JAVA SERVLET

|  |  |
| --- | --- |
| **Javax.servlet** | The javax.servlet package has several classes and interfaces that identifies the relationship between a servlet class and environment that has runtime that has an instance of a class by following a servlet container. |
| **Javax.servlet.annotation** | The Java Servlet annotation accommodates some annotations that can grant users to account annotations where they acknowledge the servlets, listenes, and filters. Metadata for a component which is declared is also cited. |
| **Javax.servlet.descriptor** | It administers access programmatically to a configuration of a web application that was collected from the descriptors of web.xml and web-fragment.xml descriptors. |
| **Javax.servlet.http** | The package of javax.servlet.http consists of classes and interfaces that define arrangements with a server class that runs the HTTP protocol and runtime environment that implements an instance of a class by a container servlet which complies. |

**Servlet** - These are where methods are defined, so all servlets could implement.

Servlet is a Java program that runs in a Web server. Servlets accept and acknowledges to requests from Web clients, mainly with HTTP.

**Methods**

|  |  |
| --- | --- |
| **Destroy()** | The servlet container calls to demonstrate to a servlet that it is being removed from the services. |
| **getServletConfig()** | A ServletConfig object is returned, it consists of startup parameters and initialization. |
| **getServletInfo()** | Information is returned about the servlet(e.g. author, copyright, and version) |
| **Init(ServletConfig config)** | Called by the servlet container to indicate to a servlet that the servlet is being placed into service. |
| **Service(ServletRequest req, ServletResponse res)** | The servlet container is called and allows the servlet to answer a request. |

**ServletConfig** - It is an object that uses servlet configuration. It is used by a servlet container where information is passed to a servlet when it is still being initialized.

**Methods**

|  |  |
| --- | --- |
| **getInitParameter(**java.lang.String name**)** | The value of the initialization parameter with a given name is pulled. |
| **getInitParameterNames()** | The names of the servlet’s initialization parameters are returned as EnumerationString objects, or an Enumeration that is empty, if initialization parameters are not found. |
| **getServletContext()** | A reference to the ServletContext is returned in which it is executed by the caller. |
| **getServletName()** | The name of its own servlet instance is returned. |

**ServletContainerInitializer** -

* It is an interface that allows library to be declared of the startup phase of a web application. Any needed programmatic registration of servlets, filters, and listeners are performed in response.

**Methods**

|  |  |
| --- | --- |
| **onStartup(**java.util<java.lang.Class<?>> **c**, ServletContext **ctx)** | The stated ServletContainerInitializer is notified and the startup of the application represented by the ServletContext that was given. |

Where

**C** -

-The array of application classes that allocates and integrates with the class types that was stated by the HandlesTypes annotation, or null if none match, or the ServletContainerInitializer with HandleTypes has not been annotated.

**Ctx** - the ServletContext of the web application that is being started and in which the classes contained in c were found

* ServletContext is being started in the web application and classes contained in c were located.