

07_assertions_and_validation.md

Assertions & Validation in JMeter

1 Why Assertions Are Mandatory

Sending requests \neq successful test.

Without assertions: ❌ 500 errors may go unnoticed ❌ Login failures look like success ❌ Invalid data passes silently

⚙️ Assertions **prove correctness** under load.

2 Types of Assertions (API Focused)

Assertion	Purpose
Response Assertion	Validate status / text
JSON Assertion	Validate JSON values
Size Assertion	Validate payload size
Duration Assertion	SLA validation
JSR223 Assertion	Custom logic

3 Response Assertion (Most Common)

Validate HTTP Status Code

Add **Response Assertion** → Apply to Main Sample

Response Code: 200

Validate Response Text

Response Text Contains: "success"

Use carefully — text may change.

4 JSON Assertion (Recommended)

Example API Response

```
{
  "status": "SUCCESS",
  "data": {
    "userId": "U123"
  }
}
```

JSON Assertion Config

JSON Path: \$.status
Expected Value: SUCCESS

✓ More stable than text matching

5 Assertions Placement (Scope)

✓ Place assertion **under the sampler** it validates ✗ Don't place at Thread Group unless intentional

Scope mistakes cause false positives.

6 Duration Assertion (SLA)

Example:

Max Response Time: 2000 ms

Use in: - Smoke tests - SLA validation

Avoid in peak load (can inflate failures).

7 Validating Login Flow

Assertions after login should validate: - Token exists - Token not empty

Example JSON Assertion:

JSON Path: \$.access_token
Expected Value: NOT EMPTY

8 Common Assertion Mistakes

✗ Asserting dynamic values ✗ Hardcoding IDs ✗ Too many assertions ✗ Validating UI text in API tests

9 Debugging Assertion Failures

Use: - View Results Tree - Assertion Result tab

Check: - Actual response - Expected value - JSON path correctness

10 Assertion Strategy (Best Practice)

Test Type	Assertion Level
Smoke	Strict
Load	Minimal
Stress	Very minimal
Soak	Health checks

11 Interview Question

Why should assertions be minimized during peak load?

Answer: They add CPU overhead and can distort performance metrics.

1 2 Mini Exercise

1. Add assertion for status code 200
2. Validate JSON success flag
3. Run with 10 users
4. Break API intentionally
5. Observe failure

Next File

 **08_timers_and_think_time.md**