

# Test Fragments in JMeter (Web-Tours Application)

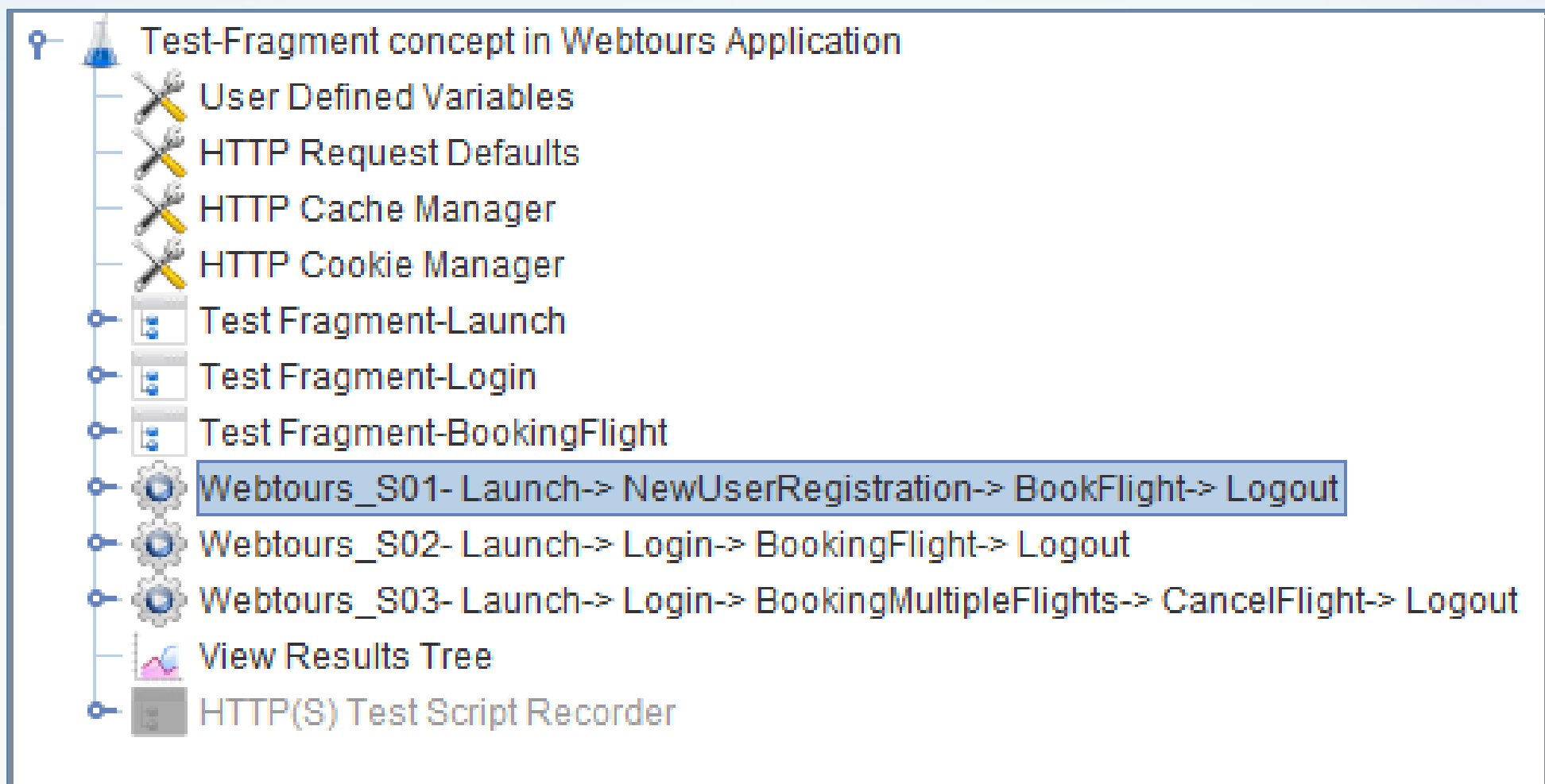
Presented by: Manisha Shinde  
Date: 27/06/2025

