

Work 14

1. Text-Search

Step 1 新增資料

```
> db.stores.insertMany(
... [
... { _id:1, name: "Jave Hut", description:"Coffee and cakes"},
... { _id:2, name: "Burger Buns", description:"Gourmet hamburgers"},
... { _id:3, name: "Coffee Shop", description:"Just coffee"},
... { _id:4, name: "Clothes Clothes Clothes", description:"Discount clothing"},
... { _id:5, name: "Jave Shopping", description:"Indonesian goods"},
... { _id:6, name: "B10723059", description:"I am Tang zhiwei"},
... ]
... )
{ "acknowledged" : true, "insertedIds" : [ 1, 2, 3, 4, 5, 6 ] }
```

Step 2 建立 Index

```
> db.stores.createIndex({name:"text", decription:"text"})
{
  "createdCollectionAutomatically" : false,
  "numIndexesBefore" : 1,
  "numIndexesAfter" : 2,
  "ok" : 1
}
```

Step 3 搜尋學號、"coffee"，排除 "java"，回傳資料排序。

```
> db.stores.find({$text:{$search:"B10723059 coffee -java"}})
{ "_id" : 6, "name" : "B10723059", "description" : "I am Tang zhiwei" }
{ "_id" : 3, "name" : "Coffee Shop", "description" : "Just coffee" }
>
_
```

----- 下一頁 -----

2. Map-Reduce

Step 1 插入資料

```
2021-06-03T07:47:36.350+0800 I CONTROL [initandlisten]
> db.orders.insertMany([
...   { _id: 1, cust_id: "Ant O. Knee", ord_date: new Date("2020-03-01"), price: 25, items: [ { sku: "oranges", qty: 5, price: 2.5 }, { sku:
"apples", qty: 5, price: 2.5 } ], status: "A" },
...   { _id: 2, cust_id: "Ant O. Knee", ord_date: new Date("2020-03-08"), price: 70, items: [ { sku: "oranges", qty: 8, price: 2.5 }, { sku:
"chocolates", qty: 5, price: 10 } ], status: "A" },
...   { _id: 3, cust_id: "Busby Bee", ord_date: new Date("2020-03-08"), price: 50, items: [ { sku: "oranges", qty: 10, price: 2.5 }, { sku:
"pears", qty: 10, price: 2.5 } ], status: "A" },
...   { _id: 4, cust_id: "Busby Bee", ord_date: new Date("2020-03-18"), price: 25, items: [ { sku: "oranges", qty: 10, price: 2.5 } ], statu
s: "A" },
...   { _id: 5, cust_id: "Busby Bee", ord_date: new Date("2020-03-19"), price: 50, items: [ { sku: "chocolates", qty: 5, price: 10 } ], stat
us: "A" },
...   { _id: 6, cust_id: "Cam Elot", ord_date: new Date("2020-03-19"), price: 35, items: [ { sku: "carrots", qty: 10, price: 1.0 }, { sku: "
apples", qty: 10, price: 2.5 } ], status: "A" },
...   { _id: 7, cust_id: "Cam Elot", ord_date: new Date("2020-03-20"), price: 25, items: [ { sku: "oranges", qty: 10, price: 2.5 } ], status
: "A" },
...   { _id: 8, cust_id: "Don Quis", ord_date: new Date("2020-03-20"), price: 75, items: [ { sku: "chocolates", qty: 5, price: 10 }, { sku:
"apples", qty: 10, price: 2.5 } ], status: "A" },
...   { _id: 9, cust_id: "Don Quis", ord_date: new Date("2020-03-20"), price: 55, items: [ { sku: "carrots", qty: 5, price: 1.0 }, { sku: "a
pples", qty: 10, price: 2.5 }, { sku: "oranges", qty: 10, price: 2.5 } ], status: "A" },
...   { _id: 10, cust_id: "Don Quis", ord_date: new Date("2020-03-23"), price: 25, items: [ { sku: "oranges", qty: 10, price: 2.5 } ], statu
s: "A" }
... ])
{
  "acknowledged" : true,
  "insertedIds" : [
    1,
    2,
    3,
    4,
    5,
    6,
    7,
    8,
    9,
    10
  ]
}
```

Step 2 定義名為 mapFunction 的函式,功能為調用將要群組化的資料。

```
> var mapFunction = function(){
...   emit(this.cust_id, this.price);
... };
> mapFunction
function () {
  emit(this.cust_id, this.price);
}
```

Step 3 定義名為 reduceFunction 的函式,功能為將 cust_id 群組化，並計算出總額。

```
> var reduceFunction = function(keyCustId, valuesPrices){
...   return Array.sum(valuesPrices);
... };
```

Step 4 利用前倆函式來進行 mapReduce，並以 my_reduce 命名輸出。

```
> db.orders.mapReduce(
...   mapFunction,
...   reduceFunction,
...   {out:"my_reduce"}
... )
{
  "result" : "my_reduce",
  "timeMillis" : 50,
  "counts" : {
    "input" : 10,
    "emit" : 10,
    "reduce" : 4,
    "output" : 4
  },
  "ok" : 1
}
```

Step 5 結果

```
> db.my_reduce.find().sort({_id:1})
{ "_id" : "Ant O. Knee", "value" : 95 }
{ "_id" : "Busby Bee", "value" : 125 }
{ "_id" : "Cam Elot", "value" : 60 }
{ "_id" : "Don Quis", "value" : 155 }
>
—
```

Map-Reduce Explain:

進行資料群組化後，可以將同一組的資料進行運算，最後輸出成另一集合。

公式

```
db.<collection>.mapReduce(  
    <Mapfunction>,  
    <Recucefunction>,  
    <output>  
)
```