

Developing RESTful API using Java

IDE used:

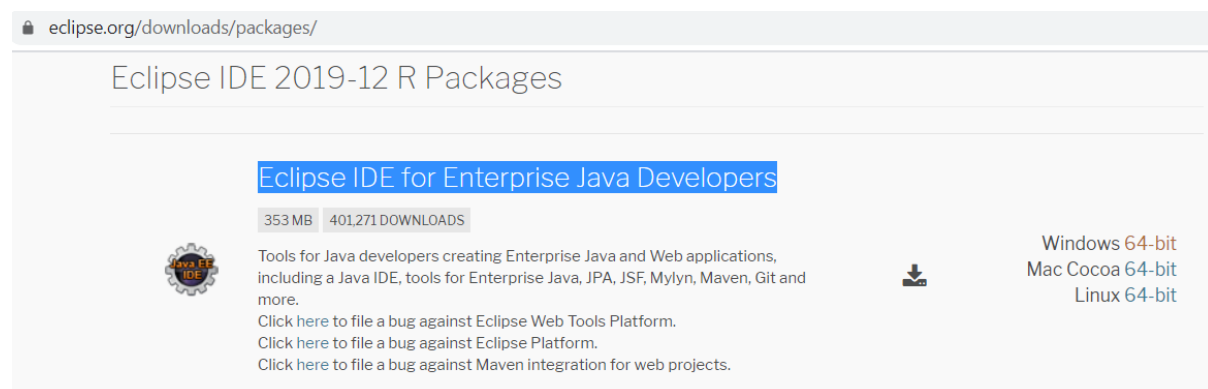
Eclipse IDE for Enterprise Java Developers.

Version: 2019-12 (4.14.0)

Download Eclipse IDE from

<https://www.eclipse.org/downloads/packages/>

The suitable package for this implementation is “Eclipse IDE for Enterprise Java Developers” :



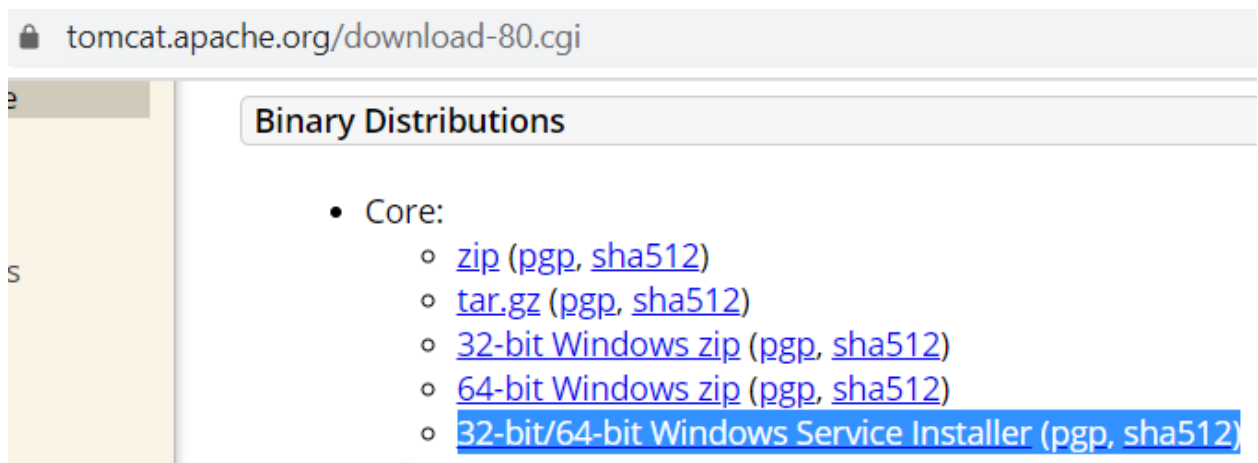
Step 1:

Install Apache Tomcat server from

<https://tomcat.apache.org/download-80.cgi>

Direct download link (for Windows):

<http://apachemirror.wuchna.com/tomcat/tomcat-8/v8.5.51/bin/apache-tomcat-8.5.51.exe>

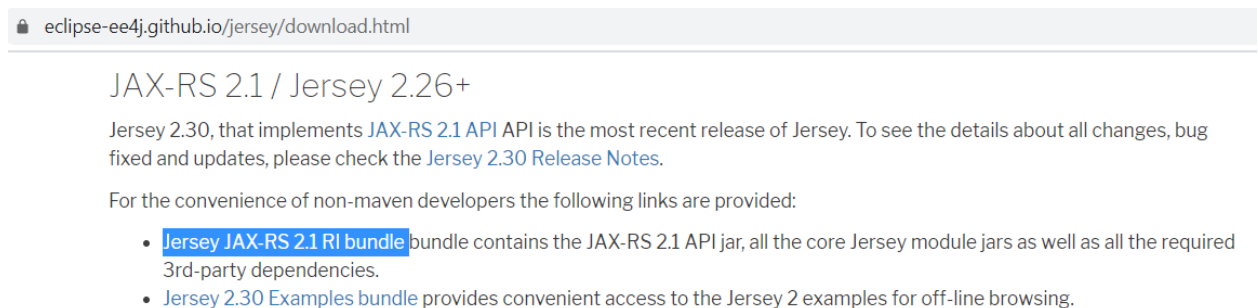


Step 2:

Download Jersey RESTful Web Services from <https://eclipse-ee4j.github.io/jersey/download.html>

Jersey JAX-RS 2.1 RI bundle direct download link:

<http://repo1.maven.org/maven2/org/glassfish/jersey/bundles/jaxrs-ri/2.30/jaxrs-ri-2.30.zip>



Alternatively, it can also be downloaded from <https://jar-download.com/artifacts/org.glassfish.jersey.bundles/jaxrs-ri/2.1/source-code>

Step 3: Open Eclipse IDE

Some useful links

Setting up tomcat : <https://www.youtube.com/watch?v=PH-bK3g2YmU>

Building RESTful API (reference):

<https://www.youtube.com/watch?v=5jQSat1cKMo>

REST/RESTful Web Services

- REST stands for Representational State Transfer. It's an architectural pattern for creating web services.
- A RESTful service is one that implements that pattern.
- REST Client applications can use HTTP GET/POST/PUT/DELETE methods to invoke Restful web services.

Java RESTful Web Services API

Java API for RESTful Web Services (JAX-RS) is the Java API for creating REST web services. JAX-RS uses annotations to simplify the development and deployment of web services. JAX-RS is part of JDK, so you don't need to include anything to use its annotations.

Restful Web Services Annotations

Some of the important JAX-RS annotations are:

- **@Path**: used to specify the relative path of class and methods. We can get the URI of a webservice by scanning the Path annotation value.
- **@GET**, **@PUT**, **@POST**, **@DELETE** and **@HEAD**: used to specify the HTTP request type for a method.
- **@Produces**, **@Consumes**: used to specify the request and response types.
- **@PathParam**: used to bind the method parameter to path value by parsing it.

REST API Implementations

There are two major implementations of JAX-RS API.

1. **Jersey**: [Jersey](#) is the reference implementation provided by Sun. For using Jersey as our JAX-RS implementation, all we need to configure its servlet in web.xml and add required dependencies. Note that JAX-RS API is part of JDK not Jersey, so we have to add its dependency jars in our application.
2. **RESTEasy**: [RESTEasy](#) is the JBoss project that provides JAX-RS implementation.

Output

(Error 404 - Page not found :O... some err in path...)

