**3. MongoDB – Map-reduces operations:**

**Implement Map reduces operation with suitable example using MongoDB.**

> db.createCollection("mapreduce")

{ "ok" : 1 }

> db.mapreduce.insert({cust\_id:1,status:'A',amount:400})

WriteResult({ "nInserted" : 1 })

> db.mapreduce.insert({cust\_id:2,status:'A',amount:400})

WriteResult({ "nInserted" : 1 })

> db.mapreduce.insert({cust\_id:1,status:'B',amount:500})

WriteResult({ "nInserted" : 1 })

> db.mapreduce.insert({cust\_id:1,status:'B',amount:400})

WriteResult({ "nInserted" : 1 })

> db.mapreduce.insert({cust\_id:4,status:'B',amount:400})

WriteResult({ "nInserted" : 1 })

> db.mapreduce.insert({cust\_id:4,status:'A',amount:400})

WriteResult({ "nInserted" : 1 })

> db.mapreduce.insert({cust\_id:1,status:'A',amount:400})

WriteResult({ "nInserted" : 1 })

> db.mapreduce.insert({cust\_id:4,status:'A',amount:600})

WriteResult({ "nInserted" : 1 })

> db.mapreduce.insert({cust\_id:4,status:'B',amount:600})

WriteResult({ "nInserted" : 1 })

> db.mapreduce.insert({cust\_id:1,status:'A',amount:600})

WriteResult({ "nInserted" : 1 })

> db.mapreduce.find().pretty()

{

"\_id" : ObjectId("61a1b903f7a71b32689267db"),

"cust\_id" : 1,

"status" : "A",

"amount" : 400

}

{

"\_id" : ObjectId("61a1b903f7a71b32689267dc"),

"cust\_id" : 2,

"status" : "A",

"amount" : 400

}

{

"\_id" : ObjectId("61a1b903f7a71b32689267dd"),

"cust\_id" : 1,

"status" : "B",

"amount" : 500

}

{

"\_id" : ObjectId("61a1b903f7a71b32689267de"),

"cust\_id" : 1,

"status" : "B",

"amount" : 400

}

{

"\_id" : ObjectId("61a1b903f7a71b32689267df"),

"cust\_id" : 4,

"status" : "B",

"amount" : 400

}

{

"\_id" : ObjectId("61a1b903f7a71b32689267e0"),

"cust\_id" : 4,

"status" : "A",

"amount" : 400

}

{

"\_id" : ObjectId("61a1b903f7a71b32689267e1"),

"cust\_id" : 1,

"status" : "A",

"amount" : 400

}

{

"\_id" : ObjectId("61a1b903f7a71b32689267e2"),

"cust\_id" : 4,

"status" : "A",

"amount" : 600

}

{

"\_id" : ObjectId("61a1b903f7a71b32689267e3"),

"cust\_id" : 4,

"status" : "B",

"amount" : 600

}

{

"\_id" : ObjectId("61a1b906f7a71b32689267e4"),

"cust\_id" : 1,

"status" : "A",

"amount" : 600

}

> db.mapreduce.aggregate([{$match:{status:"A"}},{$group:{\_id:"$cust\_id",total:{$

sum:"$amount"}}}])

{ "\_id" : 2, "total" : 400 }

{ "\_id" : 4, "total" : 1000 }

{ "\_id" : 1, "total" : 1400 }