# Introduction to Test Fragments

- **What is a Test Fragment?**
  - A special type of JMeter element designed for modularization.
  - Not executed directly unless called via **Module Controller**.
- **Why use Test Fragments?**
  - Code reuse
  - Easier maintenance
  - Better readability
- **Overview of WebTours Application**
- A sample travel booking web application
- Key features:
  - Scenario:1- Launch > New User Registration > Book flight > Logout
  - Scenario:2- Launch > Login > Booking flight> Logout
  - Scenario:3- Launch > Login > Booking Multiple Flights > cancel flight > Logout

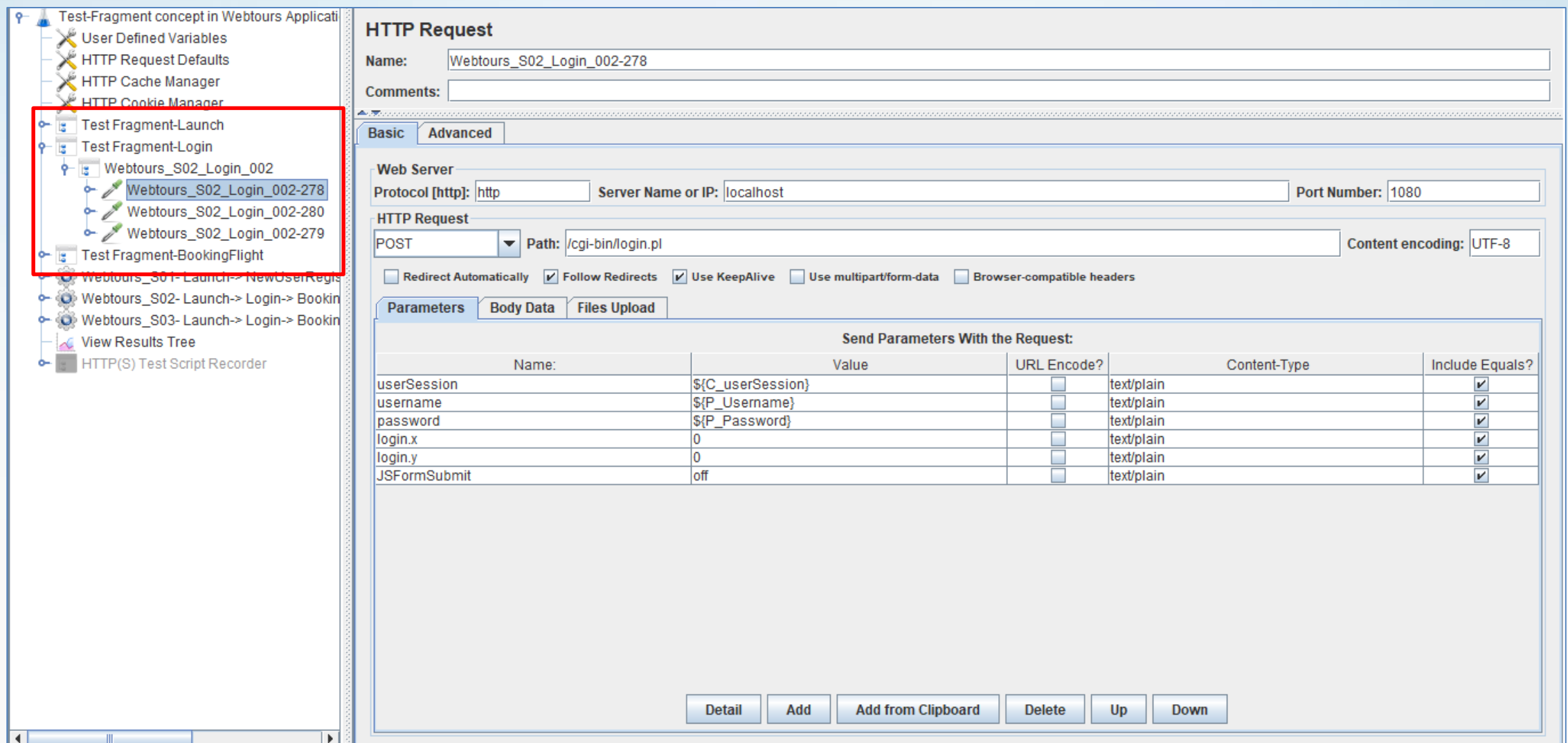
## Top-level plan contains:

