🛠 Performance Test Script Development

**1️⃣ Script Development Workflow**

The following steps outline the standard process for creating performance test scripts in tools like **Apache JMeter**:

1. 🗂 **Create Test Plan** – Top-level container for all test elements.
2. 🎥 **Record Script** – Capture user actions via proxy recorder/browser plugin.
3. 👥 **Add Thread Group & Load** – Define virtual users, ramp-up time, loop count.
4. 🌐 **Add Samplers** – Configure actual server requests (e.g., HTTP Requests).
5. ⚙️ **Add Config Elements** – Set variables, cookies, defaults, and data sources.
6. 📊 **Add Listeners** – View/analyze results during and after the test.
7. ⏱ **Add Timers** – Simulate think-time between requests.
8. 🔀 **Add Logic Controllers** – Control test flow (loops, conditionals, etc.).
9. ✅ **Add Assertions** – Validate server responses.
10. 🛠 **Add Pre & Post Processors** – Modify requests before sending / process responses after.
11. 🔄 **Parameterization & Correlation** – Make script dynamic & handle session data.

**2️⃣ Scenarios & Test Data**

| **🖼 Scenario** | **📜 Description** | **🔄 Parameterization** | **🔗 Correlation** |
| --- | --- | --- | --- |
| **1. Search** | Open the site, enter a keyword, view results. | Search keyword changes per user. | ❌ Not required. |
| **2. View Details** | Browse category & view product details. | Category & Product ID vary. | ✅ Required for session/product links. |
| **3. Add to Cart** | Signed-in user adds product to cart. | Username: test11 / Password: test11. | ✅ Required for product ID & session. |
| **4. Checkout** | Sign-in → Add to cart → Confirm order. | Login credentials. | ✅ Required for product/session IDs & tokens. |

**3️⃣ Key Concepts**

**🔄 Parameterization**

Replacing hardcoded values with variables for realistic load simulation.  
**Example:**  
Instead of username=test1 → use ${userName} from a CSV file.

**🔗 Correlation**

Extracting dynamic values from a server response and reusing them in later requests.  
**Example:**  
Extract **session ID** at login with a Regular Expression Extractor → reuse in next request.