**Jakarta RESTful Web Services**, (**JAX-RS**; formerly Java API for RESTful Web Services) is a [Jakarta EE](https://en.wikipedia.org/wiki/Jakarta_EE) [API](https://en.wikipedia.org/wiki/Application_programming_interface) specification that provides support in creating [web services](https://en.wikipedia.org/wiki/Web_service) according to the [Representational State Transfer](https://en.wikipedia.org/wiki/Representational_State_Transfer) (REST) architectural pattern.[[1]](https://en.wikipedia.org/wiki/Jakarta_RESTful_Web_Services#cite_note-1) JAX-RS uses [annotations](https://en.wikipedia.org/wiki/Java_annotation), introduced in [Java SE 5](https://en.wikipedia.org/wiki/Java_Platform,_Standard_Edition), to simplify the development and deployment of web service clients and endpoints.

From version 1.1 on, JAX-RS is an official part of [Java EE](https://en.wikipedia.org/wiki/Java_Platform,_Enterprise_Edition) 6. A notable feature of being an official part of Java EE is that no configuration is necessary to start using JAX-RS. For non-Java EE 6 environments a small entry in the web.xml [deployment descriptor](https://en.wikipedia.org/wiki/Deployment_descriptor) is required.

Specification[[edit](https://en.wikipedia.org/w/index.php?title=Jakarta_RESTful_Web_Services&action=edit&section=1)]

JAX-RS provides some annotations to aid in mapping a resource class (a [POJO](https://en.wikipedia.org/wiki/POJO)) as a web resource. The annotations use the [Java package](https://en.wikipedia.org/wiki/Java_package) javax.ws.rs. They include:

* @Path specifies the relative [path](https://en.wikipedia.org/wiki/URI_scheme#Generic_syntax) for a resource class or method.
* @GET, @PUT, @POST, @DELETE and @HEAD specify the [HTTP request type](https://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol#Request_methods) of a resource.
* @Produces specifies the response [Internet media types](https://en.wikipedia.org/wiki/Internet_media_type) (used for [content negotiation](https://en.wikipedia.org/wiki/Content_negotiation)).
* @Consumes specifies the accepted request Internet media types.

In addition, it provides further annotations to [method parameters](https://en.wikipedia.org/wiki/Parameter_(computer_programming)) to pull information out of the request. All the @\*Param annotations take a key of some form which is used to look up the value required.

* @PathParam binds the method parameter to a path segment.
* @QueryParam binds the method parameter to the value of an HTTP [query parameter](https://en.wikipedia.org/wiki/Query_string).
* @MatrixParam binds the method parameter to the value of an [HTTP matrix parameter](https://en.wikipedia.org/w/index.php?title=HTTP_matrix_parameter&action=edit&redlink=1).
* @HeaderParam binds the method parameter to an [HTTP header](https://en.wikipedia.org/wiki/List_of_HTTP_header_fields) value.
* @CookieParam binds the method parameter to a [cookie](https://en.wikipedia.org/wiki/HTTP_cookie) value.
* @FormParam binds the method parameter to a [form](https://en.wikipedia.org/wiki/Form_(HTML)) value.
* @DefaultValue specifies a [default value](https://en.wikipedia.org/wiki/Default_argument) for the above bindings when the key is not found.
* @Context returns the entire context of the object (for example @Context HttpServletRequest request).

JAX-RS 2.0[[edit](https://en.wikipedia.org/w/index.php?title=Jakarta_RESTful_Web_Services&action=edit&section=2)]

In January 2011 the JCP formed the JSR 339 expert group to work on JAX-RS 2.0. The main targets are (among others) a common client API and support for Hypermedia following the [HATEOAS](https://en.wikipedia.org/wiki/HATEOAS)-principle of [REST](https://en.wikipedia.org/wiki/REST). In May 2013, it reached the Final Release stage.[[2]](https://en.wikipedia.org/wiki/Jakarta_RESTful_Web_Services#cite_note-2)

On 2017-08-22 JAX-RS 2.1[[3]](https://en.wikipedia.org/wiki/Jakarta_RESTful_Web_Services#cite_note-3) specification final release was published. Main new supported features include [server-sent events](https://en.wikipedia.org/wiki/Server-sent_events), [reactive](https://en.wikipedia.org/wiki/Reactive_programming) clients, and JSON-B.[[4]](https://en.wikipedia.org/wiki/Jakarta_RESTful_Web_Services#cite_note-4)

Implementations[[edit](https://en.wikipedia.org/w/index.php?title=Jakarta_RESTful_Web_Services&action=edit&section=3)]

Implementations of JAX-RS include:[[5]](https://en.wikipedia.org/wiki/Jakarta_RESTful_Web_Services#cite_note-5)

