**Assignment - Performance testing**

**Q1. What is performance testing?**

**Ans :** Performance Testing is a one type of testing to ensure software applications will perform well under their expected workload. Features and Functionality supported by a software system is not the only concern. A software application's performance like its response time, reliability, resource usage and scalability do matter. The goal of Performance Testing is not to find bugs but to eliminate performance. Performance Testing uncovers what needs to be improved before the product goes to market. Performance testing will determine whether or not their software meets speed, scalability and stability requirements under expected workloads,

* Speed - Determines whether the application responds quickly.
* Scalability - Determines maximum user load the software application can handle.
* Stability - Determines if the application is stable under varying loads.

**Q 2 .Types of performance testing?**

**Ans:**

* Load testing - Load Testing is to verify that the system/application can handle the expected number of transactions and to verify the system/application behaviour under both normal and peak load conditions (no. of users). The objective is to identify performance bottlenecks before the software application goes live.
* Stress testing - involves testing an application under extreme workloads to see how it handles high traffic or data processing. The objective is to identify breaking point of an application.
* Spike testing - tests the software's reaction to sudden large spikes in the load generated by users.
* Volume testing - Under Volume Testing large no. of. Data is populated in database and the overall software system's behavior is monitored. The objective is to check software application's performance under varying database volumes.
* Scalability testing - The objective of scalability testing is to determine the software application's effectiveness in "scaling up" to support an increase in user load. It helps plan capacity addition to your software system.

**Q 3 . What is Jmeter and advantages of Jmeter over performance testing tools?**

**Ans :** JMeter is a Java based desktop application that can be used for performance testing of different kinds of client-server applications like websites, web services, databases, FTP servers etc. It is an open source tool provided by Apache with no licensing cost.

Advantages of Jmeter:

1. open source - Since JMeter is open source, developers can customize its source code as per their specific requirements.
2. Record and Playback feature - JMeter provides record and playback option along with drag and drop feature which makes it easier and faster to create scripts.
3. Can load test different kinds of - It can be used for performance testing of all kinds of applications ranging from - Web applications, web services, database, shell scripts etc.
4. Platform independent - As JMeter is 100% Java based, so it is platform independent and can run in multiple platforms.
5. Supports distributed load testing - JMeter supports distributed load testing feature in which we can create master-slave setup for carrying out load test on multiple machines.
6. Free of cost - It is an open source product with zero licensing cost.

**Q 4 .What is Thread Group in Jmeter?**

**Ans:** A *Thread Group* is a set of threads executing the same scenario. Set the number of iterations in the configuration. Thread behaviour is defined according to ramp up and destroyed once the number of iterations per thread has elapsed. *Thread Group elements* are the initial steps of Jmeter test plan. A number of threads(users) can be defined in a Thread Group. Each thread simulates a real user requesting to the server under a test.

**Q 5. Write down and explain any 5 Types of Listener in Jmeter.**

## **Ans :**

## 1: Aggregate Graphs:

Aggregate graphs allow us to generate graphs easily and let us select graph display settings.

2: Aggregate Report:

Aggregate report shows a separate table row for each differently named sampler request in the test.

## 3: Assertion Results:

The Assertion Results listener displays results of all the assertions in its scope. This listener displays samples as they go and failed assertions for related samples, if there are any. Passed assertions are not shown. It is recommended to use it with functional testing or debugging purposes.

4: View Results In Table:

View Results in table creates and displays a row for every sample/request result separately. The View Result in Table listener displays information about each sample in the form of a table. The table shows time related data for each sample, the payload data, the thread number and the sample execution result. It is recommended to use it with functional testing or debugging purposes.

5: View Results Tree:

View Results Tree displays a tree consists of all the Sampler responses along with their requests. This listener displays the samples in the order they are generated by the JMeter script ,and provides parameters and data for each of them. For instance, for each sample the HTTP sampler produces, the View Results Tree listener provides the request parameters, response parameters and the response data. This is displayed under the corresponding tabs: sampler result, request, and response data.

**Q. 6. Explain how do you record your script.**

**Ans :**

1. Installed Jmeter
2. Created test plan and added one Test thread “loginlogout”.
3. Then loginlogout -> Add -> logic controller ->Recording controller
4. Test plan -> Non test elements -> Http(s) Test script recorder
5. Started recording. Then set the proxy and added two certificates in Certificate manager in settings
6. Hit the <https://accounts-dev.nfhs.org/> url and sign in with credentials and sign out after successfully sign in.
7. In Jmeter the records generated for this whole procedure.
8. Then added view Result tree type listeners in Recording controller.
9. First time when we start the execution it fails the validation as we were not putting regular expression in it.
10. In /user/signin add post processor -> Regular Expression extractor
11. Set the name, Regular Expression: token = ‘(.+?)’ template: $1$ and match no. = 1
12. Set the X-CSRF Token as per the given name of the token in Regular Expression.and Set the Authorization value of api.
13. Now when we start execution in View Result tree now the validation will be successfully executed. The regular expression extractor is used to capture the dynamic data from the response header or body, from the main sample or subsamples for future use. It allows the user to extract values from a server response. As a post-processor, this element will execute after each Sample request in its scope, applying the regular expression, extracting the requested values, generating the template string, and storing the result in the given variable name.
14. Tried for multiple number of users.
15. Save the result in .jmx and then .jtl file.

**Q 7. Why do we need to do performance testing?**

**Ans :** Performance Testing is done to provide stakeholders with information about their application regarding speed, stability and scalability. More importantly, Performance Testing uncovers what needs to be improved before the product goes to market. Without Performance Testing, software is likely to suffer from issues such as: running slow while several users use it simultaneously, inconsistencies across different operating systems and poor usability. Performance testing will determine whether or not their software meets speed, scalability and stability requirements under expected workloads. Applications sent to market with poor performance metrics due to non existent or poor performance testing are likely to gain a bad reputation and fail to meet expected sales goals. Also, mission critical applications like space launch programs or life saving medical equipments should be performance tested to ensure that they run for a long period of time without deviations.