

- Cryptocurrency mining and blockchain operations
 Correct!
 - Elasticsearch is a distributed search and analytics engine designed for search and retrieval operations. It excels at finding relevant documents from large datasets using full-text search, filtering, sorting, and aggregations.

Question 2 of 15	
What is the fundamental relation	nship between Elasticsearch
and Apache Lucene?	

- 1 Lucene is built on top of Elasticsearch
- 2 Elasticsearch is a competitor to Lucene
- 3 They are completely independent systems
- 4 Elasticsearch is a high-level orchestration framework built on top of Lucene

Correct!

Elasticsearch serves as a distributed orchestration layer that handles cluster coordination, APIs, and real-time capabilities, while Apache Lucene provides the core search functionality and indexing engine underneath.

Question 3 of 15	P
In Elasticsearch's data model, what is the related documents, indices, and mappings?	ionship between
1 They are all equivalent concepts	
2 Documents contain indices which contain mappin	gs



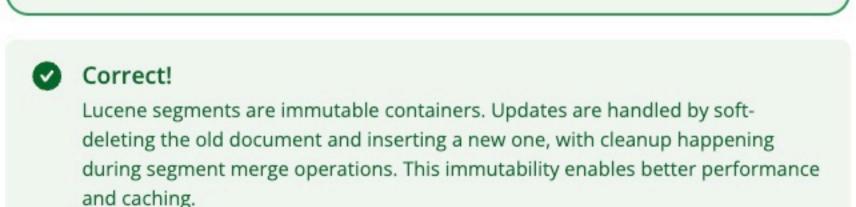
Mappings contain indices which contain documents

Indices contain documents, and mappings define the schema for those documents

Correct! An index is a collection of documents (similar to a database table), while a mapping defines the schema - specifying field types and how they should be processed and indexed for search.

upa	ated w	nen doc	uments	are mod	инеа.	
1	True					

False



Question 5 of 15
What is the primary purpose of an inverted index in Elasticsearch?

To map terms/words to the documents that contain them for fast

To store documents in reverse chronological order

- 2 To handle document updates

keyword searches

- 4 To compress document storage
- Correct!

An inverted index maps each unique word to the documents containing it, turning an O(n) scan through all documents into an O(1) lookup for keyword searches. This is the core data structure that makes text search fast.





aggregations

Metadata about documents

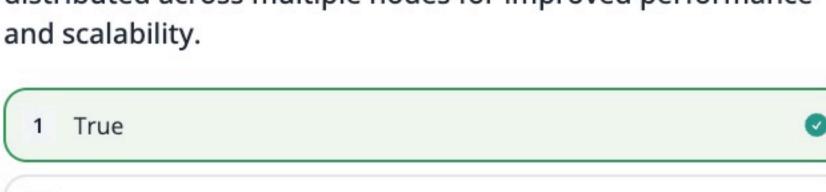
4 Document version numbers

Connect

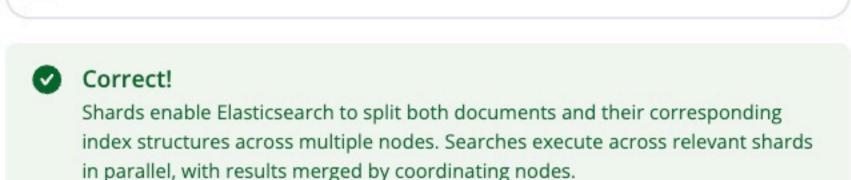
3

Correct!

Doc values provide columnar, contiguous storage of field values across all documents in a segment. This enables efficient sorting and aggregations by allowing access to specific field data without reading entire documents.



False



Question 8 of 15 How do replica shards improve Elasticsearch cluster performance?

- They enable both high availability and increased search
 - throughput by load balancing queries

They only provide data backup for high availability

- They reduce storage costs
- They speed up write operations
- Correct!

3

Replica shards serve dual purposes: they provide high availability if primary shards fail, and they increase search throughput by allowing the coordinating node to distribute search requests across all available shard copies (primary and replicas).