- Configuration elements (e.g., HTTP Cache Manager, Cookie Manager)
- Test Fragments
- Thread Groups for different test scenarios
- View Results Tree for validation



# Step 1: Add a Test Fragment

- Right-click Test Plan → Add → Test Fragment.
- Rename: ‘Test Fragment – Launch’, ‘Test Fragment – Login’, ‘Test Fragment – Booking Flight’)
- Add HTTP Request and other elements under it.



The screenshot shows the TestNG IDE interface. On the left, the 'Test-Fragment concept in Webtours Applicati' tree is visible, with 'Test Fragment-Launch' and 'Test Fragment-Login' highlighted. The main window displays the 'HTTP Request' configuration for 'Webtours\_S02\_Login\_002-278'.

**HTTP Request Configuration:**

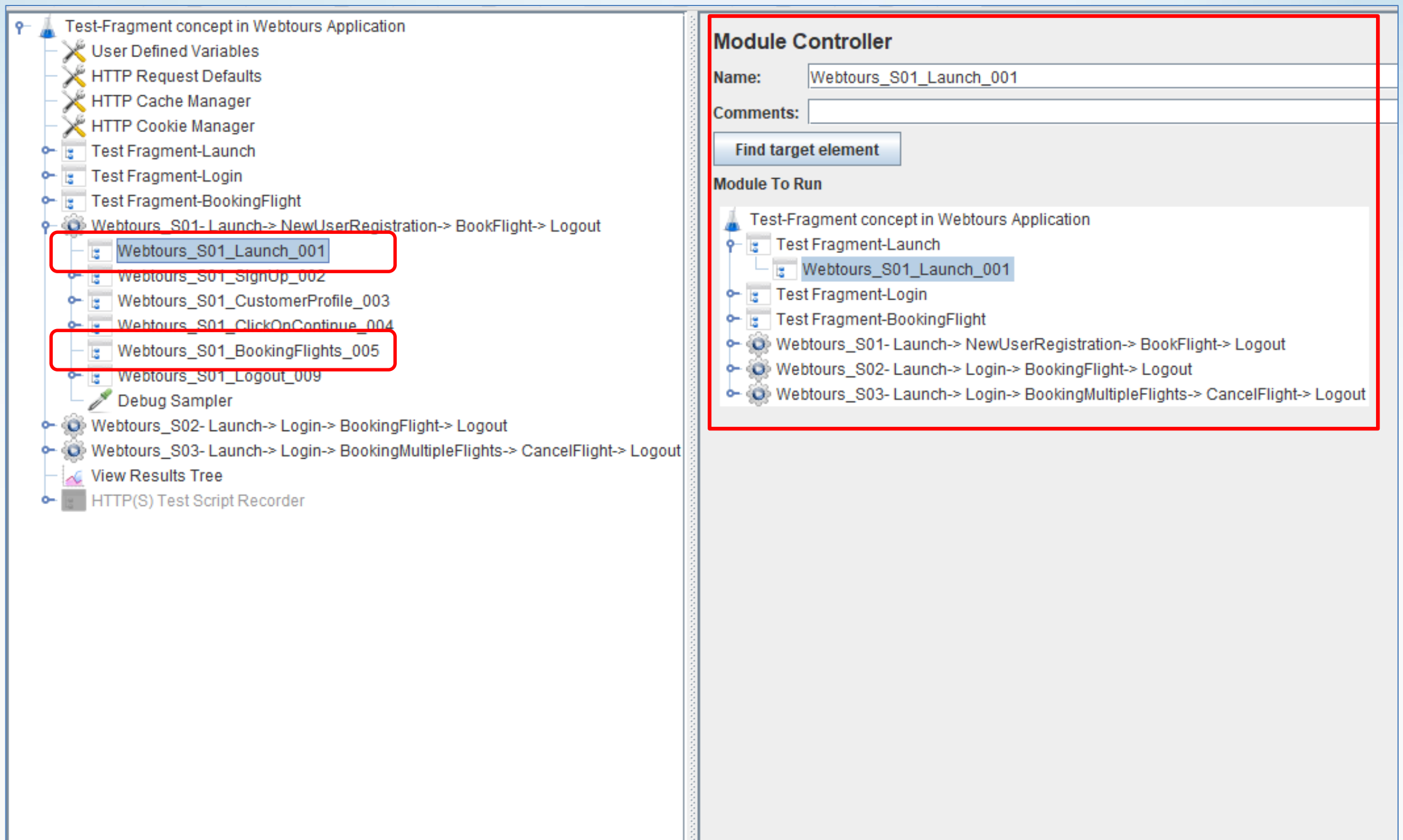
- Name:** Webtours\_S02\_Login\_002-278
- Comments:**
- Basic Tab:**
  - Web Server:** Protocol [http]: http, Server Name or IP: localhost, Port Number: 1080
  - HTTP Request:** POST, Path: /cgi-bin/login.pl, Content encoding: UTF-8
  - ☐ Redirect Automatically, ☒ Follow Redirects, ☒ Use KeepAlive, ☐ Use multipart/form-data, ☐ Browser-compatible headers
- Parameters Tab:**

