ASSIGNMENT- 3

NAME : B. Yaswanth

REG.NO : 22BRS1269

QUESTIONS BASED ON UPDATE , AGGREGATE FUNCTIONS

Q1. The total revenue generated from all sales

```
> db.sales.aggregate([
    {
  $group:{_id:'$item',
  totalPrice:{$sum:'$price'},
  },
  },
  ]);
< {
    _id: 'Americanos',
    totalPrice: 33
  }
  {
    _id: 'Mochas',
    totalPrice: 25
  }
  {
    _id: 'Cappuccino',
    totalPrice: 23
  }
  {
    _id: 'Lattes',
    totalPrice: 40
  }
```

Q2. How many “Americanos” were sold in total?

```
> db.sales.find({item : 'Americanos'}).count()
< 4
VIT3>
```

Q3. Increase the price of “Lattes” by 5 units

```
> db.sales.find({item : 'Americanos'}).count()
< 4
> db.sales.updateMany({item : 'Lattes'},{$inc:{price:5}})
< {
    acknowledged: true,
    insertedId: null,
    matchedCount: 2,
    modifiedCount: 2,
    upsertedCount: 0
  }
```

```
> db.sales.find({item:'Lattes'})
< {
    _id: 3,
    item: 'Lattes',
    price: 20,
    size: 'Grande',
    quantity: 25,
    date: 2022-01-16T09:05:00.000Z
  }
  {
    _id: 7,
    item: 'Lattes',
    price: 30,
    size: 'Tall',
    quantity: 30,
    date: 2022-02-21T10:08:00.000Z
  }
VIT3>
```

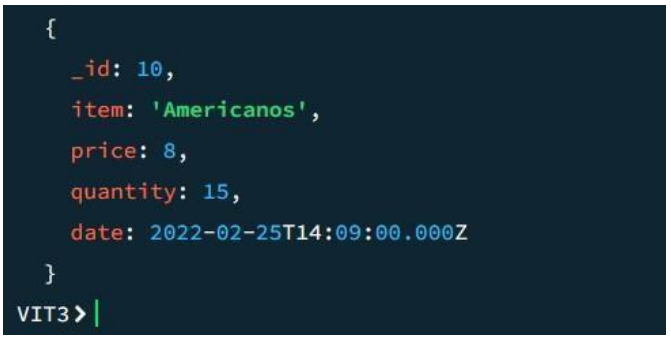
Q4. Unset the “size” field for all records where the price is less than 10.

```
> db.sales.updateMany({price:{$lt:10}},{$unset:{size:""}})
< {
    acknowledged: true,
    insertedId: null,
    matchedCount: 4,
    modifiedCount: 4,
    upsertedCount: 0
  }
VIT3>
```

```
> db.sales.find()
< {
    _id: 1,
    item: 'Americanos',
    price: 5,
    quantity: 22,
    date: 2022-01-15T08:00:00.000Z
  }
  {
    _id: 2,
    item: 'Cappuccino',
    price: 6,
    quantity: 12,
    date: 2022-01-16T09:00:00.000Z
  }
  {
    _id: 3,
    item: 'Lattes',
    price: 20,
    size: 'Grande',
    quantity: 25,
    date: 2022-01-16T09:05:00.000Z
  }
```


{
_id: 4,
item: 'Mochas',
price: 25,
size: 'Tall',
quantity: 11,
date: 2022-02-17T08:00:00.000Z
}
{
_id: 5,
item: 'Americanos',
price: 10,
size: 'Grande',
quantity: 12,
date: 2022-02-18T21:06:00.000Z
}
{
_id: 6,
item: 'Cappuccino',
price: 7,
quantity: 20,
date: 2022-02-20T10:07:00.000Z
}


{
_id: 7,
item: 'Lattes',
price: 30,
size: 'Tall',
quantity: 30,
date: 2022-02-21T10:08:00.000Z
}
{
_id: 8,
item: 'Americanos',
price: 10,
size: 'Grande',
quantity: 21,
date: 2022-02-22T14:09:00.000Z
}
{
_id: 9,
item: 'Cappuccino',
price: 10,
size: 'Grande',
quantity: 17,
date: 2022-02-23T14:09:00.000Z
}


{
_id: 10,
item: 'Americanos',
price: 8,
quantity: 15,
date: 2022-02-25T14:09:00.000Z
}
VIT3>

