# 

# Updating Documents in Elasticsearch

## 1. Introduction

In this section, we will explore how to update documents in an Elasticsearch index. Updates can include modifying existing fields or adding new fields to a document. While it appears as though the document is updated, Elasticsearch replaces the document internally due to its immutability.

## 2. Updating Existing Fields

1. To update a document, use the Update API by sending a POST request to the document's endpoint.

2. The endpoint format is as follows:

- /<index\_name>/\_update/<document\_id>

3. In the request body, include a 'doc' object with key-value pairs for the fields to be updated.

4. Example: Decreasing the 'in\_stock' field value for a product with ID 100.

POST /products/\_update/100  
{  
 "doc": {  
 "in\_stock": 3  
 }  
}

5. Response Details:

- 'result': Indicates the result of the operation. Possible values:  
 - 'updated': The document was successfully updated.  
 - 'noop': No changes were made because the supplied value matched the existing value.

6. To confirm the update, retrieve the document and check the '\_source' key.

## 3. Adding New Fields

1. To add a new field, include it in the 'doc' object of the same Update API query.

2. Example: Adding a 'tags' field with an array of string values.

POST /products/\_update/100  
{  
 "doc": {  
 "tags": ["sale", "electronics"]  
 }  
}

3. The new field will be added to the document if it does not already exist.

4. To verify, retrieve the document and check the '\_source' key for the new field.

## 4. How Updates Work Internally

1. Elasticsearch documents are immutable, meaning they cannot be modified directly.

2. When an update is performed:  
 - The existing document is retrieved.  
 - The specified changes are applied.  
 - The modified document is reindexed with the same ID, effectively replacing the original document.

3. The Update API simplifies this process, reducing the number of network requests required.

4. Advantages of the Update API:

- Efficiency: Updates occur within a single request on the shard where the document is stored.  
 - Reduced Latency: No need for multiple network roundtrips.

5. Without the Update API, you would need to:  
 - Retrieve the document.  
 - Modify its fields.  
 - Replace the document by sending a new request.

## 5. Summary

1. Documents in Elasticsearch can be updated using the Update API by sending a POST request to the document's endpoint.

2. Updates can modify existing fields or add new fields.

3. Internally, documents are replaced during updates due to their immutability, but the Update API abstracts this process.

4. The Update API enhances efficiency and reduces latency by performing the entire update operation in a single request.