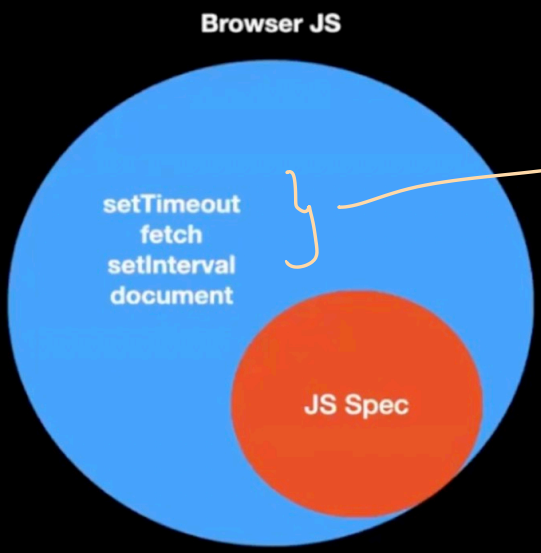
Week-4

→ DOM intro :-

The DOM (Document Object Model) API is a programming interface for web documents. It represents the page so that programs can change the document structure, style, and content. The DOM represents the document as a tree of objects; each object represents a part of the page.

Js Code → Js Engine → 010101



→ Not Part of JS
Web APIS

⇒ React :-

It helps to make DOM easy.
Virtual DOM → only update new data.
npm create vite@latest → 5173 port
main.jsx → root
app.jsx ⇒
Jsx → JavaScript + html

⇒ MongoDB :-

user → http → DB
- Browser don't support talking with DB

⇒ Database Access is 1 way. You will either get full Access or no Access. No way of getting half-restricted access.

Firebase let http server and try their best to provide granola Access.

⇒ Any DB let You do → C R U D — delete.

create Read update

⇒ In mongoose, first you have to define the schema This sounds counter intuitive since mongodb is schemaless?
That is true, but mongoose makes you define schema for things like autocompletions/ Validating data before it goes in the DB to make sure you're doing things right Schemaless Dbs can be very dangerous, using schemas in mongo makes it slightly less
Dangerous

⇒ SQL is not friendly for storing complex objects.

Ex: id: 123
name: abc
course: ["A", "B"] ← This is advantage of noSQL DB

⇒ SQL for course selling

user
id, usrID, pass, name

purchased - course
userID, courseID

course
id, title, price

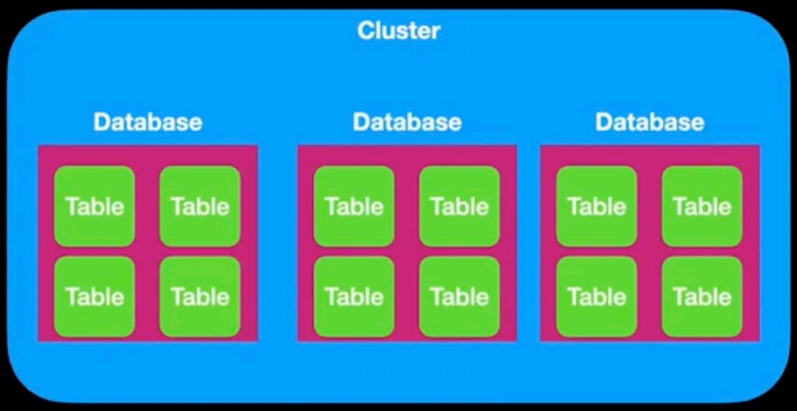
3 - Table

⇒ No SQL:-

```
{
    name: Abc,
    pass: IDK,
    courses: ["Array", "Array"]
}
```

⇒ object ID = randomly generated unique

3 Jargons to know in Databases
1. Cluster
2. Database
3. Table



⇒ why we use JWT? - bearer.
→ to save db calls. JWT will store users so with token we can verify he is existing user or not.

→ Middlewcure Job :-

→ end req

→ forword req

→ pass data along to the next fun.