

Assignment 12 : mongoDB mapReduce

```
show dbs
use Asg10
```

1. Find the sum of the amount of each customer whose status is "P":

```
// Map function
var mapFunction = function () {
  if (this.status === "P") {
    emit(this.id, this.amount);
  }
};

// Reduce function
var reduceFunction = function (customerId, amounts) {
  return Array.sum(amounts);
};

// Run the map-reduce operation
db.orders.mapReduce(mapFunction, reduceFunction, { out:
"sumAmountOfCustomersWithStatusP" });

// Query the result
db.sumAmountOfCustomersWithStatusP.find();
```

Assignment 12 : mongoDB mapReduce

2. Find the average amount of each customer:

```
// Map function
var mapFunction = function () {
  emit(this.id, { count: 1, total: this.amount });
};

// Reduce function
var reduceFunction = function (customerId, values) {
  var result = { count: 0, total: 0 };
  values.forEach(function (value) {
    result.count += value.count;
    result.total += value.total;
  });
  return result;
};

// Finalize function (to calculate the average)
var finalizeFunction = function (customerId, result) {
  result.average = result.total / result.count;
  return result;
};

// Run the map-reduce operation
db.orders.mapReduce(mapFunction, reduceFunction, { out:
"averageAmountOfCustomers", finalize: finalizeFunction });

// Query the result
db.averageAmountOfCustomers.find();
```

Assignment 12 : mongoDB mapReduce

3.Find the minimum amount of each customer:

```
// Map function
var mapFunction = function () {
    emit(this.id, this.amount);
};

// Reduce function
var reduceFunction = function (customerId, amounts) {
    return Math.min.apply(null, amounts);
};

// Run the map-reduce operation
db.orders.mapReduce(mapFunction, reduceFunction, { out:
"minAmountOfCustomers" });

// Query the result
db.minAmountOfCustomers.find();
```

Assignment 12 : mongoDB mapReduce

4. Find the maximum amount of each customer whose status is "F":

```
// Map function
var mapFunction = function () {
  if (this.status === "F") {
    emit(this.id, this.amount);
  }
};

// Reduce function
var reduceFunction = function (customerId, amounts) {
  return Math.max.apply(null, amounts);
};

// Run the map-reduce operation
db.orders.mapReduce(mapFunction, reduceFunction, { out:
"maxAmountOfCustomersWithStatusF" });

// Query the result
db.maxAmountOfCustomersWithStatusF.find();
```