* [Apache CXF](https://en.wikipedia.org/wiki/Apache_CXF), an open source [Web service](https://en.wikipedia.org/wiki/Web_service) framework
* [Jersey](https://en.wikipedia.org/wiki/Project_Jersey), the [reference implementation](https://en.wikipedia.org/wiki/Reference_implementation) from [Sun](https://en.wikipedia.org/wiki/Sun_Microsystems) (now [Oracle](https://en.wikipedia.org/wiki/Oracle_Corporation))
* [RESTeasy](http://www.jboss.org/resteasy), [JBoss](https://en.wikipedia.org/wiki/JBoss)'s implementation
* [Restlet](https://en.wikipedia.org/w/index.php?title=Restlet&action=edit&redlink=1)
* [WebSphere Application Server](https://en.wikipedia.org/wiki/WebSphere_Application_Server) from [IBM](https://en.wikipedia.org/wiki/IBM):
  + Version 7.0: via the ["Feature Pack for Communications Enabled Applications"](http://www-01.ibm.com/software/webservers/appserv/was/featurepacks/cea/index.html)
  + Version 8.0 onwards: natively
* [WebLogic Application Server](https://en.wikipedia.org/wiki/Oracle_WebLogic_Server) from [Oracle](https://en.wikipedia.org/wiki/Oracle_Corporation), see [notes](http://docs.oracle.com/cd/E24329_01/web.1211/e24963/standards.htm#g1090347)
* Apache Tuscany (<http://tuscany.apache.org/documentation-2x/sca-java-bindingrest.html>), discontinued
* Cuubez framework ([https://web.archive.org/web/20190707005602/http://cuubez.com/](https://web.archive.org/web/20190707005602/http:/cuubez.com/))
* [Everrest](https://github.com/codenvy/everrest/wiki), Codenvy's Implementation
* [Jello-Framework](http://jello-framework.com/), Java Application Framework optimized for Google App Engine, including a powerful RESTful engine and comprehensive Data Authorization model.

References[[edit](https://en.wikipedia.org/w/index.php?title=Jakarta_RESTful_Web_Services&action=edit&section=4)]

* 1. [**^**](https://en.wikipedia.org/wiki/Jakarta_RESTful_Web_Services#cite_ref-1) Hadley, p. 1.
  2. [**^**](https://en.wikipedia.org/wiki/Jakarta_RESTful_Web_Services#cite_ref-2) [*"JSR 339: JAX-RS 2.0: The Java API for RESTful Web Services"*](http://jcp.org/en/jsr/detail?id=339)*.*[*Java Community Process*](https://en.wikipedia.org/wiki/Java_Community_Process)*.*
  3. [**^**](https://en.wikipedia.org/wiki/Jakarta_RESTful_Web_Services#cite_ref-3) [*"JSR 370: Java API for RESTful Web Services (JAX-RS 2.1) Specification"*](http://jcp.org/en/jsr/detail?id=370)*.*[*Java Community Process*](https://en.wikipedia.org/wiki/Java_Community_Process)*.*
  4. [**^**](https://en.wikipedia.org/wiki/Jakarta_RESTful_Web_Services#cite_ref-4) [*"JSR 367: Java API for JSON Binding (JSON-B)"*](http://jcp.org/en/jsr/detail?id=367)*.*[*Java Community Process*](https://en.wikipedia.org/wiki/Java_Community_Process)*.*
  5. [**^**](https://en.wikipedia.org/wiki/Jakarta_RESTful_Web_Services#cite_ref-5) *Little, Mark (October 1, 2008).*[*"A Comparison of JAX-RS Implementations"*](http://www.infoq.com/news/2008/10/jaxrs-comparison)*.*
* Hadley, Marc and Paul Sandoz, eds. (September 17, 2009). [*JAX-RS: Java API for RESTful WebServices*](http://jcp.org/en/jsr/detail?id=311) (version 1.1), [Java Community Process](https://en.wikipedia.org/wiki/Java_Community_Process)

External links[[edit](https://en.wikipedia.org/w/index.php?title=Jakarta_RESTful_Web_Services&action=edit&section=5)]

* [Official website](https://projects.eclipse.org/projects/ee4j.jaxrs) [Edit this at Wikidata](https://www.wikidata.org/wiki/Q911991#P856)

**Tutorials**[[edit](https://en.wikipedia.org/w/index.php?title=Jakarta_RESTful_Web_Services&action=edit&section=6)]

* <https://javabrains.io/courses/javaee_jaxrs/>
* <http://docs.oracle.com/javaee/6/tutorial/doc/giepu.html>
* <http://www.vogella.com/tutorials/REST/article.html>
* <http://www.mkyong.com/tutorials/jax-rs-tutorials/>
* <http://www.coderpanda.com/jax-rs-tutorial/>
* <http://howtodoinjava.com/restful-web-service/>