Que	stion 9 of 15	P
	y might deep pagination using from/size become inefficie lasticsearch?	nt
1	It requires too many disk writes	
2	The cluster must retrieve and sort all preceding documents on	

- 3 It uses too much network bandwidth
- 4 It conflicts with the inverted index

✓ Correct!

each request

Deep pagination with from/size requires the cluster to retrieve, sort, and skip over all documents up to the desired offset on every request. For large offsets (like page 1000), this becomes prohibitively expensive.

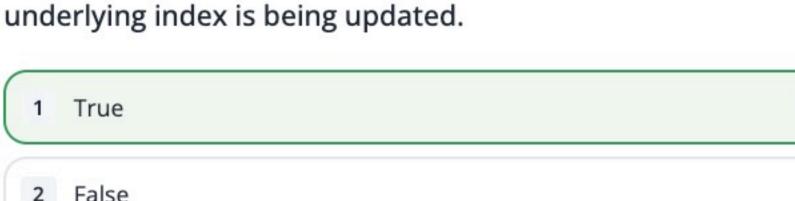
It only fetches documents after the last result from the previous

2 It provides better security

page, making deep pagination efficient

It allows random access to any page

- 4 It uses less memory for small result sets
- Correct! Search_after uses sort values from the last document of the previous page to progressively restrict the search set, avoiding the need to retrieve and sort all preceding documents. This makes deep pagination much more efficient.



Correct!

PIT cursors create a consistent snapshot of the index state that remains stable throughout the pagination process, preventing issues like missing or duplicate

results when documents are added/removed during pagination.

How does Elasticsearch's query planner optimize search performance?

- By always using the inverted index first
- 2 By caching all search results
- 3 By distributing queries randomly across nodes
- By keeping statistics on fields and keywords to choose efficient execution strategies

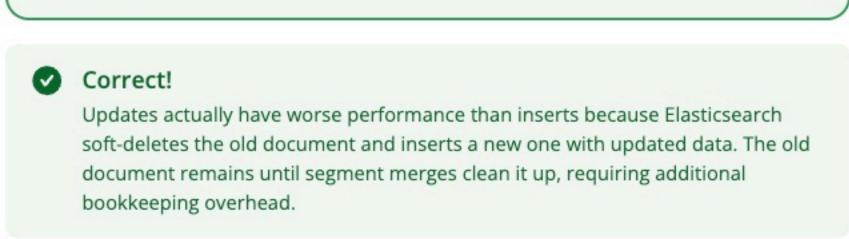


Question 12 of 15

The query planner uses statistics about field types, keyword popularity, and document characteristics to determine the most efficient execution strategy, such as deciding the optimal order for processing query terms or whether to use indexes or direct document searches.

Question 13 of 15	H
Elasticsearch update operations have insert operations because they mod	
1 True	

False



Question 14 of 15
Which approach is generally recommended when using

Elasticsearch in system design interviews?

- Use Elasticsearch alongside an authoritative data store via
- Change Data Capture (CDC)
 - 2 Use Elasticsearch as your primary database for all data

Replace all SQL databases with Elasticsearch

- 4 Use Elasticsearch only for write-heavy workloads
- Correct!

Elasticsearch is best used as a specialized search engine alongside an authoritative data store like Postgres or DynamoDB, with CDC keeping them synchronized. This provides search capabilities while maintaining data durability and consistency guarantees.

Question 15 of 15
What is a key limitation to consider when using Elasticsearch
for frequently updated data?

- Updates corrupt the inverted index
 - 2 It only supports batch updates
 - 3 It cannot handle any updates
 - 4 Updates require soft deletion and re-insertion, creating performance overhead until segment merges occur

Correct!

Quarties 15 of 15

Elasticsearch handles updates by soft-deleting the old document and inserting new data, leaving deleted documents in segments until merge operations clean them up. This creates performance penalties for workloads with frequent updates, making it less suitable for rapidly changing data.