

## ST1501 CA2 Individual Task

Class: DIT2B22


Group No:

<u>AdmNo</u>	<u>Name</u>
<u>P2045302</u>	<u>Muhammad Nuhayah Putra Bin Ramlee</u>

### Individual Task

Item	SQL statement for JSON format
UnSold – create a collection to store those products which have never been sold	<pre>-----UnSold----- select p.product_id, p.product_name, b.brand_name, c.category_name, p.model_year, p from   BikeSalesGroup6.Production.products p ,   BikeSalesGroup6.Production.brands b ,   BikeSalesGroup6.Production.categories c where   product_id not in (     select distinct product_id from BikeSalesGroup6.sales.order_items   )and P.brand_id = B.brand_id and P.category_id = C.category_id for json path</pre>
ZeroStock – create a collection to store products which have zero stock	<pre>-----ZeroStock ----- select p.product_id,p.product_name,b.brand_name,c.category_name,p.model_year,p.list_price from   Production.products p   inner join Production.brands b   on p.brand_id=b.brand_id   inner join Production.categories c   on p.category_id=c.category_id where p.product_id not in( select distinct product_id from Production.stocks) for json path</pre>
Stock – create a collection to store products which have stock	<pre>---Stock collection ---- Select  product_id,store_id,quantity  from  Production.stocks for json</pre>

Item	MongoDB command	Results
Count of Unsold Collection	db.unSold.count({})	<pre>&gt; use bikes switched to db bikes &gt; db.unSold.count({}) 14 &gt;</pre>
Count of ZeroStock Collection	db.ZeroStock.count({})	<pre>&gt; db.ZeroStock.count({}) 8 &gt;</pre>
Count of Stock Collection	db.Stock.count({})	<pre>&gt; use bikes switched to db bikes &gt; db.Stock.count({}) 939 &gt;</pre>
Query 1	db.ZeroStock.find({ \$and: [{"category_name": "Road Bikes"}, {"list_price": {\$lt: 2000}}], {product_name: 1, _id: 0}).pretty()	<pre>"product_name" : "Trek Checkpoint ALR 4 Women's - 2019" } "product_name" : "Trek Checkpoint ALR 5 - 2019" } "product_name" : "Trek Checkpoint ALR 5 Women's - 2019" }</pre>
Query 2	db.unSold.distinct("category_name")  db.unSold.distinct("brand_name")	<pre>&gt; db.unSold.distinct("category_name") [   "Children Bicycles\r",   "Electric Bikes\r",   "Mountain Bikes\r",   "Road Bikes" ] &gt; db.unSold.distinct("brand_name") [ "      Electra \r", "      Surly \r", "      Trek \r" ] &gt;</pre>
Query 3	db.unSold.aggregate([{\$group: { _id: {brand_name: "\$brand_name", category_name: "\$category_name"} } }])	<pre>&gt; db.unSold.aggregate([{\$group: { _id: {brand_name: "\$brand_name", category_name: "\$category_name"} } }]) { "_id" : { "brand_name" : "      Trek \r", "category_name" : "Road Bikes" } } { "_id" : { "brand_name" : "      Electra \r", "category_name" : "Electric Bikes\r" } } { "_id" : { "brand_name" : "      Trek \r", "category_name" : "Children Bicycles\r" } } { "_id" : { "brand_name" : "      Surly \r", "category_name" : "Mountain Bikes\r" } } { "_id" : { "brand_name" : "      Trek \r", "category_name" : "Mountain Bikes\r" } } { "_id" : { "brand_name" : "      Electra \r", "category_name" : "Children Bicycles\r" } } &gt;</pre>