Name:	Value	URL Encode?	Content-Type	Include Equals?
userSession	\${C_userSession}	<input type="checkbox"/>	text/plain	<input checked="" type="checkbox"/>
username	\${P_username}	<input type="checkbox"/>	text/plain	<input checked="" type="checkbox"/>
password	\${P_password}	<input type="checkbox"/>	text/plain	<input checked="" type="checkbox"/>
login.x	0	<input type="checkbox"/>	text/plain	<input checked="" type="checkbox"/>
login.y	0	<input type="checkbox"/>	text/plain	<input checked="" type="checkbox"/>
JSFormSubmit	off	<input type="checkbox"/>	text/plain	<input checked="" type="checkbox"/>

Buttons at the bottom: Detail, Add, Add from Clipboard, Delete, Up, Down.

## Step 2: Call Fragment with Module Controller

- Under Thread Group → Add → Logic Controller → Module Controller.
- Select corresponding Test Fragment from dropdown.
- Repeat for other fragments (e.g., Launch, BookingFlight).



The screenshot displays the Ivavsys application interface, divided into two main panels. The left panel, titled "Test-Fragment concept in Webtours Application", shows a tree view of test fragments. Two fragments are highlighted with red boxes: "Webtours\_S01\_Launch\_001" and "Webtours\_S01\_BookingFlights\_005". The right panel, titled "Module Controller", shows the configuration for a new module controller. The "Name" field is set to "Webtours\_S01\_Launch\_001". The "Comments" field is empty. A "Find target element" button is present. The "Module To Run" section shows a list of test fragments, with "Webtours\_S01\_Launch\_001" selected.

**Test-Fragment concept in Webtours Application**

- User Defined Variables
- HTTP Request Defaults
- HTTP Cache Manager
- HTTP Cookie Manager
- Test Fragment-Launch
- Test Fragment-Login
- Test Fragment-BookingFlight
- Webtours\_S01- Launch-> NewUserRegistration-> BookFlight-> Logout
  - Webtours\_S01\_Launch\_001**
  - Webtours\_S01\_SignUp\_002
  - Webtours\_S01\_CustomerProfile\_003
  - Webtours\_S01\_ClickOnContinue\_004
  - Webtours\_S01\_BookingFlights\_005**
  - Webtours\_S01\_Logout\_009
- Debug Sampler
- Webtours\_S02- Launch-> Login-> BookingFlight-> Logout
- Webtours\_S03- Launch-> Login-> BookingMultipleFlights-> CancelFlight-> Logout
- View Results Tree
- HTTP(S) Test Script Recorder

