Databases
MongoDB

Qinshift
Academy

AIMS AND GOALS OF THIS COURSE

- Understanding the advanced concepts of Node.js
- Learning about ORMs & DBs
- Learning how to write type safe code with Typescript
- Getting familiar with a BE framework – Nest.js and its consisting parts

WHAT ARE DATABASES?

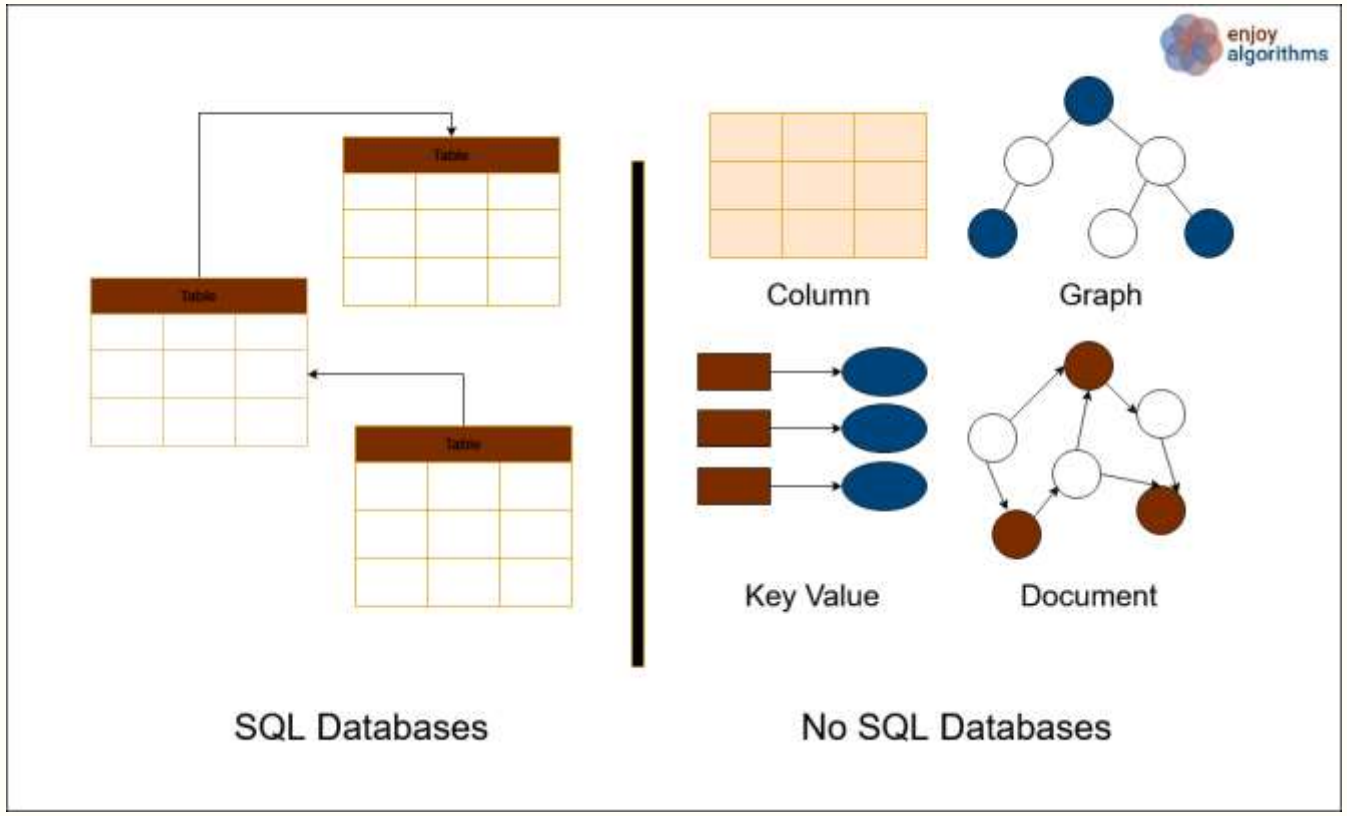
- A **database** is an organized collection of data, so that it can be easily accessed and managed.
- The **main purpose** of the database is to operate a large amount of information by storing, retrieving, and managing data.
- Databases are generally accessed electronically from a computer system and are usually controlled by a **database management system** (DBMS).

