**Mongo DB**

* Databases are used to store structured data
* Databases support various operations on the data
  + Query
  + Insert
  + Update
  + Delete
* SQL (structured query language) based relational databases have been very popular means of storing data
* NoSQL databases are increasingly becoming popular to address some challenges encountered w SQL databases (e.g., mongoDB)

NoSQL databases

* 4 catagories
  + Document databases (e.g., MongoDB)
  + Key-value databases (e.g., Redis)
  + Column-family databases (e.g., Cassandra)
  + Graph databases (e.g., Neo4J)

Document databases

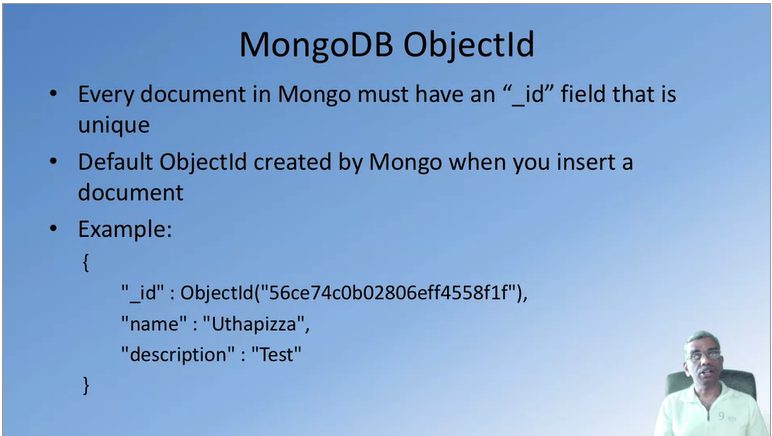
* Document databases are built around documents
* A self-contained piece of information
  + Can be stored in many formats such as JSON
* Collection: Documents can be organized in a collection
* Database: a set of collections

Why NoSQL?

* Scalability: SQL databases cannot offer availability and consistency simultaneously, while NoSQL databases can offer both
  + Availability
  + Consistency
  + Partition tolerance
* Ease of deployment
  + No object-relation mapping required
    - This is usually needed when mapping data from an SQL database to objects in the native language (an intermediate gateway is required for this)

MongoDB

* Document database
  + Server can support multiple databases
  + A database is a set of collections
  + A collection is a set of documents
  + Document is effectively a JSON document w some additional features
* Mongo stores the documents in BSON (Binary JSON) format
  + Supports length prefix on each value
    - Easy to skip over a field
  + Information abt the type of a field value
  + Additional primitive types not supported by raw JSON like UTC date time, raw binary, and ObjectId
* MondoDB ObjectId
  + Every document in Mongo must have an “\_id” field that is unique
  + Default ObjectId created by Mongo when you insert a document
  + Example:



* + id.getTimestamp() returns the timestamp in ISO Date format