**Servlets:**

1. **Explain the life cycle of a Servlet.**

The web container maintains the life cycle of a servlet instance. The life cycle of a servlet includes 5 stages:

1. Servlet class is loaded.
2. Servlet instance is created.
3. ***init*** method is invoked.
4. ***service*** method is invoked.
5. ***destroy*** method is invoked.

| **1) Servlet class is loaded**  The classloader is responsible for loading the servlet class.  By default, the servlet class is loaded when the first request for the servlet is received by the web container.  This default behavious can be alterd by using <load-on-startup>n</load-on-startup> while configuring a servlet in web.xml  <servlet>  <servlet-name>Servlet-URL</servlet-name>  <servlet-class>com.javapapers.Servlet-Class</servlet-class>  <load-on-startup>2</load-on-startup>  </servlet>  load-on-startup has to be a +ve integer value. Using this option is referred to as pre-initializing the servlet.  **2) Servlet instance is created**  The web container creates the instance of a servlet after loading the servlet class.  The servlet instance is created only once in the servlet life cycle.  **3) *init* method is invoked**  The web container calls the init method only once after creating the servlet instance. The init method is used to initialize the servlet. It is the life cycle method of the javax.servlet.Servlet interface. Syntax of the init method is given below:  *public void init(ServletConfig config) throws ServletException*  **4) *service* method is invoked**  The web container calls the service method each time when request for the servlet is received. If servlet is not initialized, it follows the first three steps as described above then calls the service method. If servlet is initialized, it calls the service method. Notice that servlet is initialized only once. The syntax of the service method of the Servlet interface is given below:  *public void service(ServletRequest request, ServletResponse response)*  *throws ServletException, IOException*  **5) *destroy* method is invoked**  The web container calls the destroy method before removing the servlet instance from the service. It gives the servlet an opportunity to clean up any resource for example memory, thread etc. The syntax of the destroy method of the Servlet interface is given below:  public void destroy()  NOTE: Loading the servlet class, creating an object of servlet and the web-container calling the init() method happens only once during the life-cycle of a servlet. By default these activities happen when the first web-request is received for a servlet. But, if <load-on-startup> is used, then all these three activities happen even before the first request is received. They happen at the time of the web-container startup. |
| --- |

1. **Explain the life cycle methods of a Servlet.**

The life cycle methods of a servlet are *init, service* and *destroy.*

1. **How many objects of a servlet are created and why is it so?**

Only one object. Servlets are by default multithreaded, so single instance of a servlet may be invoked from multiple requests threads.

1. **When does the object of a servlet get instantiated?**

When the first request is received by the webserver for a given servlet.

1. **Is it possible to have the object of a servlet instantiated even before the first request is received by the web container? (How pre-initialization a servlet?)**

Yes. We must use *load-on-startup* tag in the web.xml file at the time of configuring the servlet.

1. Explain the concept of lazy-loading and eager-loading of servlets.
2. Explain the difference between ServletConfig and ServletContext
3. What is difference between PrintWriter and ServletOutputStream?
4. Explain the difference between GenericServlet and HttpServlet.
5. Explain the different methods of HttpServlet and explain their usage.
6. Explain the functionality of any 3 methods of your choice from the interface HttpServletRequest.
7. Explain the functionality of any 3 methods of your choice from the interface HttpServletResponse.
8. What is the need for Session tracking? Explain how session tracking is done with servlets.
9. What do you mean by saying that HTTP is a stateless protocol?
10. What is a cookie? How is it different from HttpSession?
11. Explain the need for Servlet Filters, how to implement filters and how to configure them in web.xml file.
12. How to upload a file on to the server using servlets?
13. Explain the annotations used in Servlet 3.

There are mainly 3 annotations used for the servlet.

1. @WebServlet : for servlet class.  *[here we can use @WebInitParam annotation]*
2. @WebListener : for listener class.
3. @WebFilter : for filter class.

https://www.wideskills.com/servlets/annotation-details-in-servlet-3