WHAT ARE DATABASES?

- There are many databases available like:
  - MySQL
  - Oracle
  - MongoDB
  - PostgreSQL
  - SQL Server, etc.
- **SQL** or Structured Query Language is used to operate on the data stored in a database.

TYPES OF DATABASES

Databases are widely divided into two major types or categories, namely, **Relational** or Sequence Databases and **Non-relational** or Non-sequence databases or No SQL databases.



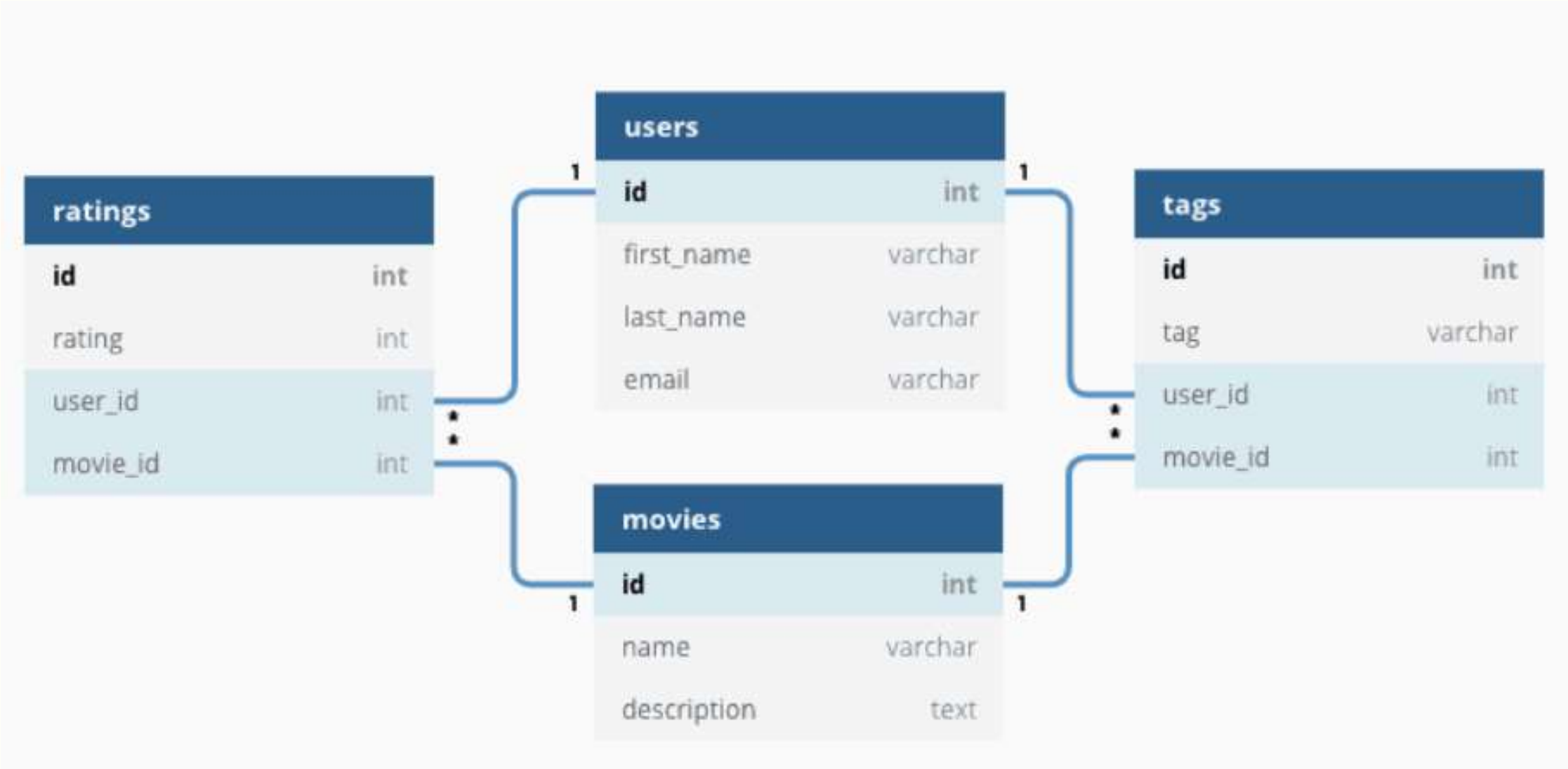
RELATIONAL DATABASE

A relational database is the most common type of database. It uses **schema**, which is a template used to dictate the data structure stored within the database.

Tables, which are also called entities, are all **in relation to each other**.

There are **keys** associated with tables in a relational database. They provide a quick database summary or access to the particular row or column you want to check.

RELATIONAL DATABASE



Benefits of using SQL databases

Relational databases follow a strict schema, meaning that each new entry must have different components that make it fit in that preformed template. It enables the data to be predictable and easily assessable.

They are well structured and significantly reduce the chances of errors.

NON-RELATIONAL DATABASE