**Module Controller**

Name: Webtours\_S01\_Launch\_001

Comments:

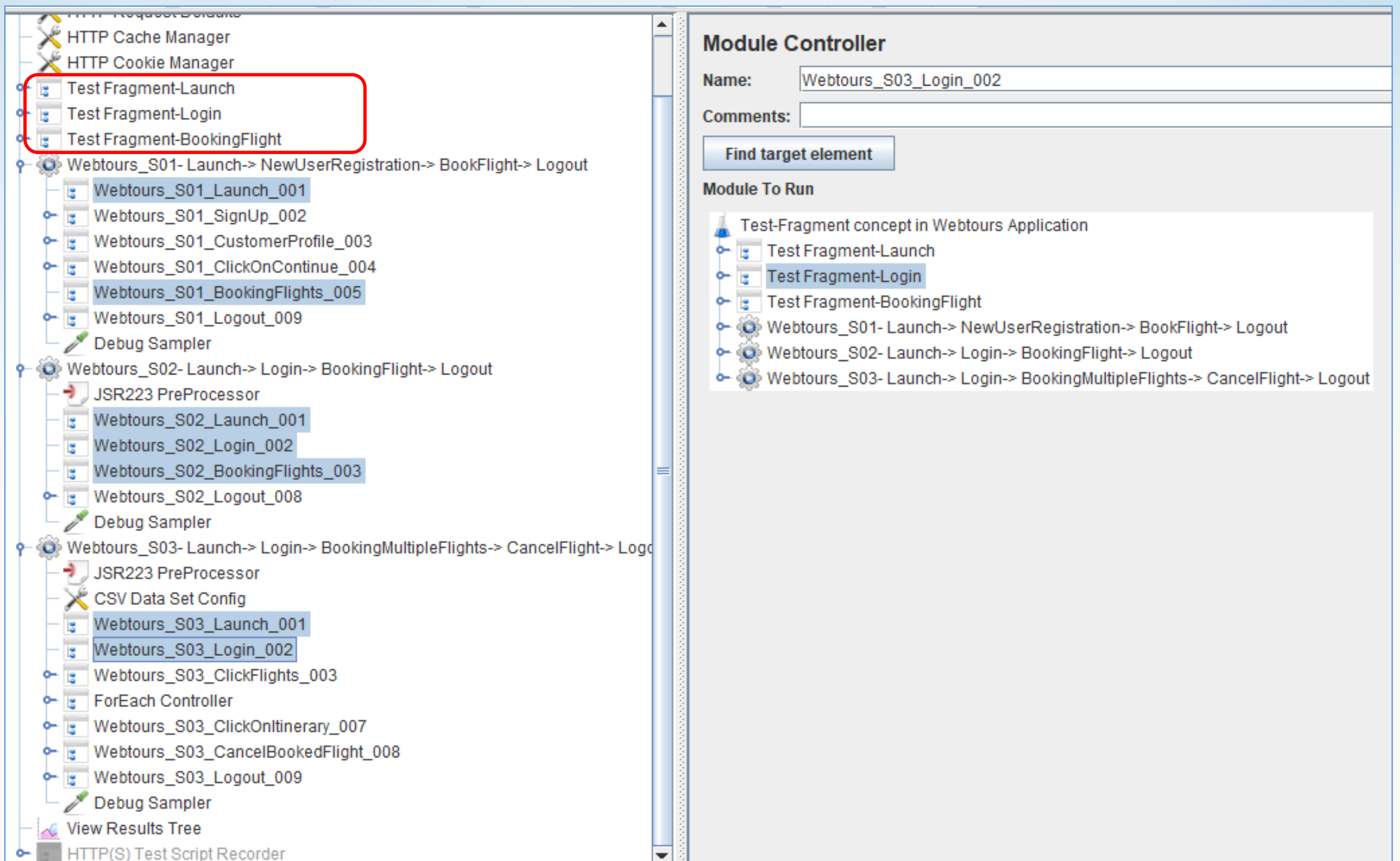
Find target element

**Module To Run**

- Test-Fragment concept in Webtours Application
  - Test Fragment-Launch
    - Webtours\_S01\_Launch\_001**
  - Test Fragment-Login
  - Test Fragment-BookingFlight
  - Webtours\_S01- Launch-> NewUserRegistration-> BookFlight-> Logout
  - Webtours\_S02- Launch-> Login-> BookingFlight-> Logout
  - Webtours\_S03- Launch-> Login-> BookingMultipleFlights-> CancelFlight-> Logout

# Step 3: Build Scenario Using Fragments

- Arrange Module Controllers in the required sequence.
- Example: Launch → Login → BookFlight → Logout.
- Use same fragments in multiple scenarios.



The screenshot displays the Ivavsys test management interface. On the left, a tree view shows the test hierarchy. The 'Test Fragment-Login' is highlighted with a red box. The right pane shows the 'Module Controller' configuration for 'Webtours\_S03\_Login\_002'. The 'Module To Run' section lists the sequence of test fragments: 'Test Fragment-Login', 'Test Fragment-BookingFlight', and 'Webtours\_S01- Launch-> NewUserRegistration-> BookFlight-> Logout'.

**Test Fragment-Login**

**Module Controller**

Name: Webtours\_S03\_Login\_002

Comments:

