

ADVANCED DATABASE MANAGEMENT
SYSTEMS LAB – db.collection.aggregate()

Submitted To,
Lisha Varghese
Assistant Professor
Amal Jyothi College Of Engineering
Kanjirapally

Submitted By,
Teena Rose Mathew
RMCA-B Batch
Roll no: 35
Date: 25/09/2021

db.collection.aggregate()

Definition

db.collection.aggregate(**pipeline**, **options**)

Calculates aggregate values for the data in a collection.

Parameter	Type	Description
pipeline	array	A sequence of data aggregation operations or stages. See the aggregation pipeline operators for details. The method can still accept the pipeline stages as separate arguments instead of as elements in an array; however, if you do not specify the pipeline as an array, you cannot specify the options parameter.
options	document	Optional. Additional options that aggregate() passes to the aggregate command.

output

```
> db.purchase_orders.insertMany([{"product":"tooth paste",total: 4.75,customer:"Mike"},{"product":"gitar",total: 199.99,customer:"Karan"},{"product":"milk",total: 11.33,customer:"Mike"},{"product":"pizza",total: 8.58,customer:"Binu"}])
{
  "acknowledged" : true,
  "insertedIds" : [
    ObjectId("614ec493b96898a118437e72"),
    ObjectId("614ec493b96898a118437e73"),
    ObjectId("614ec493b96898a118437e74"),
    ObjectId("614ec493b96898a118437e75")
  ]
}
```

purchase_orders.purchase_orders

DOCUMENTS 4 TOTAL SIZE 306B AVG. SIZE 77B INDEXES 1 TOTAL SIZE 20.0KB

Documents Aggregations Schema Explain Plan Indexes Validation

Full { field: "value" } OPTIONS FIND RESET

ADD DATA VIEW

Displaying documents 1 - 4 of 4

```

{ "_id": "6056c1107a34ec248448198d3435caf48",
  "product": "tooth paste",
  "total": 4.75,
  "customer": "Mike" }

{ "_id": "6056c1107a34ec248448198d3435caf49",
  "product": "gitar",
  "total": 199.99,
  "customer": "Karan" }

{ "_id": "6056c1107a34ec248448198d3435caf4a",
  "product": "milk",
  "total": 11.33,
  "customer": "Mike" }

{ "_id": "6056c1107a34ec248448198d3435caf4b",
  "product": "pizza",
  "total": 8.5,
  "customer": "Binu" }

```

```

> db.purchase_orders.aggregate([{$match: {} },{$group: {_id: "$customer", total: {$sum: "$total"}}}])
{ "_id" : "Mike", "total" : 16.08 }
{ "_id" : "Binu", "total" : 8.5 }
{ "_id" : "Karan", "total" : 199.99 }
>

```

```

> db.purchase_orders.aggregate([{$match: {} },{$group: {_id: "$product", total: {$sum: "$total"}}}])
{ "_id" : "gitar", "total" : 199.99 }
{ "_id" : "milk", "total" : 11.33 }
{ "_id" : "tooth paste", "total" : 4.75 }
{ "_id" : "pizza", "total" : 8.5 }
>

```

purchase_orders.purchase_orders

DOCUMENTS 4 TOTAL SIZE 306B AVG. SIZE 77B INDEXES 1 TOTAL SIZE 20.0KB

Documents Aggregations Schema Explain Plan Indexes Validation

COLLATION Untitled SAVE SAMPLE MODE AUTO PREVIEW

4 Documents in the Collection Preview of Documents in the Collection

Select an operator to construct expressions used in the aggregation pipeline stages. [Learn more](#)

```

{ "_id": "6056c1107a34ec248448198d3435caf48",
  "product": "tooth paste",
  "total": 4.75,
  "customer": "Mike" }

{ "_id": "6056c1107a34ec248448198d3435caf49",
  "product": "gitar",
  "total": 199.99,
  "customer": "Karan" }

{ "_id": "6056c1107a34ec248448198d3435caf4a",
  "product": "milk",
  "total": 11.33,
  "customer": "Mike" }

{ "_id": "6056c1107a34ec248448198d3435caf4b",
  "product": "pizza",
  "total": 8.5,
  "customer": "Binu" }

```