Another common type of database is non-relational. The non-relational form of database organization is more forgiving in its structure and form than relational databases. Instead of tables with columns and rows, they have collections of different categories illustrated by documents. So, there can be multiple documents in one collection. Also, they may or may not follow any particular pattern or schema.

NON-RELATIONAL DATABASE

Key	Document

1001	<pre>{
	  "CustomerID": 99,
	  "OrderItems": [
	  { "ProductID": 2010,
	  "Quantity": 2,
	  "Cost": 520
	  },
	  { "ProductID": 4365,
	  "Quantity": 1,
	  "Cost": 18
	  }],
	  "OrderDate": "04/01/2017"
	}</pre>

1002	<pre>{
	  "CustomerID": 220,
	  "OrderItems": [
	  { "ProductID": 1285,
	  "Quantity": 1,
	  "Cost": 120
	  }],
	  "OrderDate": "05/08/2017"
	}</pre>

NON-RELATIONAL DATABASE

The different types of non-relational databases are:

- Key-Value Stores - only stores and provides quick and simple knowledge regarding key-value pairs (Redis / Amazon DynamoDB)
- Wide Column Stores - stores and manages humongous amounts of data in tables or multiple columns (CassandraDB)
- Document Stores - Data gets stored in JSON documents, and these documents resemble those of key-value and wide-column. Some of the most famous NoSQL databases fall into this category, namely, Couchbase and MongoDB
- Graph Databases - show the connections between different data points. They are used to analyze different types of data and their relationship with each other.

