(1) Which components have you used in Load Runner?

~ In LoadRunner, various components work together to facilitate performance testing of applications. The main components include:

~ **Load Generator** generates the load against the application by following scripts.

~ **VuGen (Virtual User Generator)** for generating and editing scripts.

~ **Controller controls,** launches and sequences instances of Load Generator - specifying which script to use, for how long etc. During runs the Controller receives real-time monitoring data and displays status.

~ **Agent process** manages connection between Controller and Load Generator instances.

~ **Analysis** assembles logs from various load generators and formats reports for visualization of run result data and monitoring data.

(2) How can you set the number of Vusers in LoadRunner?

~ You can set the number of Vusers in the controller section while creating your scenarios. Many other advanced options like ramp-up, ramp-down of Vusers are also available in the Controller section.

(3) What is Correlation?

~ Correlation in LoadRunner is a critical process that ensures the smooth execution of scripts during load testing. It handles the dynamic nature of client-server communication, making scripts more robust and reliable.

(4) What is the process for developing a Vuser Script?

~ The script development process in VUGen

~ **Record the Script:** Usually, this is the first step of scripting where every user action is recorded into a script.

~ **Replay and Verify:** Once the script is recorded, reply to the script to ensure its working right. Verify any impact through application frontend or database.

~ **Enhance the Script:** Once recording has been verified, enhance script by adding checkpoints, validating data, adding transactions and rendezvous points.

~ **Replay and Verify:** As earlier, re-play your script and verify that everything is working as

~ **Configure Runtime Settings:** Configure and control pacing duration, think time variation, proxy settings and whether you wish to ignore any external resources.

~ **Use for Load Scenarios:** Formulate load scenarios based on test objectives. Use load distribution and geo-wide agents to make real life scenarios.

(5) How does LoadRunner interact with the application?

~ LoadRunner can simulate thousands of users concurrently using application software, recording and later analyzing the performance of key components of the application. LoadRunner can generate the scripts by recording them, such as logging HTTP requests between a client web browser and an application's web server.

(6) How many VUsers are required for load testing?

~ The number of VUsers required depends on your system under test, network configurations, hardware settings, memory, operating system, software applications objective of a performance test. There can not be any generic value for Vuser.

(concurrent users) x (requests per user per minute) = total requests per minute

For example, if you run a load test with 10,000 virtual users, each making a request every 20 seconds (3 requests per minute), then you're making 30,000 requests per minute, which equals 500 requests per second.
As you can see in the calculation above, you can decrease the number of users and increase the number of requests per minute per user and still have the same requests per minute. For example, here are a few scenarios that all generate 30,000 requests per minute:

(10,000) x (3) = 30,000
(5,000) x (6) = 30,000
(1,000) x (30) = 30,000
(10) x (3,000) = 30,000

Considering that cost is often proportional to the number of concurrent virtual users, the question arises: Instead of testing with 10,000 virtual users and 3 requests per minute, can you test with fewer users and more requests per second and get the same test results? After all, you're hitting the back end with the same total number of requests per minute.

(7) What is the relationship between Response Time and Throughput?

~ The Throughput shows the amount of data in bytes that the Vusers received from the server in a second. When It is compared with transaction response time, throughput and response time get decreased.

~ The peak throughput and highest response time would occur approximately at the same time.

(8) What is Automation Testing?

~ Automation Testing or Test Automation is a software testing technique that performs using special automated testing software tools to execute a test case suite.

(9) Which Are The Browsers Supported By Selenium Ide?

~ Three components of selenium are: Selenium WebDriver, Selenium IDE, Selenium RC. Browsers supported by selenium are: Google chrome, Internet explorer 7 onwards, Safari, Opera, Firefox.

(10) What are the benefits of Automation Testing?

~ Some benefits of Automation Testing as follows:

- ~ 70% faster than the manual testing
- ~ Wider test coverage of application features
- ~ Reliable in results
- ~ Ensure Consistency
- ~ Saves Time and Cost
- ~ Improves accuracy
- ~ Human Intervention is not required while execution
- ~ Increases Efficiency
- ~ Better speed in executing tests
- ~ Reusable test scripts
- ~ Test Frequently and thoroughly
- ~ More cycle of execution can be achieved through automation
- ~ Early time to market

(11) What are the advantages of Selenium?

~ Advantages of Selenium as follow:

- ~ Open Source
- ~ Supports all browsers like IE, Firefox, Mozilla, Safari
- ~ Supports all Operating Systems.
- ~ Supports all programming languages Java, Ruby, C# and Python.
- ~ Run multiple tests at a time.

(12) Why should testers opt for Selenium and not QTP?

~ With Selenium, you can easily build a Data Driven and Keyword Driven automation framework. Selenium WebDriver runs faster than Selenium RC. It supports executing tests on headless browsers. Selenium also supports iOS and Android platforms.