10 – Listeners & Result Analysis

🎯 Objective

Learn how to capture test data and, more importantly, **how to interpret it**. Understand why "Average Response Time" is a trap and why "90th Percentile" is the industry standard.

1️⃣ What is a Listener?

A Listener is a component that gathers data from the executed samples and displays it as text, tables, or graphs.

- **Input:** Sample Results (success/fail, latency, bytes).
- **Output:** Reports (HTML, CSV, GUI Tables).

2️⃣ The "Golden Rule" of Listeners

❌ **NEVER use GUI Listeners (Graph Results, View Results Tree, Table) during a heavy load test.**

Why?
These listeners consume huge amounts of memory (RAM) to render graphics for every single request. If you run 5,000 users with "View Results Tree" enabled, JMeter will crash (Out of Memory Error) before the server does.

✅ **Best Practice:**

- Use **View Results Tree** ONLY for debugging (1-5 users).
- Use **Summary Report** or **Aggregate Report** for small local load tests.
- Use **CLI (Non-GUI) Mode** and generate HTML reports for real load tests (covered in Module 12).

3️⃣ Key Metrics Explained (The Interview Stuff)

🔹 Average (Mean)

- **Definition:** Sum of all response times / Number of requests.

- **Problem:** It hides spikes. If 99 requests take 1s and 1 request takes 100s, the average is ~2s. You think the system is fine, but one user is furious.
- **Verdict:** Good for trends, bad for SLAs.

🔹 90th Percentile (P90) ⭐⭐⭐

- **Definition:** The value below which 90% of the samples fall.
- **Example:** If P90 is **2.5s**, it means **90% of your users got a response in 2.5s or less**.
- **Why it matters:** It ignores the "lucky" fast users and focuses on the "unlucky" slow ones. This is the standard for SLAs.

🔹 95th / 99th Percentile (P95 / P99)

- Used for critical systems (Payment Gateways, Banking). We guarantee that 99% of transactions are fast.

🔹 Throughput (TPS / RPS)

- **Definition:** Transactions Per Second (or Requests Per Second).
- **Meaning:** How much load the server is handling.
- **Goal:** Higher is better (until errors start appearing).

🔹 Error %

- **Definition:** Failed requests / Total requests.
- **Goal:** Should be **0.00%** for functionality.
- **Acceptable in Load:** Maybe < 1% (depending on requirements, e.g., timeouts under extreme stress).

4️⃣ Common Listeners

📊 Summary Report

- The most efficient GUI listener.
- **Columns:** #Samples, Average, Min, Max, Std. Dev., Error %, Throughput.
- **Use for:** Quick health check during a test.

📈 Aggregate Report

- Similar to Summary Report but includes **Percentiles (90%, 95%, 99%)**.
- **Use for:** Validating SLAs locally.

🌳 View Results Tree

- Shows the exact Request and Response data (HTML/JSON).
- **Use for:** Debugging scripts. "Why did Login fail?" -> Check Response Data tab.
- ⚠️ **Turn OFF during load test.**

📡 Backend Listener (Advanced)

- Sends data to InfluxDB / Prometheus / Azure Monitor in real-time.
- **Use for:** Enterprise-grade monitoring dashboards (Grafana).

5️⃣ Analyzing a Sample Report

Imagine this result:

Label	Samples	Average	Min	Max	90% Line	Error %
Login	1000	200ms	100ms	5000ms	250ms	0.00%

Analysis:

1. **Stability:** 0% Error is good.
2. **Performance:** Average (200ms) looks great.
3. **Spikes:** Max is 5000ms (5 seconds)! Someone waited 5 seconds.
4. **Consistency:** P90 is 250ms. This means the 5s spike was rare (an outlier). 90% of people were happy.
5. **Conclusion:** The API is generally healthy, but we have occasional outliers (Cold starts? GC pauses?).

6️⃣ Interview Question

"Client requires an API response time of < 2 seconds. In your test, Average is 1.5s, but P90 is 2.8s. Did the test pass?"

Answer:
"**No, the test failed.**

SLAs are almost always based on Percentiles, not Averages. Since the 90th percentile is 2.8s, it means 10% of users (or more) are experiencing slowness above the 2s limit. We need to optimize."

🎯 Mini Exercise

1. Run your test with **50 users** (Loop 5).
2. Add an **Aggregate Report**.
3. Look at the **90% Line**.
4. Look at the **Max**.
5. Write down one sentence explaining the difference between the two for your specific run.