Chit 9 –

use orders

db.createCollection("orderinfo")

db.orderinfo.insert([

... {

... "cust\_id": 123,

... "cust\_name": "abc",

... "status": "A",

... "price": 250

... },

... {

... "cust\_id": 123,

... "cust\_name": "abc",

... "status": "A",

... "price": 300

... },

... {

... "cust\_id": 123,

... "cust\_name": "abc",

... "status": "A",

... "price": 450

... },

... {

... "cust\_id": 456,

... "cust\_name": "xyz",

... "status": "B",

... "price": 600

... },

... {

... "cust\_id": 456,

... "cust\_name": "xyz",

... "status": "B",

... "price": 100

... },

... {

... "cust\_id": 456,

... "cust\_name": "xyz",

... "status": "A",

... "price": 900

... },

... {

... "cust\_id": 789,

... "cust\_name": "def",

... "status": "A",

... "price": 300

... },

... {

... "cust\_id": 789,

... "cust\_name": "def",

... "status": "A",

... "price": 150

... },

... {

... "cust\_id": 789,

... "cust\_name": "def",

... "status": "B",

... "price": 850

... },

... {

... "cust\_id": 789,

... "cust\_name": "def",

... "status": "B",

... "price": 1000

... }

... ])

QUERIES

• Find the total price for each customer and display in the order of total price.

db.orderinfo.aggregate(

... [

... { $group:

... {\_id: "$cust\_id",

... total\_price:

... {$sum: "$price"}

... }

... },

... { $sort:

... {total\_price: 1}

... }

... ])

• Find the distinct customer names

db.orderinfo.distinct("cust\_name")

• Display the “price” of customers whose status is 'A'

db.orderinfo.find(

... { status: 'A' },

... { cust\_name: 1, price: 1, \_id: 0 }

... ).pretty()

• Delete the customers whose status is 'A'

db.orderinfo.remove({status: 'A'})