Mongo DB

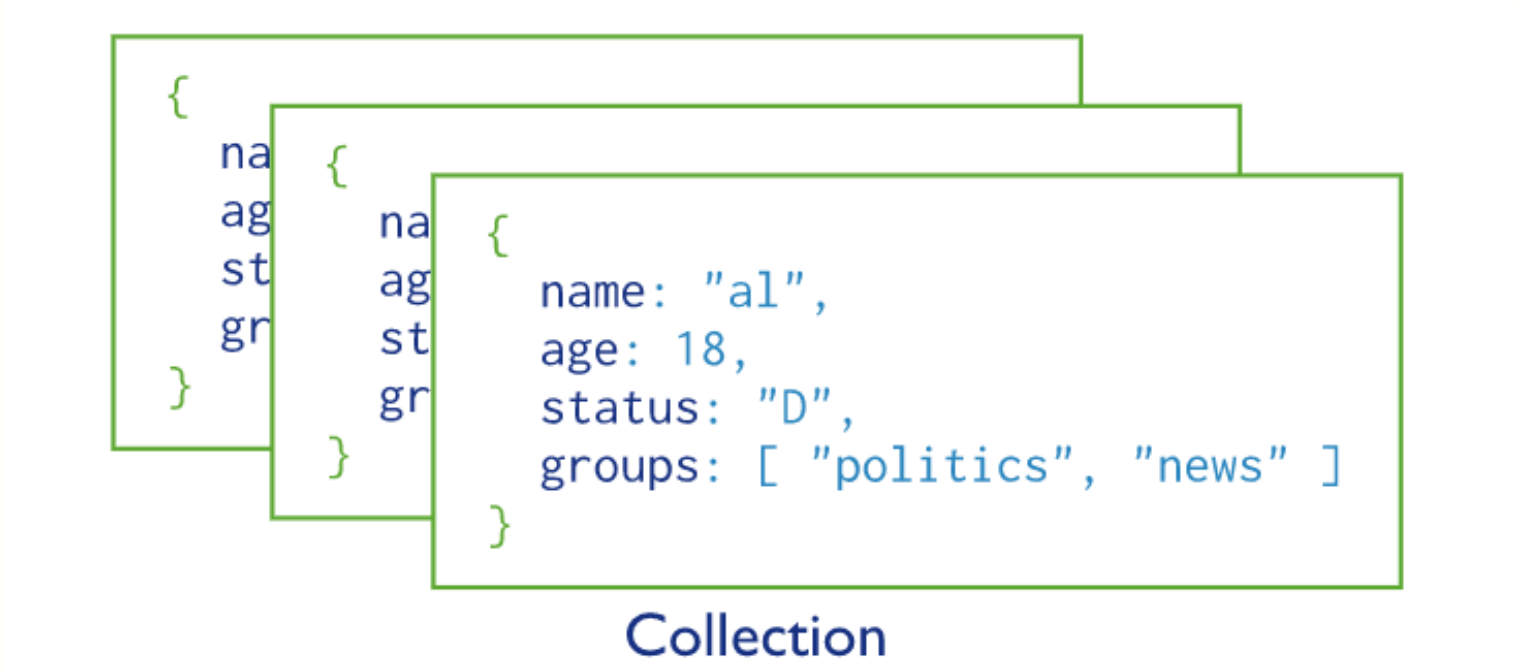
Mongo DB

MongoDB is a source-available cross-platform document-oriented database program. Classified as a NoSQL database program, MongoDB uses JSON-like documents with optional schemas.

HOW MONGO STORES DATA?

MongoDB stores data records as documents (specifically BSON documents) which are gathered together in collections.

A database stores one or more collections of documents. Collections are analogous to tables in relational databases.



DOCUMENTS

MongoDB documents are composed of field-and-value pairs and have the following structure:

DOCUMENTS

The value of a field can be any of the BSON data types, including other documents, arrays, and arrays of documents. For example, the following document contains values of varying types:

```
{
    _id: ObjectId("5099803df3f4948bd2f98391"),
    name: { first: "Alan", last: "Turing" },
    birth: new Date('Jun 23, 1912'),
    death: new Date('Jun 07, 1954'),
    contribs: [ "Turing machine", "Turing test", "Turingery" ],
    views : NumberLong(1250000)
}
```

DOCUMENTS

Documents have the following restrictions on field names:

- The field name _id is reserved for use as a primary key; its value must be unique in the collection, is immutable, and may be of any type other than an array. If the _id contains subfields, the subfield names cannot begin with a ($) symbol.
- Field names cannot contain the null character.
- The server permits storage of field names that contain dots (.) and dollar signs ($).
- MongodB 5.0 adds improved support for the use of ($) and (.) in field names. There are some restrictions.

Questions?

Trainer Name

Trainer

trainer@mail.com

Assistant Name

Assistant

asistant@mail.com

Qinshift Academy