Q5. Find the average quantity sold for each item where the price is greater than $10, then sort these averages in descending order, skip the first result, and limit the output to 2 items.

```
> db.sales.aggregate([{$match:{price:{$gt:10}}},
  {$group:{_id:"$item",
  averageQuantity:{$avg:"$quantity"}
  }},
  {$sort:{averageQuantity:-1}},{$skip:1},{$limit:2}
  ])
< {
    _id: 'Mochas',
    averageQuantity: 11
  }
```

Q6. Identify the total quantity sold for each item, but only include those items where the maximum price recorded is at least $25. Then, sort the results by the total quantity in ascending order and limit the result to 1 item.

```
> db.sales.aggregate([
    {$group : {_id: "$item", totalQuantity: {$sum: "$quantity"}, maxPrice: {$max:"$price"}}},
    {$match: {maxPrice: {$gte:25}}},
    {$sort: {totalQuantity: 1}},
    {$limit: 1}
  ])
< {
    _id: 'Mochas',
    totalQuantity: 11,
    maxPrice: 25
  }
VIT3>
```

QUERIES RELATED TO INDEXING :

```
>_MONGOSH

use vit5

switched to db vit5
```

```
])

{  acknowledged:
true,

 insertedIds: {
```

```
    '0': 1,
    '1': 2,
    '2': 3,
    '3': 4,
    '4': 5,
    '5': 6,
    '6': 7
  }
}
db.products.find()

{
  _id: 1,
  name: 'xPhone',
  price: 799,  releaseDate: 2011-05-
14T00:00:00.000Z,

  spec: {
ram: 4,

    screen: 6.5,
    cpu: 2.66
  },
  color: [
    'white',
    'black'
  ],
```

```
  storage: [
    64,
    128,
    256
  ]
}
```

```
{
 _id: 2,
 name: 'xTablet',
price: 899,
 releaseDate: 2011-09-01T00:00:00.000Z,
 spec: {
ram: 16,
   screen: 9.5,
   cpu: 3.66
 },
 color: [
   'white',
   'black',
   'purple'
 ],
 storage: [
   128,
   256,
   512
 ]
}
{
 _id: 3,
 name: 'SmartTablet',
price: 899,
 releaseDate: 2015-01-14T00:00:00.000Z,
spec: {
   ram: 12,
   screen: 9.7,
   cpu: 3.66
 },
```

```
 color: [
'blue'

 ],
 storage: [
   16,
   64,
   128
 ]
}
{
 _id: 4,
 name: 'SmartPad',
price: 699,

 releaseDate: 2020-05-14T00:00:00.000Z,
spec: {

   ram: 8,
   screen: 9.7,
cpu: 1.66

 },
color: [

   'white',
   'orange',
   'gold',
   'gray'
 ],
 storage: [
   128,
   256,
   1024
 ]
}
```

```
{
 _id: 5,  name:
'SmartPhone',

 price: 599,  releaseDate: 2022-09-
14T00:00:00.000Z,

 spec: {
   ram: 4,
screen: 9.7,

   cpu: 1.66
 },
 color: [
   'white',
   'orange',
   'gold',
   'gray'
 ],
 storage: [
   128,
   256
 ]
}
{
 _id: 6,
 name: 'xWidget',
 spec: {
ram: 64,

   screen: 9.7,
cpu: 3.66

 },
 color: [
   'black'
```

```
 ],
storage: [

   1024
```

```
 ]

}

{

 _id: 7,  name:
'xReader',

 price: null,

 spec: {
ram: 64,

   screen: 6.7,
cpu: 3.66

 },

 color: [

   'black',

   'white'

 ],
storage: [

   128

 ]

}

db.products.createIndex({price : 1})
price_1

db.products.getIndexes()

[

 { v: 2, key: { _id: 1 }, name: '_id_' },

 { v: 2, key: { price: 1 }, name: 'price_1' }

]

db.products.find({price:699})

{
```

```
 _id: 4,
 name: 'SmartPad',
price: 699,
 releaseDate: 2020-05-14T00:00:00.000Z,
spec: {
   ram: 8,
   screen: 9.7,
   cpu: 1.66
 },
 color: [
   'white',
   'orange',
   'gold',
   'gray'
 ],
storage: [
   128,
   256,
   1024
 ]
}
db.products.dropIndex({price:1})

{ nIndexesWas: 2, ok: 1 }
db.products.getIndexes()

[ { v: 2, key: { _id: 1 }, name: '_id_' } ]

db.products.createIndex({name:-1,price:1}) name_-
1_price_1

db.products.getIndexes()

[
