

```
public void init(ServletConfig config) throws ServletException;
```

↑

This is used internally by webcontainer

1. loading
2. instantiation
3. initialization
 

```
init(ServletConfig config)
    {
        ...
    }
```

↑ local variable
4. service
 

```
public void service(Sreq, Sresp) throws SE, IOE
    {
        ServletConfig config = getServletConfig();
        System.out.println(config); // null
    }
```
5. deInstantiation

```
public void init() throws ServletException;
```

↑

This is used by developer in his code

1. loading
2. instantiation
3. initialization
 

```
init(ServletConfig config)
    {
        ...
    }
```
4. service
5. destroy

```


RequestHeader → Internally used to give a proper response
               → User details and its supported language format and so on.....

RequestBody → "user supplied inputs will be available"
             → It contains user information

ResponseHeader → MIME(MultiPurpose Internet Mail Exchange)
               → response information from the server
                 a. content-type
                 b. content-length
                 c. modified date etc.....

ResponseBody → "Actual information"
             → Actual information from the server, this
               information is displayed to the user.

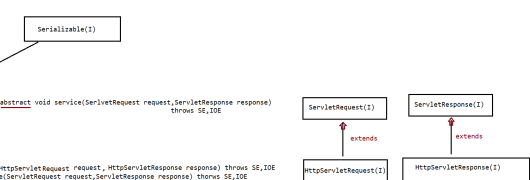
```



The diagram illustrates the structure of an HTTP Request. It consists of a box divided into three horizontal sections: **RequestLine**, **RequestHeaders**, and **RequestBody**. To the right of the box, three arrows point from the text descriptions to their respective sections:
 

- An arrow from "GET (Query String is sent)" points to the **RequestLine** section.
- An arrow from "browser details would be available" points to the **RequestHeaders** section.
- An arrow from "empty" points to the **RequestBody** section.

RequestLine	← POST
RequestHeaders	← browser details would be available
RequestBody	← (Query String is sent)



```

p;

rvJet(ac)
doPost(HSR req, HSR resp) throws SE,IOE
[MS]400 status code saying HTTP Method POST not s upported by this U
doGet(HSR req, HSR resp) throws SE,IOE
[MS]400 status code saying HTTP Method GET not s upported by this U

```

**Structure of Project**

```

resources
ProjectName
    [resources] WEB-INF
        [resources] classes
        [resources] lib *class private-area
        [resources] web.xml
    [resources] resources
    [resources] META-INF
    [resources] HTML files public area
  
```

**Annotations:**

- `http://localhost:9099/Firatapp/test`: it is configured in app for container
- `[resources] WEB-INF`: The region/files which can't be accessed by the clients from address bar(sending the request). This is solely meant for webcontainer only.
- `[resources] HTML files`: The region/files which can be accessed directly by the clients from address bar(sending the request)

