Q67 - Asynchronous Processing in Servlets [12]

Assume the AsyncListener is defined as follows:

com.nullhaus.MyListener

```
package com.nullhaus;

import javax.servlet.*;
```



```
public class MyListener implements AsyncListener {
    public void onComplete(AsyncEvent event) {
        System.out.println("#Async listener [onComplete]");
    }

    public void onError(AsyncEvent event) {
        System.out.println("#Async listener [onError]");
    }

    public void onStartAsync(AsyncEvent event) {
        System.out.println("#Async listener [onStartAsync]");
    }

    public void onTimeout(AsyncEvent event)  {
        System.out.println("#Async listener [onTimeout]");
    }
}
```

What will be the result of the **first** GET request to the following servlet:

```
package com.nullhaus;

import javax.servlet.*;
import javax.servlet.http.*;
import javax.servlet.annotation.*;

@WebServlet(urlPatterns = "/foo/*",
        name="NullServlet",
        asyncSupported=true)
public class NullServlet extends HttpServlet {
    public void doGet(HttpServletRequest req,
                HttpServletResponse resp)
                      throws ServletException {

        final AsyncContext ac = req.startAsync();

        try {
            Class lClass = Class.forName("com.nullhaus.MyListener");
            AsyncListener al = ac.createListener(lClass);

            ac.addListener(al);
        } catch (ClassNotFoundException e) {
            e.printStackTrace();
        }
```

```
        ac.setTimeout(3000);

        ac.start(new Thread() {
            public void run() {
                try {
                    Thread.sleep(5000);
                } catch (InterruptedException e) {
                    e.printStackTrace();
                }
                ac.complete();
            }
        });
    }
}
```

a. This code compiles

b. The "#Async listener [onTimeout]" message will be to the console/log file

c. The "#Async listener [onStartAsync]" message will be to the console/log file

d. The "#Async listener [onError]" message will be to the console/log file

e. The "#Async listener [onComplete]" message will be to the console/log file

f. A runtime exception will be thrown when accessing this servlet

g. This code doesn't compile

Hide answer

a, b, d, e, f

Reference: pages 17 - 18, 2.3.3.3 "Asynchronous processing"

Explanation: After the given timeout, the listener `onTimeout(-)` method will be invoked, as well as the `onError(-)`.

After that, the `onComplete(-)` method will be invoked (response for the `ac.complete()` invocation). This invocation will be followed by the `IllegalStateException`, as it's illegal to complete the asynchronous processing **if the request has already been dispatched (timeout)**.