**Basic Interview Questions**

**1. Compare NoSQL & RDBMS**

|  |  |  |
| --- | --- | --- |
| **Criteria** | **NoSQL** | **RDBMS** |
| Data format | Does not follow any order | Organized and structured |
| Scalability | Very Good | Average |
| Querying | Limited as no Join Clause | Using SQL |
| Storage mechanism | Key-Value Pair, document, column storage, etc. | Data & relationship stored in different tables |

**2. What is NoSQL?**

NoSQL encompasses a wide variety of different database technologies that were developed in response to a rise in the volume of data stored about users, objects and products. The frequency in which this data is accessed, and performance and processing needs. Relational databases, on the other hand, were not designed to cope with the scale and agility challenges that face modern applications, nor were they built to take advantage of the cheap storage and processing power available today.

**3. What are the features of NoSQL?**

When compared to relational databases, NoSQL databases are more scalable and provide superior performance, and their data model addresses several issues that the relational model is not designed to address:

* Large volumes of structured, semi-structured, and unstructured data
* Agile sprints, quick iteration, and frequent code pushes
* Object-oriented programming that is easy to use and flexible
* Efficient, scale-out architecture instead of expensive, monolithic architecture

### ****4. Explain the difference between NoSQL v/s Relational database?****

The history seem to look like this:  
Google needs a storage layer for their inverted search index. They figure a traditional RDBMS is not going to cut it. So they implement a NoSQL data store, BigTable on top of their GFS file system. The major part is that thousands of cheap commodity hardware machines provides the speed and the redundancy.Everyone else realizes what Google just did.Brewers CAP theorem is proven. All RDBMS systems of use are CA systems. People begin playing with CP and AP systems as well. K/V stores are vastly simpler, so they are the primary vehicle for the research.

Software-as-a-service systems in general do not provide an SQL-like store. Hence, people get more interested in the NoSQL type stores.I think much of the take-off can be related to this history. Scaling Google took some new ideas at Google and everyone else follows suit because this is the only solution they know to the scaling problem right now. Hence, you are willing to rework everything around the distributed database idea of Google because it is the only way to scale beyond a certain size. Get to know more about this NoSQL vs. SQL – What is Better? that can help you grow in your career.

**If you want to kick-start your journey in databases check out our**[***Database Courses***](https://intellipaat.com/course-cat/database/)**.**

### ****5. Explain “Polyglot Persistence” in NoSQL?****

In 2006, Neal Ford coined the term polyglot programming, to express the idea that applications should be written in a mix of languages to take advantage of the fact that different languages are suitable for tackling different problems.

Complex applications combine different types of problems, so picking the right language for each job may be more productive than trying to fit all aspects into a single language.Similarly, when working on an e-commerce business problem, using a data store for the shopping cart which is highly available and can scale is important, but the same data store cannot help you find products bought by the customers’ friends—which is a totally different question. We use the term polyglot persistence to define this hybrid approach to persistence.

These are described in NoSQL’s online reference guide and in [NoSQL community.](https://intellipaat.com/community/)

## Intermediate Interview Questions

### ****6. How does NoSQL DB budget memory?****

The Replication Node manages the data in a NoSQL DB store and is the main consumer of memory. The Java heap and cache size used by the Replication Node can be important performance factors. By default, the Replication Node heap and cache are calculated by NoSQL DB based on the amount of memory available to the Storage Node.

We recommend that you specify the available memory for a Storage Node using the -memory\_mb flag for makebootconfig, or the memory\_mb Storage Node parameter. If you do not define memory\_mb, it will default to the memory available on the node. NoSQL DB will then use 85% of memory\_mb as the heap for the Replication Node processes hosted by that Storage Node. If the Storage Node hosts more than one Replication Node, the memory will be divided evenly between all RNs.

If the number of Replication Nodes on a Storage Node changes, the per-RN memory will be recalculated dynamically. The percentage used for heap is controlled by the rnHeapPercent Storage Node parameter. You can choose to override the default value of 85%.Each Replication Node uses a cache, and the size of that cache defaults to 70% of the Replication Node heap. You can override the 70% default by setting the rnCachePercent Replication Node parameter.

The Replication Node heap can also be specified directly by setting the -Xmx in the Replication Node javaMiscParams parameter. Likewise, the Replication Node cache can be set directly with the cache Size Replication Node parameter. While that’s possible, it’s advisable to use the Storage Node memory\_mb setting.

As an example, suppose you specify that a Storage Node may use 3000 MB of memory, by setting memory\_mb to 3000. If that Storage Node hosts two Replication Nodes, the heap for each RN will be (3000 \* .85)/2 = 1275MB. Each RN cache will be (1275 \* .70) = 892MB.

