**Elastic Search** => Highly scalable open-source full-text search and analytics engine. It allows you to store, search, and analyze big volumes of data quickly and in near real time.

Elastic search is a RESTful distributed search engine built on top of Apache Lucene and released under an Apache license. It is Java-based and can search and index document files in diverse formats.

### Elastic search is a database that stores, retrieves, and manages document-oriented and semi-structured data.

**Lucene**-> Full featured information retrieval library.

Elastic search is standing as a NOSQL DB because:

* It’s easy-to-use.
* Has a great community.
* Compatibility with JSON.
* Broad use cases.

**Backend components**

## Node(Database)

## Cluster

## Index(Table)

## Document(Row)

## Shard and Replicas -> Ability to subdivide your index into multiple pieces called shards.

# The Elastic stack:-

# ElasticSearch :- Elastic search is a highly scalable open-source full-text search and analytics engine. The transformed data from Log stash is Store, Search, and indexed.

# LogStash :- Log stash is an open source, server-side data processing pipeline that ingests data from a multitude of sources simultaneously, transforms it, and then sends it to collect.

# Kibana*:-* Lets you visualize your Elastic search data and navigate the Elastic Stack.

# 

# *Question :-*

# 1. What is the storage model elastic search is following?

# For example Oracle is using relational model, Alfresco is using "document model" and Apache Jackrabbit is using "hierarchical model".

# 2. Log data stored in elastic search is persistent/ permanent? Orv Elastic Search deletes log data after certain period?

# 3. How we will manage/backup this data?

# 4. Log/data files in Elastic Search is human-readable?

# *Answers :-*

# 1. The storage model is a Document model. Everything is a document. The documents are of a particular type and they are stored in an index.

# 2. Data send to ES is stored on disk. It can be then read, searched or deleted through a REST API.

# 3. The Data is managed through the rest API. Usually for log centralisation, the logs are stored in date-based index

# (one index for today, one for yesterday and so on), so to delete the logs from one day, you delete the relevant index. Curator can help in this case. ES offers a backup and restore module.

# 4. To access the data in ES, you'll have to use the REST API or use the Kibana client.

# Elassandra:-

# First of all, Cassandra is able to handle large amounts of inserts without much problem and at high speeds. This is possible thanks to Cassandra’s storage engine, that stores data sequentially on disk for each table, but of course, nothing comes for free. The price paid to make them possible, comes when you issue a read. Reads are slower, because the engine has to sweep the disk to look for the most recent version of the object that it is looking for. To sum this up, Cassandra will perform much better in environments with more inserts than reads.

# *Elasticsearch provides us with a wide spectrum of queries and aggregations to apply on the data you hold in your indexes. The whole point of using Elasticsearch, is being able to do queries on your data. “Bool”, “Full text” and “Range”, are some of the most commonly used queries in Elasticsearch, and of course, the aggregations.*