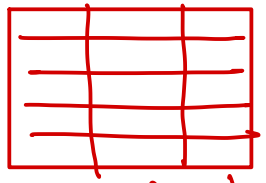
all RDBMS → SQL

↓

Relational

↓

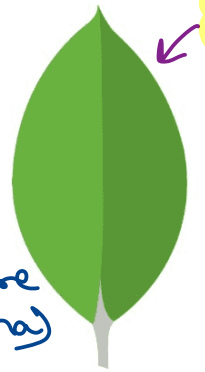
tabular [T1 ↔ T2]



Rows & cols

↑

- fixed structure (schema)
- 100s of GB
- high consistency / transactions.



mongo

↑

cassandra

neo 4j

↑

redis

- flexible schema
- 100s of TB / PB (scaling)
  - data
  - clients
- economical
- eventual consistency

NoSQL Databases

→ no standard query language X

Trainer: Mr. Nilesh Ghule

Not Only SQL

Document oriented databases

Java Script Object Notation

- Document contains data as key-value pair as JSON or XML.
- Document schema is flexible & are added in collection for processing.
- RDBMS tables → Collections
- RDBMS rows → Documents
- RDBMS columns → Key-value pairs in document
- Examples: MongoDb, CouchDb, …

b1 JSON → Java Script Object Notation.

```
{
  "id" : 1,        → int
  "title" : "Let us C",   → string
  "author" : "Kanetkar",
  "price" : 240.4   → double
}
```

r1 → JSON document

```
{ id : 1,
  name : "Nilesh",
  age : 38,
  hobbies : ["Program", "Reading", ...]
  addr : { area : "Katraj", city : "Pune", pin : 411046 },
  political : false,
  height : 5.9,
  bloodgroup : null
}
```

MongoDb Databases

Trainer: Mr. Nilesh Ghule

Mongo Db

- Developed by 10gen in 2007
- Publicly available in 2009
- Open-source database which is controlled by 10gen
- Document oriented database → stores JSON documents
- Stores data in binary JSON. → (BSON) → faster access.
- Design Philosophy
  - MongoDB wasn't designed in a lab and is instead built from the experiences of building large scale, high availability, and robust systems.

PBs 24x7

JSON

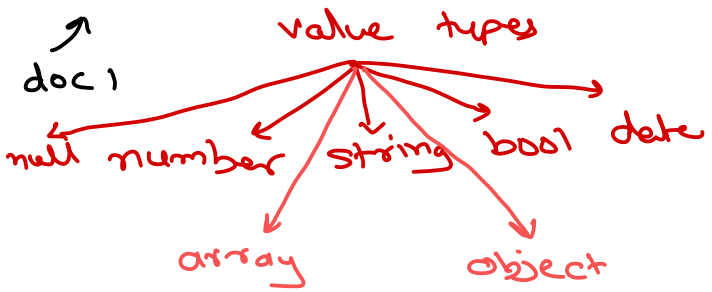
- Java Script Object Notation
- Hierarchical way of organizing data
- Mongo stores JSON data into Binary form. → data type internally identified by a number.

people → collection

doc 1

```
{
    "name" : "Nitin",
    "mobile" : "9881208115",
    "pin" : 411037,
    "sal" : 50000.00
}
```

key, value

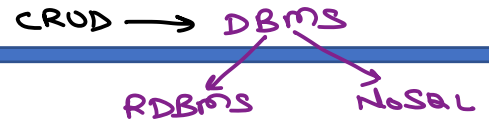


doc 2

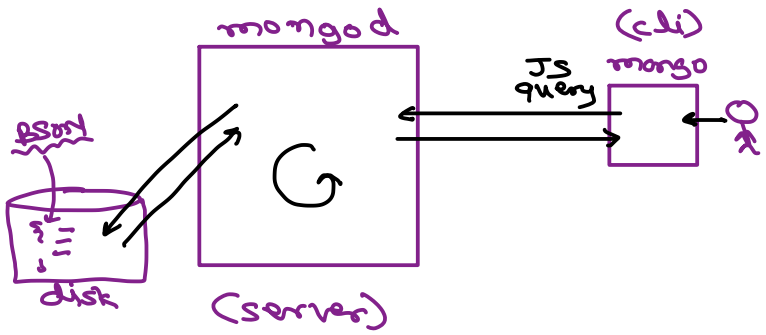
```
{
    "name" : "Nilesh",
    "addr" : {
        "area" : "katraj",
        "city" : "pune",
        "pin" : 411046
    },
    "hobbies" : ["programming", "reading", "cooking"],
    "age" : 37,
    "sal" : 300000.0,
    "political" : false,
    "rating" : null
}
```

string, json object, array, number, boolean, null

Mongo Server and Client



- MongoDb server (mongod) is developed in C, C++ and JS.
- MongoDb data is accessed via multiple client tools
  - mongo : client shell (JS).
  - mongofiles : stores larger files in GridFS.
  - mongoimport / mongoexport : tools for data import / export.
  - mongodump / mongorestore : tools for backup / restore.



- MongoDb data can be accessed in application through client drivers available for all major programming languages e.g. Java, Python, Ruby, PHP, Perl, …
- Mongo shell is follows JS syntax and allow to execute JS scripts.

MongoDb: Data Types

data	bson	values
null	10	
boolean	8	true, false
number	1 / 16 / 18	123, 456.78, NumberInt(“24”), NumberLong(“28”)
string	2	“....”
date	9	new Date(), ISODate(“yyyy-mm-ddThh:mm:ss”)
array	4	[…, …, …, …]
object	3	{ … }

Mongo - INSERT

- show databases;
- use database;
- db.contacts.insert({name: "nilesh", mobile: "9527331338"});
- db.contacts.insertMany([

  {name: "nilesh", mobile: "9527331338"},

  {name: "nitin", mobile: "9881208115"}

  ]);
- Maximum document size is 16 MB.
- For each object unique id is generated by client (if _id not provided).
  - 12 byte unique id :: [counter(3) | pid(2) | machine(3) | timestamp(4)]

Mongo – QUERY

- db.contacts.find(); → returns cursor on which following ops allowed:
  - hasNext(), next(), skip(n), limit(n), count(), toArray(), forEach(fn), pretty()
- Shell restrict to fetch 20 records at once. Press "it" for more records.
- db.contacts.find({ name: "nilesh" });
- db.contacts.find({ name: "nilesh" }, { _id:0, name:1 });
- Relational operators: $eq, $ne, $gt, $lt, $gte, $lte, $in, $nin
- Logical operators: $and, $or, $nor, $not
- Element operators: $exists, $type
- Evaluation operators: $regex, $where, $mod
- Array operators: $size, $elemMatch, $all, $slice

Thank you!

Nilesh Ghule <nilesh@sunbeaminfo.com>