services which are made available over the internet so that applications written in different technologies can exchange the data

There are two types of webservices

1. SOAP webservice = Simple Object Access Protocol
2. REST webservice = Representational State Transfer

SOAP vs REST

|  |  |
| --- | --- |
| SOAP | REST |
| It exchanges data in XML format only, it is a old webservice before REST | It can exchange the data in various formats like XML, JSON, CSV, TEXT, HTML and so on |
| It uses SOAP protocol, that will have set of rules how the XML structure should be | It doesn’t use any protocol for the structure, however it uses HTTP protocol to transfer the data |

Note: JSON is the most widely used datastructure compare to XML, because JSON is light weight and easy to parse

REST webservices

It is a webservice that can share the data between the applications in various structure like XML, JSON, Text, HTML, CSV

It uses HTTP methods to specify the webservice operations so that consumer & producer will know what operation they are doing, there are following HTTP methods

1. GET: fetch/read
2. POST: store/create
3. PUT: modify / update
4. DELETE: remove/delete

Important points while creating/accessing REST webservices

1. URL: To locate the webservcies
2. HTTP Methods: To specify what kind of operations the consumer & producer wants to do

How to create webservices in Java

Java has a standard specification to create webservices which is Jax-RS, this provides various annotations you can use to create webservices, however you must have an implementation of Jax-RS to create webservice, below are the implementations of Jax-RS

1. Jersey - Eclipse
2. RestEasy - Jboss
3. Apache CFX - Apache

JAX-RS annotations

1. @Path: To specify the webservice URL
2. @GET: To specify the HTTP method get
3. @POST, @PUT & @DELETE: to specify different http methods
4. @Produces & @Consumes: Specifies what kind of data a webservice can produce/consume

How to create our first webservice

We must create a dynamic web project and configure a server & then convert into maven project(download libraries - pom.xml)

List of libraries we need for webservices using jersey

1. jersey-server
2. jersey-container-servlet-core
3. jersey-hk2
4. jersey-media-json-binding