Assignment No:5(Group C)

Title: Create a database with suitable example using MongoDB and

implement

1.Inserting and saving document(batch insert,insert validation)

2.Removing document

3.Updating document (document replacement,using modifiers,insert,updating multiple documents,returning updated documents)

Batch No: T1 Roll No:06

-----------------------------------------------------------------

> use aggrigation

switched to db aggrigation

> db.student.insert({'name':'john','id':1,'branch':'computer','class':'BE'})

WriteResult({ "nInserted" : 1 })

> db.student.insert({'name':'mayuri','id':2,'branch':'civil','class':'TE'})

WriteResult({ "nInserted" : 1 })

> db.student.insert({'name':'aman','id':3,'branch':'ETC','class':'TE'})

WriteResult({ "nInserted" : 1 })

> db.student.insert({'name':'anand','id':4,'branch':'ETC','class':'SE'})

WriteResult({ "nInserted" : 1 })

> db.student.find()

{ "\_id" : ObjectId("59d3d038f8efd48f38c26f0f"), "name" : "john", "id" : 1, "branch" : "computer", "class" : "BE" }

{ "\_id" : ObjectId("59d3d09ff8efd48f38c26f10"), "name" : "mayuri", "id" : 2, "branch" : "civil", "class" : "TE" }

{ "\_id" : ObjectId("59d3d0bdf8efd48f38c26f11"), "name" : "aman", "id" : 3, "branch" : "ETC", "class" : "TE" }

{ "\_id" : ObjectId("59d3d0d6f8efd48f38c26f12"), "name" : "anand", "id" : 4, "branch" : "ETC", "class" : "SE" }

> db.student.find().pretty()

{

"\_id" : ObjectId("59d3d038f8efd48f38c26f0f"),

"name" : "john",

"id" : 1,

"branch" : "computer",

"class" : "BE"

}

{

"\_id" : ObjectId("59d3d09ff8efd48f38c26f10"),

"name" : "mayuri",

"id" : 2,

"branch" : "civil",

"class" : "TE"

}

{

"\_id" : ObjectId("59d3d0bdf8efd48f38c26f11"),

"name" : "aman",

"id" : 3,

"branch" : "ETC",

"class" : "TE"

}

{

"\_id" : ObjectId("59d3d0d6f8efd48f38c26f12"),

"name" : "anand",

"id" : 4,

"branch" : "ETC",

"class" : "SE"

}

> db.student.aggregate([{$group : {\_id : "$branch", total : {$sum : 1}}}])

{ "id" : "ETC", "total" : 2 }

{ "id" : "civil", "total" : 1 }

{ "id" : "computer", "total" : 1 }

> db.student.ensureIndex({"name":1})

{

"createdCollectionAutomatically" : false,

"numIndexesBefore" : 3,

"numIndexesAfter" : 3,

"note" : "all indexes already exist",

"ok" : 1

}