Find target element

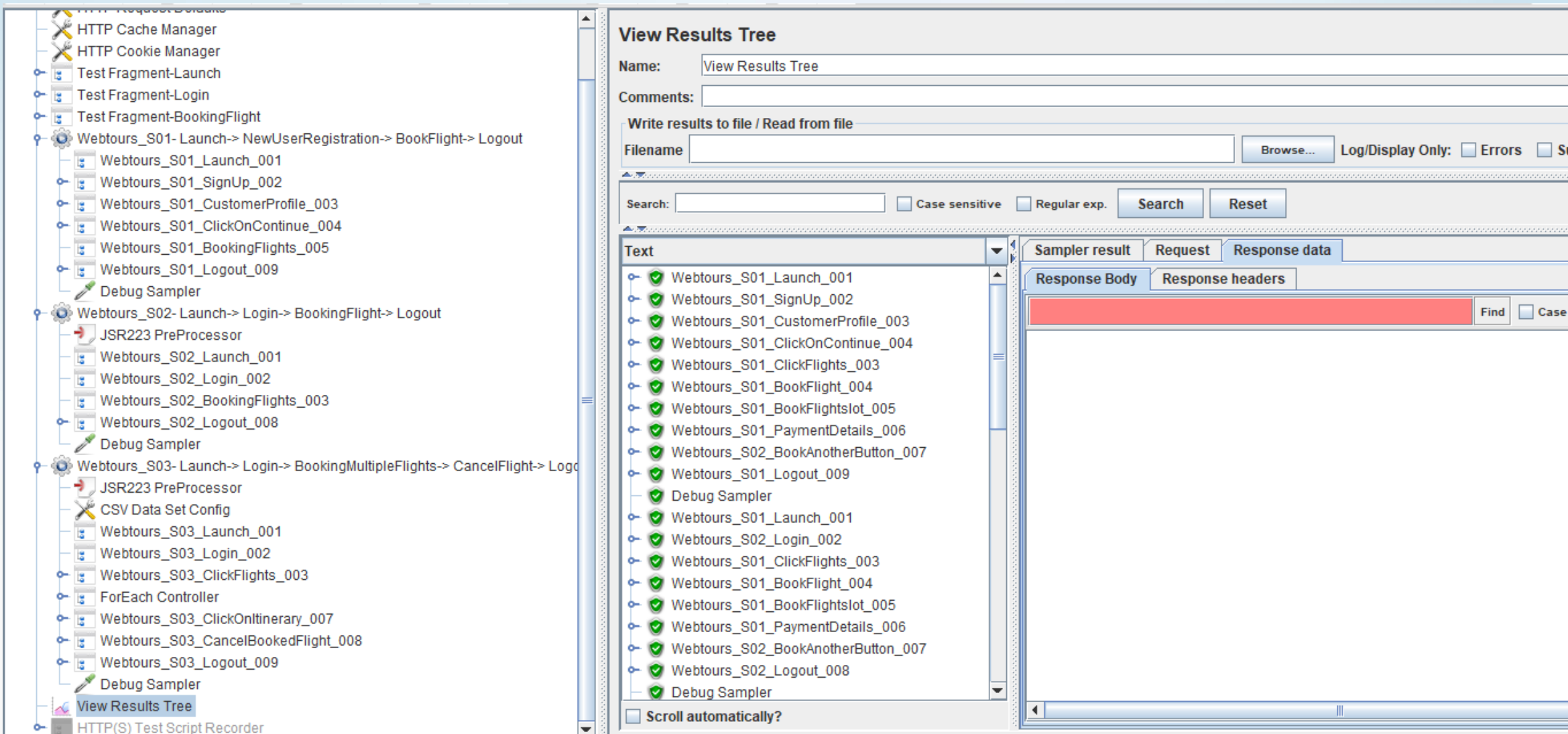
**Module To Run**

- Test-Fragment concept in Webtours Application
  - Test Fragment-Login
  - Test Fragment-BookingFlight
  - Webtours\_S01- Launch-> NewUserRegistration-> BookFlight-> Logout
  - Webtours\_S02- Launch-> Login-> BookingFlight-> Logout
  - Webtours\_S03- Launch-> Login-> BookingMultipleFlights-> CancelFlight-> Logout



# Step 4: Run and Validate

- Add Thread Group, Config Elements (Cache, Cookie, etc.).
- Use View Results Tree to validate flow.
- Ensure all fragments are being called properly.



The screenshot displays the JMeter View Results Tree window. On the left, a tree view shows the test plan structure, including three scenarios (S01, S02, S03) and their respective steps. The 'View Results Tree' element is selected. The main panel shows the 'Text' tab, listing the steps of the test plan, each with a green checkmark indicating successful execution. The 'Response data' tab is also visible, showing the 'Response Body' and 'Response headers' for the selected step.

**View Results Tree**

Name: View Results Tree

