**ITCS443 Parallel and Distributed System**

Lab5: MongoDB

1. Display all distinct cuisines

**Ans** db.restaurants.distinct(“cuisine”);

1. Display only “restaurant\_id”, “name”, and “cuisine” fields of all restaurants without the “\_id” field

**Ans** db.restaurants.find({}, {restaurant\_id: true, name: true, cuisine: true, \_id: false});

1. Find restaurants (display only “name”) in Manhattan (“borough” field equals “Manhattan”)

**Ans** db.restaurants.find({"borough": "Manhattan"}, {"name": 1});

1. Find restaurants (display only “name”) in Manhattan which has cuisine “Bakery”

**Ans** db.restaurants.find({"borough": "Manhattan", "cuisine": "Bakery"}, {"name": 1});

1. Find restaurants (display “name” and “borough”) in Manhattan or in Bronx

**Ans** db.restaurants.find({$or: [{"borough": "Manhattan"}, {"borough": "Bronx"}]}, {"name":1, "borough":1}));

1. Find restaurants (display “name”, “borough”, and “cuisine”) in Manhattan or in Bronx which has cuisine “Bakery”

**Ans** db.restaurants.find({$or: [{"borough": "Manhattan", "cuisine": "Bakery"}, {"borough": "Bronx", "cuisine": "Bakery"}]}, {"name":1, "borough":1});

1. Find restaurants (display only “name”) in Bronx, sorted by “name” in descending order

**Ans** db.restaurants.find({"borough": "Bronx"}, {"name":1}).sort({"name": -1});

1. Find all restaurant in Bronx with “zipcode” equals 10462

**Ans** db.restaurants.find({"borough": "Bronx", "address.zipcode": "10462"});