 { v: 2, key: { _id: 1 }, name: '_id_' },
```

```
 { v: 2, key: { name: -1, price: 1 }, name: 'name_-1_price_1' }

]

db.products.dropIndex("name_-1_price_1")

{ nIndexesWas: 2, ok: 1 }

db.products.getIndexes()

[ { v: 2, key: { _id: 1 }, name: '_id_' } ]
db.products.createIndex({"spec.ram":-1,"spec.cpu":1,"spec.screen":1})


spec.ram_-1_spec.cpu_1_spec.screen_1
db.products.find({"spec.ram":4}).explain()


{
```

```
 explainVersion: '1',

 queryPlanner: {
namespace: 'vit5.products',


   indexFilterSet: false,
parsedQuery: {


     'spec.ram': {

       '$eq': 4

     }

   },    queryHash:
'87BC90A8',


   planCacheKey: 'A7542BD7',

   maxIndexedOrSolutionsReached: false,
maxIndexedAndSolutionsReached: false,


   maxScansToExplodeReached: false,

   winningPlan: {
stage: 'FETCH',


     inputStage: {
stage: 'IXSCAN',


       keyPattern: {

         'spec.ram': -1,

         'spec.cpu': 1,
```

```
          'spec.screen': 1

        },

        indexName: 'spec.ram_-1_spec.cpu_1_spec.screen_1',

        isMultiKey: false,
multiKeyPaths: {


          'spec.ram': [],

          'spec.cpu': [],

          'spec.screen': []

        },
isUnique: false,


        isSparse: false,
isPartial: false,


        indexVersion: 2,
```

```
        direction: 'forward',

        indexBounds: {

          'spec.ram': [

            '[4, 4]'

          ],

          'spec.cpu': [

            '[MinKey, MaxKey]'

          ],

          'spec.screen': [

            '[MinKey, MaxKey]'

          ]

        }

      }

    },

    rejectedPlans: []

  },

  command: {
find: 'products',
```

```
   filter: {
     'spec.ram': 4
   },
   '$db': 'vit5'
 },
 serverInfo: {
   host: 'LAPTOP-E0TPMAQ3',
port: 27017,
   version: '7.0.11',
db.products.getIndexes({"spec.ram":4})

[
 { v: 2, key: { _id: 1 }, name: '_id_' },
 {
   v: 2,    key: { 'spec.ram': -1, 'spec.cpu': 1,
'spec.screen': 1 },
   name: 'spec.ram_-1_spec.cpu_1_spec.screen_1'
 }
]
db.products.find({"spec.ram":4})


{
 _id: 5,
 name: 'SmartPhone',
 price: 599,  releaseDate: 2022-09-
14T00:00:00.000Z,
 spec: {
ram: 4,
   screen: 9.7,
   cpu: 1.66
 },
 color: [
```

```
   'white',
   'orange',
   'gold',
   'gray'
 ],
 storage: [
   128,
   256
 ]
}
{
 _id: 1,
 name: 'xPhone',
price: 799,

 releaseDate: 2011-05-14T00:00:00.000Z,
 spec: {
ram: 4,

   screen: 6.5,
cpu: 2.66

 },
 color: [
   'white',
   'black'
 ],
storage: [

   64,
   128,
   256
 ]
}
use VIT6
```

```
switched to db VIT6
db.createCollection("blog")

{ ok: 1 }
```

```
> db.blog.insertMany( [
    {
      _id: 1,
      content: "This morning I had a cup of coffee.",
      about: {author: "Bharathi", writer:"Bharathi Varsha"},
      keywords: [ "coffee" ]
    },
    {
      _id: 2,
      content: "Who likes chocolate ice cream for dessert?",
      about: {author : "Sri", writer : "Sri Devi"},
      keywords: [ "poll" ]
    },
    {
      _id: 3,
      content: "My favorite flavors are strawberry and coffee",
      about: {author : "Sri",writer : "Sri Thanvi"},
      keywords: [ "food", "dessert" ]
    }
  ])
```

```
{  acknowledged:
true,
```

```
 insertedIds: {
   '0': 1,
   '1': 2,
   '2': 3
 }
}
db.blog.find()
{
```

```
 _id: 1,  content: 'This morning I had a cup
of coffee.',

 about: {

   author: 'Bharathi',
writer: 'Bharathi Varsha'

 },

 keywords: [