Comments:

Write results to file / Read from file

Filename:  Browse... Log/Display Only: ☐ Errors ☐ Success

Search:  ☐ Case sensitive ☐ Regular exp. Search Reset

Text

- Webtours\_S01\_Launch\_001
- Webtours\_S01\_SignUp\_002
- Webtours\_S01\_CustomerProfile\_003
- Webtours\_S01\_ClickOnContinue\_004
- Webtours\_S01\_BookingFlights\_005
- Webtours\_S01\_Logout\_009
- Debug Sampler
- Webtours\_S02\_Launch-> Login-> BookingFlight-> Logout
- JSR223 PreProcessor
- Webtours\_S02\_Launch\_001
- Webtours\_S02\_Login\_002
- Webtours\_S02\_BookingFlights\_003
- Webtours\_S02\_Logout\_008
- Debug Sampler
- Webtours\_S03- Launch-> Login-> BookingMultipleFlights-> CancelFlight-> Logout
- JSR223 PreProcessor
- CSV Data Set Config
- Webtours\_S03\_Launch\_001
- Webtours\_S03\_Login\_002
- Webtours\_S03\_ClickFlights\_003
- ForEach Controller
- Webtours\_S03\_ClickOnItinerary\_007
- Webtours\_S03\_CancelBookedFlight\_008
- Webtours\_S03\_Logout\_009
- Debug Sampler
- View Results Tree
- HTTP(S) Test Script Recorder

Sampler result Request Response data

Response Body Response headers

Find Case

Scroll automatically?