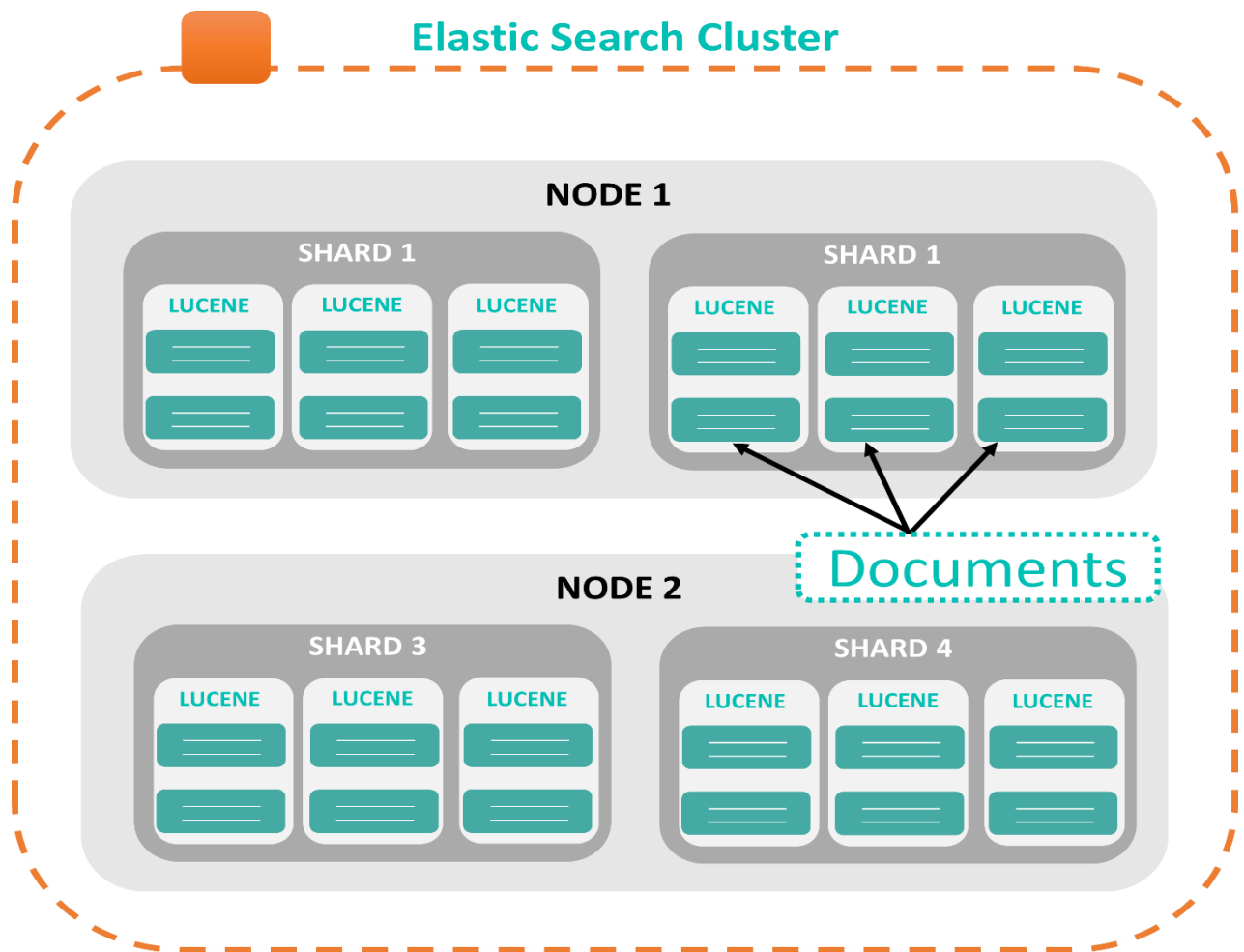# Q.) Elastic Search Architecture?

Elastic Search is comprised of several different parts, mentioned below:-

Elastic Search Cluster

NODE 1

SHARD 1

LUCENE   LUCENE   LUCENE

SHARD 1

LUCENE   LUCENE   LUCENE

Documents

NODE 2

SHARD 3

LUCENE   LUCENE   LUCENE

SHARD 4

LUCENE   LUCENE   LUCENE

1. **Node**: A node is a single instance of an elastic search running on a machine.
   - On one CPU or machine, you run (1, N) nodes or elastic search instances. But running only one node per machine is recommended for better performance.
2. **Document**: A document is a JSON Object. As shown below.
   -
     ```json
     {
       "id": 12345,
       "name": "iPhone 13 Pro",
       "brand": "Apple",
       "category": "smartphone",
       "price": 999,
       "description": "The latest and greatest iPhone"
     }
     ```
3. **Index**: An index is a collection of documents that share similar characteristics. It is similar to a table in a traditional relational database management system.

   For example, let's say you have an e-commerce website that sells books, and you want to index all the book details such as the book title, author, publisher, publication date, price, and description. You can create an index named "books" in Elasticsearch and store all the book details in that index.

   Each document within the "books" index would represent a single book and contain all the information related to that book. The document would have fields for book title, author, publisher, publication date, price, and description, among other attributes.

BOOK INDEX (Elastic Search)

Document 1

```json
{
  "title": "The Hobbit",
  "author": "J.R.R. Tolkien",
  "price": 10.99,
  "publication_date": "1937-09-21",
  "description": "A hobbit goes on an epic..."
}
```

Document 2

```json
{
  "title": "To Kill a Mockingbird",
  "author": "Harper Lee",
  "price": 8.99,
  "publication_date": "1960-07-11",
  "description": "A young girl growing up in..."
}
```

BOOK TABLE (MySQL)

Title	Author	Price	Publication_date	Description
The Hobbit	J.R.R. Tolkien	10.99	1937-09-21	A hobbit goes on an epic...
To Kill a Mockingbird	Harper Lee	8.99	1960-07-11	A young girl growing up in...

4. Shard: A shard is a smaller subset of an index. A shard is a smaller subset of a larger index that stores and indexes a portion of the data. An index can be split into multiple shards to distribute the data across multiple nodes in a cluster.

   For example, let's say we have an index called "products" with 10 million product documents, and we want to split the index into four shards. Elasticsearch will divide the index into four smaller pieces, each containing 2.5 million product documents. Each shard will be stored on a different node in the cluster.

5. Lucene Index: A Lucene index is a mapping of words and the corresponding documents containing the word. Lucene index is a java-library that is used to create inverted indexes inside a shard. Each shard will have its own Lucene index. Inverted indexes in Elasticsearch are a data structure that allows for fast full-text searches. They are used to map words or terms to the documents that contain them.

   so let's see an example. Suppose that we have two recipes with the following titles: "The Best Pasta Recipe with Pesto" and "Delicious Pasta Carbonara Recipe." The following table shows what the inverted index would look like:-

Term	Document #1	Document #2
best	X	
carbonara		X
delicious		X
pasta	X	X
pesto	X	
recipe	X	X
the	X	
with	X	

6. Elastic Search Cluster: A group of one or more nodes is called a cluster. Each node in the cluster can store and process data and communicate with other nodes in the cluster to perform searches and queries.

   For example, let's say you are a website owner who wants to provide a powerful search capability for your users. You decide to use Elasticsearch to index and search your website's content. You set up an Elasticsearch cluster with three nodes, each running on a separate server.