Single Threaded Servlets

- We Know that, by default servlets are Multi Threaded in Nature. Hence following are the 3 ways to create Single Threaded Servlet.
    1. By implementing "javax.servlet.SingleThreadModel" Marker Interface
    2. By Synchronizing Service Method
    3. By Synchronizing a Block of Code in Service Method (preferred)
- SingleThreadModel is a Marker Interface which ensures that servlets handle only one request at a time i.e. Container start handling the requests "Synchronously"
- This interface is "Deprecated" in Servlet API 2.4
- **Example :-**

```
public class MyServlet extends HttpServlet / GenericServlet implements
SingleThreadModel

{

        //Servlet Code Goes Here

}
```

Note: **List of Marker Interfaces**

- **Java / JDK**
    1. java.io.Serializable
    2. java.lang.Cloneable
    3. java.util.RandomAccess
    4. java.util.EventListener
    5. java.rmi.Remote
- **Servlet API**
    1. javax.servlet.SingleThreadModel

Load On Start-up

- The **load-on-startup** element of **web-app** loads the servlet at the time of deployment or server start.
- This is a **web.xml** configuration elements used to configure a servlet to create servlet object during start up time of the server or application deploy on server. It is also known as **pre initialization** of servlet. This element need integer value to configure.
- servlet is loaded at first request. That means it consumes more time at first request. If we specify the load-on-startup in web.xml, servlet will be loaded at project deployment time or server start. So, it will take less time for responding to first request.
- **What happens when we pass positive integer value to the <load-on-startup> element?**

  If we pass the positive value, the lower integer value servlet will be loaded before the higher integer value servlet. In other words, container loads the servlets in ascending integer value. The 0 value will be loaded first then 1, 2, 3 and so on.

```
<web-app>
 ....

  <servlet>
   <servlet-name>servlet1</servlet-name>
   <servlet-class>edu.jspiders.Servlet1</servlet-class>
   <load-on-startup>0</load-on-startup>
  </servlet>

  <servlet>
   <servlet-name>servlet2</servlet-name>
   <servlet-class>com.tutorial4us.Servlet2</servlet-class>
   <load-on-startup>1</load-on-startup>
  </servlet>


  <servlet>
   <servlet-name>servlet3</servlet-name>
   <servlet-class>com.tutorial4us.Servlet2</servlet-class>
   <load-on-startup>-10</load-on-startup>
  </servlet>
```

```
</web-app>
```

- Let's try to understand it by the example given above:

- There are 3 servlets defined, 2 servlets which has positive value will be loaded at the time of project deployment or server start. But, servlet1 will be loaded first then servlet2 and servlet3 will be ignored by the container.
- If a negative value is configure then a container ignores this tag and waits for first request to create an object of a servlet but a container does not throws any exception.

How to Configure load-on-startup using annotation?

```
import javax.servlet.ServletConfig;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;

@WebServlet(
        urlPatterns = "/myController",
        loadOnStartup = 1
        )
public class StartupServlet extends HttpServlet {

          public  StartupServlet()
          { System.out.println("inside constructor"); }

    public void init(ServletConfig config) {
        System.out.println("My servlet has been initialized");
    }
        // implement servlet doPost() and doGet()...
}
```