   'coffee'

 ]

}

{

 _id: 2,  content: 'Who likes chocolate ice cream for
dessert?',

 about: {
author: 'Sri',

   writer: 'Sri Devi'

 },
keywords: [

   'poll'

 ]

}

{

 _id: 3,

 content: 'My favorite flavors are strawberry and coffee',
about: {

   author: 'Sri',
```

```
   writer: 'Sri Thanvi'

 },
keywords: [

   'food',

   'dessert'
```

```
 ]

}
db.blog.createIndex({content:'text'})


content_text
db.blog.getIndexes()


[

 { v: 2, key: { _id: 1 }, name: '_id_' },

 {

   v: 2,

   key: { _fts: 'text', _ftsx: 1 },
name: 'content_text',


   weights: { content: 1 },
default_language: 'english',


   language_override: 'language',

   textIndexVersion: 3

 }

] db.blog.find({content: 'My favorite flavors are strawberry and
coffee'})


{

 _id: 3,  content: 'My favorite flavors are strawberry
and coffee',


 about: {
author: 'Sri',


   writer: 'Sri Thanvi'

 },
keywords: [


   'food',

   'dessert'

 ]
```

```
}
```

```
db.blog.find({$text : {$search : 'cream'}})

{

 _id: 2,  content: 'Who likes chocolate ice cream for
dessert?',


 about: {

   author: 'Sri',
writer: 'Sri Devi'


 },
keywords: [


   'poll'

 ]

}

db.blog.dropIndex("content_text")

{ nIndexesWas: 2, ok: 1 }
db.blog.createIndex({"$**":"text"})


$**_text
db.blog.getIndexes()


[

 { v: 2, key: { _id: 1 }, name: '_id_' },

 {

   v: 2,    key: { _fts: 'text',
_ftsx: 1 },


   name: '$**_text',

   weights: { '$**': 1 },
default_language: 'english',


   language_override: 'language',
textIndexVersion: 3


 }

]
db.blog.find({$text:{$search:"Raj"}})


db.blog.find({$text:{$search:"Sri"}})

{
```

```
 _id: 3,
```

```
 content: 'My favorite flavors are strawberry and coffee',

 about: {
author: 'Sri',


   writer: 'Sri Thanvi'

 },

 keywords: [

   'food',

   'dessert'

 ]

}

{

 _id: 2,  content: 'Who likes chocolate ice cream for
dessert?',


 about: {

   author: 'Sri',
writer: 'Sri Devi'


 },
keywords: [


   'poll'

 ]

}

db.blog.find({$text:{$search:"Sri  Varsha"}})

{

 _id: 3,

 content: 'My favorite flavors are strawberry and coffee',
about: {


   author: 'Sri',
writer: 'Sri Thanvi'


 },
```

```
 keywords: [
    'food',
    'dessert'
 ]
}
```

```
{
 _id: 2,  content: 'Who likes chocolate ice cream for
dessert?',

 about: {
author: 'Sri',

    writer: 'Sri Devi'
 },
keywords: [

    'poll'
 ]

}

{
 _id: 1,
 content: 'This morning I had a cup of coffee.',
 about: {    author:
'Bharathi',

    writer: 'Bharathi Varsha'
 },
 keywords: [
    'coffee'
 ]

}
db.blog.find({$text:{$search:"harathi"}})

db.blog.find({$text:{$search:"coffee  cream"}})

{
 _id: 1,
```

```
 content: 'This morning I had a cup of coffee.',
about: {

   author: 'Bharathi',

   writer: 'Bharathi Varsha'

 },

 keywords: [

   'coffee'

 ]
```

```
}

{

 _id: 3,

 content: 'My favorite flavors are strawberry and coffee',
about: {

   author: 'Sri',

   writer: 'Sri Thanvi'

 },

 keywords: [

   'food',

   'dessert'

 ]

}

{

 _id: 2,  content: 'Who likes chocolate ice cream for
dessert?',

 about: {
author: 'Sri',

   writer: 'Sri Devi'

 },
keywords: [

   'poll'

 ]
```

```
}

db.blog.find({$text:{$search:'\"ice cream for\"'}})

{

 _id: 2,  content: 'Who likes chocolate ice cream for
dessert?',


 about: {

   author: 'Sri',
writer: 'Sri Devi'


 },
keywords: [


   'poll' ]}
```