TASK – 15/09/2020

**Task 1 : Add and remove fields**

db.Profile.updateMany**({},{**$set**:{**"joindate"**:** new Date**()}})**

db.Profile.updateMany**({},{**$unset**:{**"joindate"**:**""**}})**

**Task 2 : Perform following queries**

// Select name which dont contain "bc"

db.Profile.find**({**"NAME"**:{**$not**:**/bc/**}})**

// cursor queries

var p = db.Registration.find**({**RegNo**:{**$gt**:**2**}})**

while**(**p.hasNext**())** print**(**tojson**(**p.next**()))**

**Task 3 : Perform following queries**

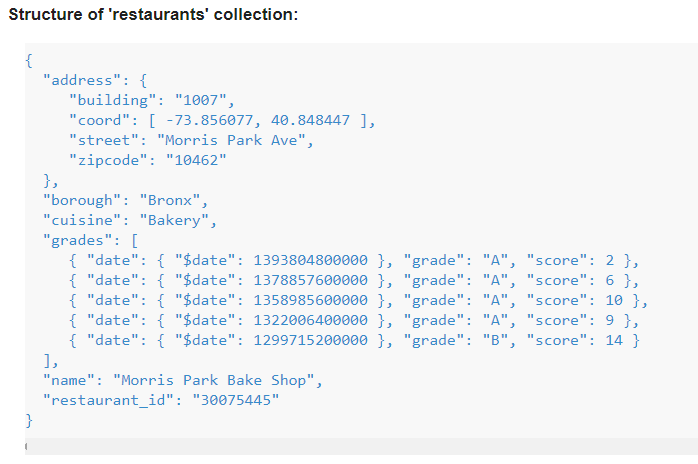
// 1. Using the available Collection/Table perform the Group By in mongoDb.

db.sales.aggregate**([{**$group**:{**\_id **:** "$item"**,**totalSaleAmount**:** **{** $sum**:** **{** $multiply**:** **[** "$price"**,** "$quantity" **]}}}}])**

// 2. Using the available Collection/Table perform the Orderby By in mongoDb

db.Profile.find**(** **{** $query**:** **{},** $orderby**:** **{** NAME **:** -1 **}** **}** **)**

**Lab Task : Perform following queries**



// 1. Write a MongoDB query to arrange the name of the restaurants in descending along with all the columns

db.restaurants.find**()**.sort**({**name**:**-1**})**

// 2. Write a MongoDB query to arranged the name of the cuisine in ascending order and for that same cuisine borough should be in descending order.

db.restaurants.aggregate**([{**$sort**:{**cuisine**:**1**,**borough**:**-1**}}])**

// 3. Write a MongoDB query to know whether all the addresses contains the street or not.

db.restaurants.find**({**"address.street"**:{**$ne**:**null**}})**

// 4. Write a MongoDB query which will select all documents in the restaurants collection where the coord field value is Double.

db.restaurants.find**({**"address.coord"**:{**$type**:**"double"**}})**

// 5. Write a MongoDB query which will select the restaurant Id, name and grades for those restaurants which returns 0 as a remainder after dividing the score by 7.

db.restaurants.find**({**"grades.score"**:{**$mod**:[**7**,**0**]}},{**\_id**:**0**,**restaurant\_id**:**1**,**name**:**1**,**grades**:**1**})**

// 6. Write a MongoDB query to find the restaurant name, borough, longitude and attitude and cuisine for those restaurants which contains 'mon' as three letters somewhere in its name

db.restaurants.find**({**name**:**/mon/**},{**\_id**:**0**,**name**:**1**,**borough**:**1**,**"address.coord"**:**1**,**cuisine**:**1**})**

// 7. Write a MongoDB query to find the restaurant name, borough, longitude and latitude and cuisine for those restaurants which contain 'Mad' as first three letters of its name.

db.restaurants.find**({**name**:**/^Mad/**},{**\_id**:**0**,**name**:**1**,**borough**:**1**,**"address.coord"**:**1**,**cuisine**:**1**})**