Running an Embedded Tomcat Server

Tomcat Configuration:

- **Tomcat Instance**: new Tomcat() initializes an embedded Tomcat server.
- **Persistent Server**: tomcat.getServer().await() ensures the server keeps running and listens for requests.

HTTPServlet:

service() Method: Handles incoming HTTP requests and generates appropriate responses.

```
Tomcat tomcat = new Tomcat();
tomcat.start();
```

Persistent Server

Ensure the server keeps running:
tomcat.getServer().await();

Steps:

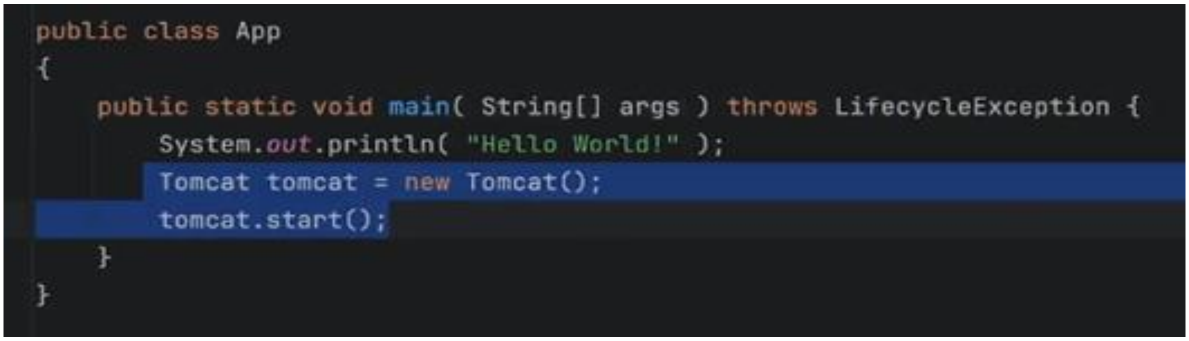
1. Create HelloServlet class
2. Making a Servlet by extending HttpServlet
3. Write the service(HttpServletRequest request, HttpServletResponse response) method.

=> For servlets, you need to send requests through a browser.

=> By default, embedded Tomcat is not running.

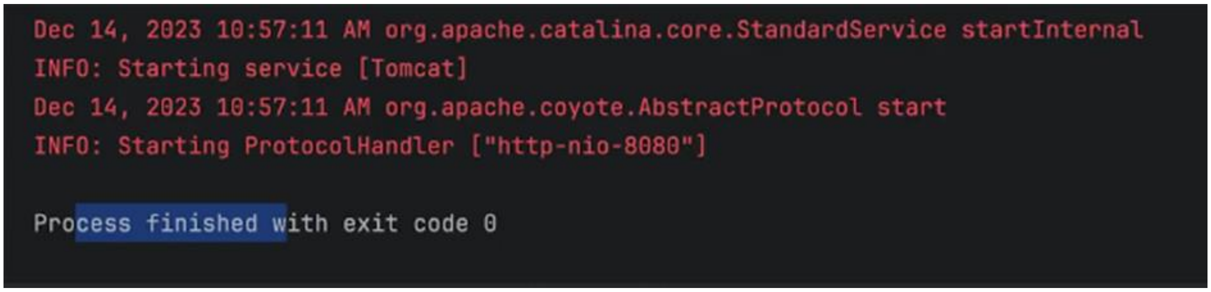
In the main method

```
Tomcat tomcat=new Tomcat();
tomcat.start();
```



```
public class App
{
    public static void main( String[] args ) throws LifecycleException {
        System.out.println( "Hello World!" );
        Tomcat tomcat = new Tomcat();
        tomcat.start();
    }
}
```

After doing that, something happens.



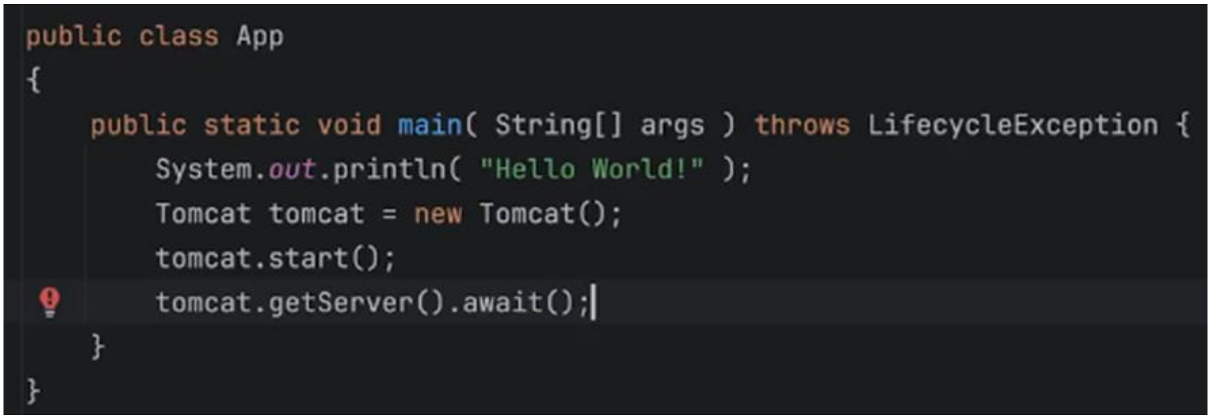
```
Dec 14, 2023 10:57:11 AM org.apache.catalina.core.StandardService startInternal
INFO: Starting service [Tomcat]
Dec 14, 2023 10:57:11 AM org.apache.coyote.AbstractProtocol start
INFO: Starting ProtocolHandler ["http-nio-8080"]

Process finished with exit code 0
```