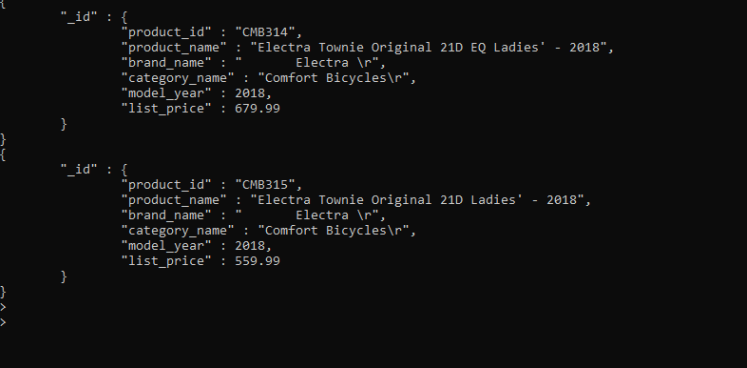
Query 4	<pre>db.unSold.aggregate([  {   \$lookup: {     from: "Stock",     localField: "product_id",     foreignField: "product_id",     as:"Unsold_with_Stock"   }  },  {   \$match:    {     "Unsold_with_Stock":     {        \$ne: [ ]     }   } }  ]).pretty()</pre>	 <pre>{   "_id" : ObjectId("62ea13d26a2cd491b4efa56b"),   "product_id" : "CHB267",   "product_name" : "Trek Precaliber 12 Girl's - 2018",   "brand_name" : "Trek",   "category_name" : "Children Bicycles",   "model_year" : 2018,   "list_price" : 199.99,   "Unsold_with_Stock" : [     {       "_id" : ObjectId("62ea0ff06bb6f128b67f6bea"),       "product_id" : "CHB267",       "store_id" : "ST1",       "quantity" : 27     },     {       "_id" : ObjectId("62ea0ff06bb6f128b67f6beb"),       "product_id" : "CHB267",       "store_id" : "ST2",       "quantity" : 21     },     {       "_id" : ObjectId("62ea0ff06bb6f128b67f6bec"),       "product_id" : "CHB267",       "store_id" : "ST3",       "quantity" : 12     }   ] }</pre>
---------	---	--

# Query 5

```
db.unSold.aggregate([
  {
    $lookup:
    {
      from: "Stock",
      localField: "product_id",
      foreignField: "product_id",
      as: "Unsold_with_Stock"
    }
  },
  {
    $match:
    {
      "Unsold_with_Stock":
      {
        $ne: []
      }
    }
  },
  {
    $unwind: "$Unsold_with_Stock"
  },
  {
    $group:
    {
      _id: "$product_id",

      quantity:{$sum:
"$Unsold_with_Stock.quantity"}
    }
  },
  {
    $sort: {"quantity": -1}
  }
]).pretty()
```

```
... ]).pretty()
{
  "_id" : "CHB267", "quantity" : 60 }
{
  "_id" : "CHB284", "quantity" : 59 }
{
  "_id" : "MOB1", "quantity" : 55 }
{
  "_id" : "RDB154", "quantity" : 44 }
{
  "_id" : "CHB291", "quantity" : 41 }
{
  "_id" : "ELB195", "quantity" : 40 }
{
  "_id" : "MOB125", "quantity" : 34 }
{
  "_id" : "MOB121", "quantity" : 26 }
}
```

Query 6	<pre>db.ZeroStock.aggregate([   {     \$lookup: {       from: "unSold",       localField: "product_id",       foreignField: "product_id",       as: "Sold_with_Stock"     },     \$match: {       "Sold_with_Stock": {         \$eq: []       }     },     \$group: {       _id: {         product_id: "\$product_id",         product_name: "\$product_name",         brand_name: "\$brand_name",         category_name: "\$category_name",         model_year: "\$model_year",         list_price: "\$list_price"       }     }   } ]).pretty()</pre>	 <pre>{   "_id" : {     "product_id" : "CMB314",     "product_name" : "Electra Townie Original 21D EQ Ladies' - 2018",     "brand_name" : "Electra",     "category_name" : "Comfort Bicycles",     "model_year" : 2018,     "list_price" : 679.99   } }, {   "_id" : {     "product_id" : "CMB315",     "product_name" : "Electra Townie Original 21D Ladies' - 2018",     "brand_name" : "Electra",     "category_name" : "Comfort Bicycles",     "model_year" : 2018,     "list_price" : 559.99   } }</pre>
---------	---	---