### ****7. How to script NoSQL DB configuration?****

You may find that you want to build the same NoSQL DB configuration repeatedly for testing purposes. The Admin CLI commands can be scripted in several ways.Many uses of the Admin CLI are simple commands, such as java -jar kvstore.jar makebootconfig to initially configure a StorageNode, shown above.

These are as amenable to scripting as any other UNIX commands and will not be discussed further here.The interactive commands available in java -jar kvstore.jar runadmin, among which are those used to create and execute plans, can be scripted in two ways. You can create a file containing the sequence of commands that you want to run, and run them in a batch using java -jar kvstore.jar runadmin load -file <script>.For example, a script file named deploy.kvs could contain commands such as the following:

configure -name mystore

plan deploy-datacenter -name boston -rf 3 -wait

plan deploy-sn -dcname boston -host localhost -port 5000 -wait

plan deploy-admin -sn sn1 -port 5001 -wait

You could execute this script by issuing the command

java -jar kvstore.jar runadmin -host localhost -port 5000 load -file

deploy.kvs

Another way to script commands is to run individual CLI commands as separate shell command lines. Trailing arguments in this command line are considered to be a single CLI command.

This usage mode lets you use features of a more capable scripting language, such as a UNIX shell, and provides more flexibility for integrating NoSQL DB commands with other commands that are not available in the the interactive java -jar kvstore.jar runadmin environment.

The same sequence of commands as those from the example above could be couched in a shell script this way:

#!/bin/sh

HOST=localhost

PORT=5000

HTTPPORT=5001

KVADMIN=”java -jar lib/kvstore.jar runadmin -host $HOST -port $PORT”

# Each CLI command that follows “$KVADMIN” is executed in a new invocation of

runadmin

$KVADMIN configure -name mystore

$KVADMIN plan deploy-datacenter -name boston -rf 3 -wait

$KVADMIN plan deploy-sn -dcname boston -host localhost -port $PORT -wait

$KVADMIN plan deploy-admin -sn sn1 -port $HTTPPORT –wait

### ****8. Does NoSQL Database Interact With Oracle Database?****

NoSQL Database supports retrieving records through the Oracle Database External Table functions. This makes it possible to perform some queries from Oracle Database and retrieve records from NoSQL Database.

### ****9. What is the difference between NoSQL & Mysql DBs’?****

NoSQL databases are becoming a major part of the database landscape today, and with their handful of advantages, they can be a real game changer in the enterprise arena. However, NoSQL isn’t ripe yet, and professionals in the industry need to approach it with caution.

This is because it lacks the maturity that SQL databases like MySQL offer. If your application doesn’t fall into the category of the likes of Google, Yahoo, Facebook or Wikipedia, you should reconsider your options for using NoSQL and stick with MySQL instead. Not only is there a major skills gap with finding NoSQL professionals, but issues like analytics, performance reporting and migration also need to be considered.

Learn more about NoSQL Database read this [*NoSQL Tutorial*](https://intellipaat.com/blog/nosql-database-tutorial/).

### ****10. Explain Oracle NoSQL database?****

The Oracle NoSQL Database is a distributed key-value database. It is designed to provide highly reliable, scalable and available data storage across a configurable set of systems that function as storage nodes.Data is stored as key-value pairs, which are written to particular storage node(s), based on the hashed value of the primary key. Storage nodes are replicated to ensure high availability, rapid failover in the event of  a node failure and optimal load balancing of queries. Customer applications are written using an easy-to-use Java/C API to read and write data.  
Oracle NoSQL Driver links with the customer application, providing access to the data via appropriate storage node for the requested key.  A web based console as well as command line interface is available for easy administration of the cluster.

### ****11. When should I use a NoSQL database instead of a relational database?****

A relational database enforces ACID. So, you will have schema based transaction oriented data stores. It’s proven and suitable for 99% of the real world applications. You can practically do anything with relational databases.But, there are limitations on speed and scaling when it comes to massive high availability data stores.

For example, Google and Amazon have terabytes of data stored in big data centers. Querying and inserting is not per formant in these scenarios because of the blocking/schema/transaction nature of the RDBMs. That’s the reason they have implemented their own databases (actually, key-value stores) for massive performance gain and scalability.NoSQL databases have been around for a long time – just the term is new. Some examples are graph, object, column, XML and document databases.

Interested in learning NoSQL? Click here to learn more in this [*NoSQL Training*](https://intellipaat.com/nosql-cassandra-hbase-training/)!