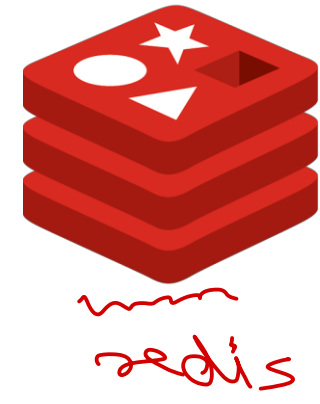
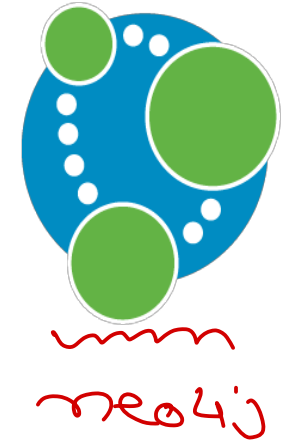
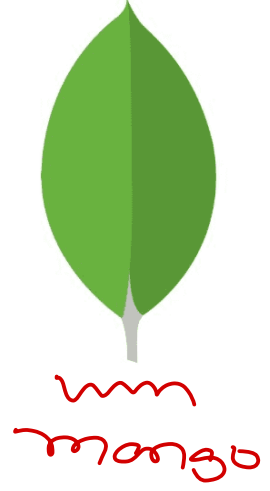


~~SOL~~

→ NoSQL

Anti



# NoSQL Databases

Trainer: Mr. Nilesh Ghule

# Mongo- Aggregation Pipeline

- db.collection.aggregate( [ { stage1 }, { stage2 }, ... ] );
- \$project → select columns (existing or computed)
- \$match → where clause (criteria)
- \$group → group by
  - { \$group: { \_id: <expr>, <field1>: { <accum1> : <expr1> }, ... } }
  - The possible accumulators are: \$sum, \$avg, ...
- \$unwind → extract array elements from array field
- \$lookup → left outer join
  - { \$lookup: { from: other\_col, localField: cur\_col\_field, foreignField: other\_col\_field, as: arr\_field\_alias } }
- \$out → put result of pipeline in another collection (last operation)

*\$sort*

*\$limit*

*\$skip*



# Mongo - MapReduce

- For processing large volumes of data into useful aggregate results.
- Mongodb applies map phase to each input document. The map function emits key-value pairs.
- For keys with multiple values, mongodb applies reduce phase.
- All map-reduce functions written in JS and are executed in mongod server process.
- The map-reduce works on single collection data.
- The output of map-reduce can be written into some collection.
- The input & output collections can be sharded.
- The aggregation framework provides better performance. MR is used for functionalities not available in aggregation framework.



# Mongo – DELETE

- `db.contacts.remove(criteria);`
- `db.contacts.deleteOne(criteria);`
- `db.contacts.deleteMany(criteria);`
- `db.contacts.deleteMany({});` → delete all docs, but not collection
- `db.contacts.drop();` → delete all docs & collection as well : efficient



# Mongo – UPDATE

- `db.contacts.update(criteria, newObj);`
- Update operators: `$set`, `$inc`, ~~`$dec`~~, `$push`, `$each`, `$slice`, `$pull`
- In place updates are faster (e.g. `$inc`, `$dec`, ...) than setting new object. If new object size mismatch with older object, data files are fragmented.
- Update operators: `$addToSet`
- example: `db.contacts.update( { name: "peter" },`  
`{ $push : { mobile: { $each : ["111", "222" ], $slice : -3 } } } );`
- `db.contacts.update( { name: "t" }, { $set : { "phone" : "123" } }, true );`
  - If doc with given criteria is absent, new one is created before update.





Thank you!

Nilesh Ghule <nilesh@sunbeaminfo.com>

