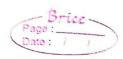
Rupesh Dharme 31124 Accionnes



Assignment 11 Page:
Title: Mongape map-reduce operations
Problem: Implement map-reduce operations with suitable example using mongoob
Objective: To understand concept of map-reduce as data processing paradigm for condensing large volumes of data into useful aggreation results.
Mongoob 5.0 Mongoob 5.0 Mongoob 5.0
Theory: Map reduce is a data processing paradigm for condensing, large volumes of data into useful aggregated results. following is the syntax of mapreduce:
db.collection_name. mapReduce (Function () { emit(key, value); }, function (key, value) { return reduce function}, { query: document, sut: document }



	In the above and a
	that mans a
	a volue with a key and emits a
	key-value pair.
	001.
	Reduce is a Is Function that reduces or
	groups all the documents have no the same
_	key:
_	out specifies the location of the map reduce
_	query result.
	Query specifies the optional selection criteria
	for selecting documents.
	Conclusion:
	Succesfully implemented mapkeduce
	Operation in mongons
-	

```
show dbs
use assignment11
db
db.materials.insertMany([
name: 'aluminium',
rate:200,
weight:'Light'
},
name: 'steel, mild',
rate:300,
weight:'Medium'
},
name: 'steel, stainless',
rate:100,
weight: 'Heavy'
},
name: 'abs plastic',
rate:100,
weight:'Light'
},
name:'PVC',
rate:200,
weight:'Light'
}
])
db.materials.find().pretty()
db.materials.mapReduce(
function(){
emit(this.name,this.rate);},
function(key, value){
return Array.sum(value)},
query:{weight:'Light'},
out:'total_cost'
```

db.total_cost.find().pretty()

```
> use assignment11
switched to db assignment11
assignment11
    . name:'aluminium',
. rate:200,
. weight:'Light'
      rate:300,
weight:'Medium'
      name:'steel, stainless',
rate:100,
weight:'Heavy'
      name: 'abs plastic',
rate:100,
weight: 'Light'
      name:'PVC',
rate:200,
weight:'Light'
              "acknowledged" : true,
"insertedIds" : [
                            edIds" : [
ObjectId("61972a3a250bf58329980b8b"),
ObjectId("61972a3a250bf58329980b8c"),
ObjectId("61972a3a250bf58329980b8c"),
ObjectId("61972a3a250bf58329980b8c"),
ObjectId("61972a3250bf58329980b8c"),
                                                                            O H 🕝 🥲 🔚 🔒 🧿 🐧 🚺 📵
 Type here to search
                                                                                                                                                                                                                                                ^ ➡ ∰ Φ)  ENG 10:14 ☐
                             ObjectId("61972a3a250bf58329980b8c"),
ObjectId("61972a3a250bf58329980b8c"),
ObjectId("61972a3a250bf58329980b8c"),
ObjectId("61972a3a250bf58329980b8c"),
             "_id" : ObjectId("61972a3a250bf58329980b8b"),
"name" : "aluminium",
"rate" : 200,
"weight" : "Light"
             "_id" : ObjectId("61972a3a250bf58329980b8c"),
"name" : "steel, mild",
"rate" : 300,
"weight" : "Medium"
              "_id" : ObjectId("61972a3a250bf58329980b8d"),
"name" : "steel, stainless",
"rate" : 100,
"weight" : "Heavy"
             "_id" : ObjectId("61972a3a250bf58329980b8e"),
"name" : "abs plastic",
"rate" : 100,
"weight" : "Light"
             "_id" : ObjectId("61972a3a250bf58329980b8f"),
"name" : "PVC",
"rate" : 20",
"weight" : "Light"
                                                                             O H , O M A O
                                                                                                                                                                                                                                                 ヘ ■ 口 ゆ) === ENG 10:15 □
 Type here to search
```

```
db.total_cost.find().pretty()
{ "_id" : "abs plastic", "value" : 100 }
{ "_id" : "aluminium", "value" : 200 }

{ "_id" : "PVC", "value" : 200 }

* CAProgram Files\MongoDB\Server\S.O\bin\mongo.exe

> db.materials.mapReduce(
... function(){
... emit(this.name,this.rate);},
... function(key, value){
... return Array.sum(value)},
... {
... query:{weight:'Light'},
... out:'total_cost'
... }
... )
... )
... }
... ("result" : "total_cost", "ok" : 1 }
```