Performance Test

Objective:
If you were tasked with designing a performance test for https://magento.softwaretestingboard.com/, which specific part of the website would you choose to test and why? Please describe your testing approach and the parameters you would use. You don't need to include any actual codes. Only the test scenarios would be fine.

Evaluation:
It actually depends on the need; either you want to test the single page of the website to enhance the performance, or you want to test the specific part of the website under a specific load where you expect it to work. For both procedures, you follow a structured process that includes planning, tool selection, test script creation, execution, and analysis when performing performance testing for a website.

Since Magento software is an e-commerce website, not all pages are equally critical under load. If I were tasked with designing a performance test, I'd focus on the most business-critical and high-traffic user journeys by using a hybrid approach.

Hybrid Performance Testing Approach for a Website:

To ensure both frontend speed and backend scalability, I use a two-pronged strategy:

1. Frontend Performance Testing (Lighthouse & GTmetrix)

These tools help evaluate page load speed, Core Web Vitals, and user experience metrics. I use them to:

- Optimize First Contentful Paint, LCP, and Time to Interactive
- Identify render-blocking resources, image issues, and poor caching
- Improve overall UX, SEO, and frontend performance

Ideal for identifying slow-loading elements and improving client-side efficiency.

2. Backend Load Testing (JMeter / K6 / Gatling)

To assess server-side resilience and scalability, I simulate real-world traffic and stress scenarios. I use these tools to:

- Measure performance under concurrent users.
- Test critical workflows like checkout, search, and login.
- Identify bottlenecks in APIs, databases, and infrastructure.

Ideal for validating system behaviour under load, stress, and endurance conditions.

Results:

Combining both ensures the site is:

- Fast and optimized for end-users (Frontend)
- Stable and scalable under high traffic (Backend)