Theory Questions:

BIG DATA:

* What do you consider as a Big Data?

Big Data is a combination of a huge amount of data that may be analysed computationally to reveal correlations, patterns, trends…etc. The traditionals IT applications cannot handle this amount of data easily, so use different processes and tools is needed to handle the data in an efficient way.

<https://en.wikipedia.org/wiki/Big_data>

* What are the differences between SQL and NoSQL databases?

SQL requires that you use predefined schemas to determine the structure of your data before you work with it. In addition, all of your data must follow the same structure. This can require significant up-front preparation.

A No SQL database, on the other hand, has dynamic schema for unstructured data, and data is stored in many ways: it can be column-oriented, document-oriented, graph-based or organized as a KeyValue store. This flexibility means that:

* You can create documents without having to first define their structure
* Each document can have its own unique structure
* The syntax can vary from database to database, and
* You can add fields as you go.

<http://www.thewindowsclub.com/difference-sql-nosql-comparision>

* What is Hadoop?

Is an open source framework which support distributed applications. This framework allow you to work with thousand of nodes and a huge amount of data. This open source project is based on Java technology.

<https://www.bernardmarr.com/default.asp?contentID=1080>

* What is parquet file?

Is a columnar storage file format in Big Data environments which provides efficiency in the data management (data compression, encoding…).

PERFORMANCE TESTING:

In order to reach a good performance testing execution, is very important plan the tests base on the Non Functional Requirements provided (NFR). The NFR will define the business process to automatize, the scenarios and the type of tests execute.

* Load Testing: This test will be executed with the load expected in Production env. The objective is to identify performance bottlenecks in the environment with an expected load.
* Endurance Testing: It is similar to the test before but in a long period of time. Usually we will execute this test during a period of time similar to a day work of the users which are going to use the application under test.
* Volume Testing: The objective is to check software application’s performance under varying database volumes. The performance of an application can vary a lot depending of the amount of data included in the database, so we should add an amount of data similar that we will have in Prod environment.
* Capacity Testing: This test will be conducted in conjunction with capacity planning, which you use to plan for future growth, so we will planed this tests base on the expectation in future usage levels of the application.
* Stress testing: The objective is to identify breaking and critical points of an application. In order to create a good stress tests, is important increase gradually the number of concurrent users or transactions per second, in order so see where is the break and critical point of the system.

In order to ensure that tests can be re-run in future, is very important to record the most important requests in the tests, parametrize properly and use good regular expressions in the scripts.

Important metrics (KPIs) to check from the performance tool:

* Average response time
* Transactions per second
* Hits per second
* % of errors.
* Throughput
* Concurrent threads…

From my point of view, is very important to store the results in a management testing tool, in order to see the progress of the performance of the application. (testlink, HP ALM …etc).

The status of the test will be determined by the comparison between the results and KPI’s.