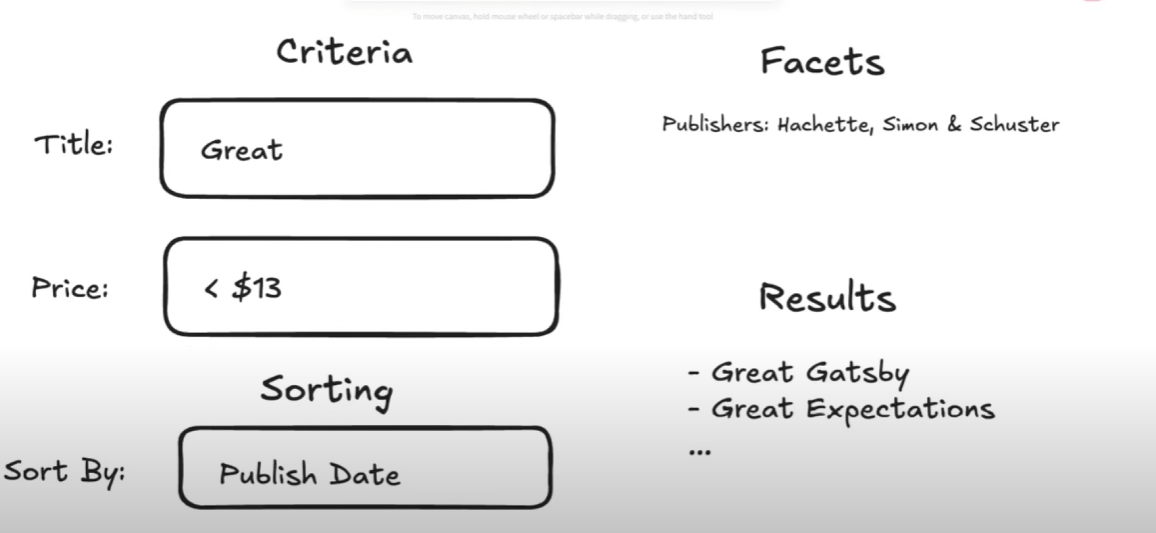
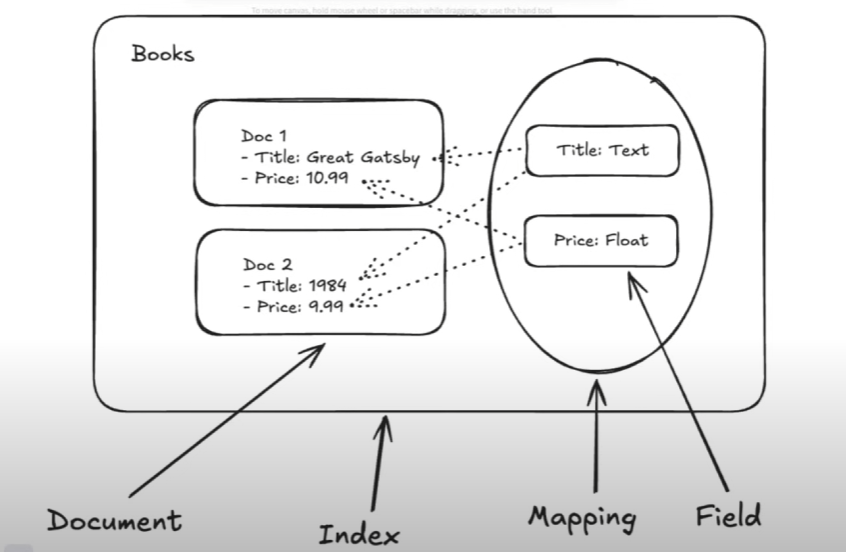
# 1. How to use elastic search

https://www.youtube.com/watch?v=PuZvF2EyfBM

****

A basic search experience is combined of a few elements shown above. And how does elastic search think?



There a few key concepts:

* Document -> JSON Blobs that can potentially contain anything
* Index -> How we structure data to make retrieval really fast, in this case, it’s a collection of documents we want to make indexable, in which we have two concepts
* Mapping -> The schema I want to be searchable on my documents, telling the search engine which fields are searchable and which are their types
* Field



Elastic Search provides a Restful API

* The first one creates a **books** index, with 1 shard and 1 replica
* I don’t have to specify a mapping, but I can, because if ES cannot infer what fields the mapping has I can cause issues.
* **If a subfield changes a lot like reviews, it’s actually more useful to have a separate index for them**

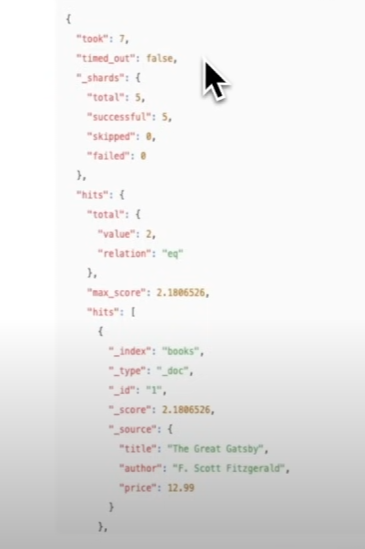


This is an example of how to ingest a document, ES is going to respond with a success message, with some useful info like id and version number (Useful for handling concurrency)



The top example is a rejected request since the version number does not match, this gives a chance to handle the collision/concurrency.





This is the Search API of ES, which is quite simple.

**TF-IDF** -> Term Frequency – Inverse Document Frequency -> Query “Elasticsearch Guide”   
And I get 2 docs (1000 words) and (5000 words), which one would be better to return? A score boost Is gotten if the word frequency is higher