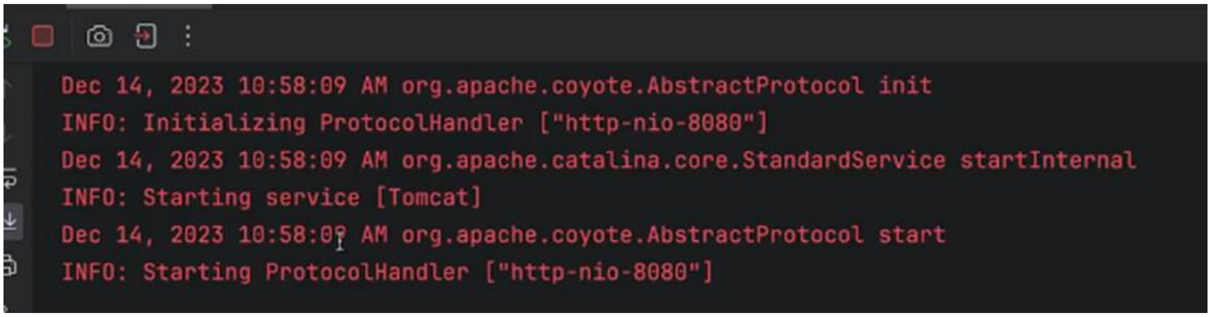
But still *localhost:8080 is* not running.

Now make Tomcat keep running.

Add tomcat.getServer().await()

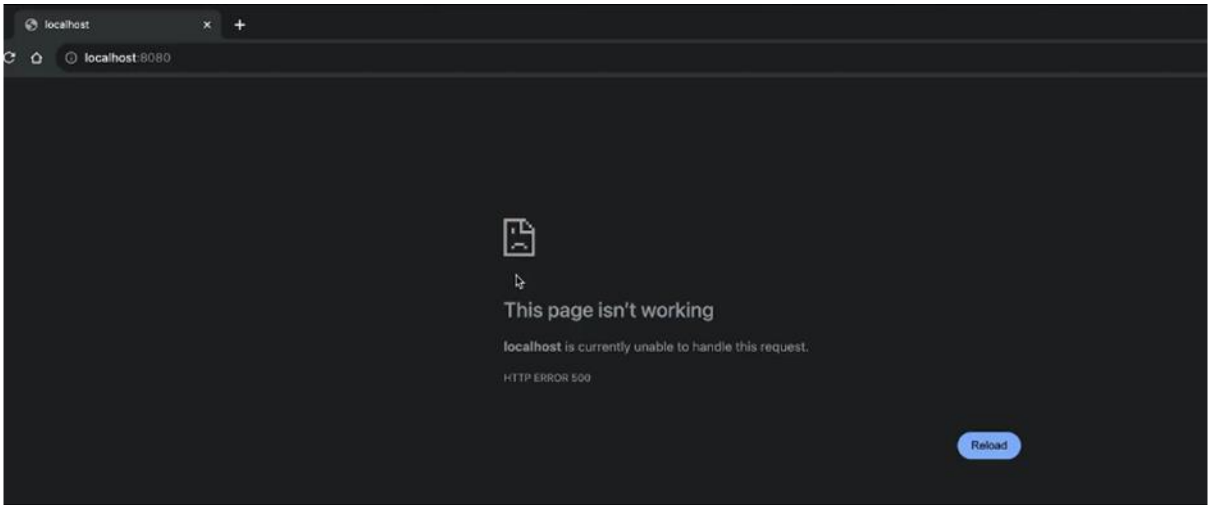


```
public class App
{
    public static void main( String[] args ) throws LifecycleException {
        System.out.println( "Hello World!" );
        Tomcat tomcat = new Tomcat();
        tomcat.start();
        tomcat.getServer().await();
    }
}
```



```
Dec 14, 2023 10:58:09 AM org.apache.coyote.AbstractProtocol init
INFO: Initializing ProtocolHandler ["http-nio-8080"]
Dec 14, 2023 10:58:09 AM org.apache.catalina.core.StandardService startInternal
INFO: Starting service [Tomcat]
Dec 14, 2023 10:58:09 AM org.apache.coyote.AbstractProtocol start
INFO: Starting ProtocolHandler ["http-nio-8080"]
```

Now we found the server does not stop.



This page isn't working

localhost is currently unable to handle this request.

HTTP ERROR 500

But get some different error message (not able to handle the request).

How does Tomcat knows which page I handle?

=> Now we do mapping, which is in the next lecture.