

# mongoDB® AGGREGATION

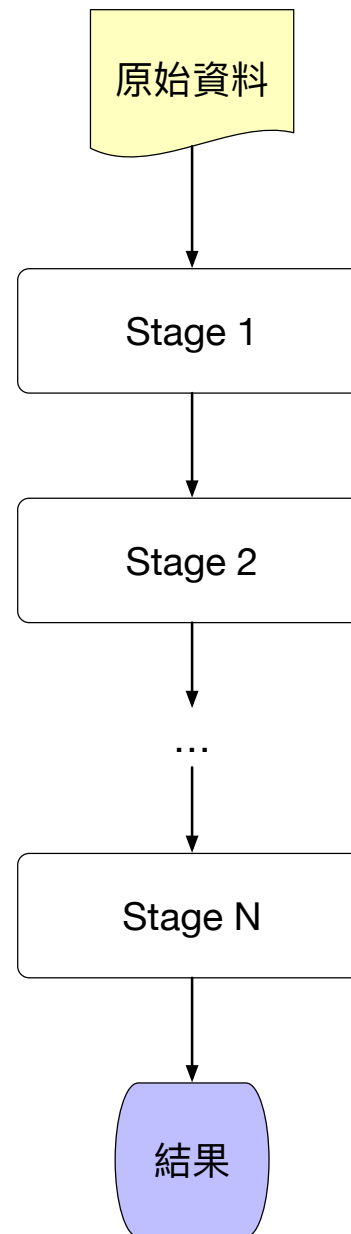
朱克剛

# PIPELINE

將資料經過各階段的處理產生最後要的結果

下一階段所需要的輸入就是上一階段的輸出

每個階段選擇適當的 **pipeline** 指令來處理資料



# PYTHON範本

```
import pymongo
from pprint import pprint

client = pymongo.MongoClient()
db = client.test

pipeline = []

cursor = db.test.aggregate(pipeline)
pprint(list(cursor))
```

# 列出 AQI 大於 50 的資料

`$addFields`內容

```
{
  iAQI: {
    $toInt: '$AQI'
  }
}
```

`$match`內容

```
{
  iAQI: {
    $gte: 50
  }
}
```

運算指令

`$toBool`

`$toDate`

`$toInt`

`$toDouble`

`$toString`

...

符號

`>`

`>=`

`<`

`<=`

`=`

`<>`

運算指令

`$gt`

`$gte`

`$lt`

`$lte`

`$eq`

`$ne`

# 群組

- ◆ 計算各縣市平均 AQI 指數，並排序
- ◆ 要群組的欄位放到 `_id` 裡面

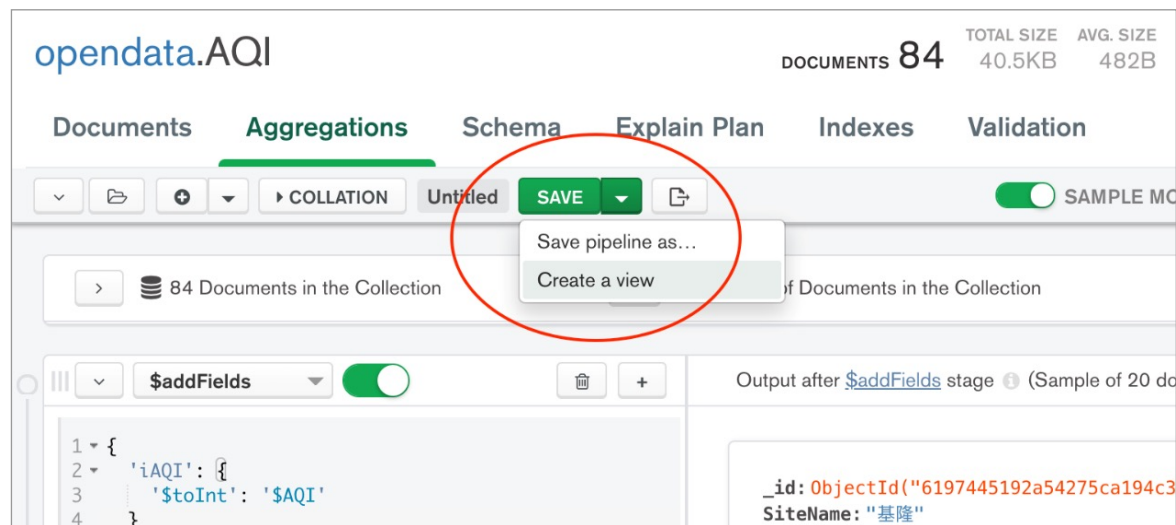
stage1	增加iAQI欄位，內容為整數型態的AQI指數
stage2	對County欄位群組，群組後計算iAQI平均值
stage3	列出step2的所有欄位，並針對平均值欄位四捨五入
stage4	針對平均值欄位做逆向排序

```
pipeline = [  
  {'$addField': {'AQI': {'$toInt': '$iAQI'}}},  
  {'$group': {'_id': '$County', 'avgAQI': {'$avg': '$iAQI'}}},  
  {'$project': {'avgAQI': {'$round': ['$avgAQI', 0]}}},  
  {'$sort': {'avgAQI': -1}}  
]
```

# VIEW

使用

- `db.name_of_view.find()`



# 群組後算筆數

- ◆ 計算每個群組中有多少筆資料
- ◆ 由於 `$count` 無法使用於 `group`，所以算群組中的資料筆數參考以下作法

```
pipeline = [  
  { '$group':  
    {  
      '_id': '$County',  
      'count': { '$sum': 1 }  
    }  
  }  
]
```

# 條件

Pipeline 指令為 \$match

◆查詢 AQI 指數大於 50 的縣市

```
[
  {
    '$addFields': {
      'iAQI': {
        '$toInt': '$AQI'
      }
    }
  }, {
    '$match': {
      'iAQI': {
        '$gte': 50
      }
    }
  }
]
```



# \$LOOKUP：相當於 LEFT JOIN

## ◆列出AQI指數最高的地點

```
pipeline = [  
  {'$addField': {'AQI': {'$toInt': '$iAQI'}}},  
  {'$group': {'_id': None, 'maxAQI': {'$max': '$iAQI'}}},  
  {'$project': {'maxAQI': {'$toString': '$maxAQI'}}},  
  {'$lookup': {  
    'from': 'AQI',  
    'localField': 'maxAQI',  
    'foreignField': 'iAQI',  
    'as': 'result'  
  }}  
]
```

結果放在 result  
陣列中

# 練習

請列出監測站最多的縣市

# GEOJSON 相關資料

<https://zh.wikipedia.org/zh-tw/GeoJSON>

<https://geojson.org>

<https://geojson.io>

# GEOJSON 格式轉換

\$addFields內容

```
{  
  geometry: {  
    type: 'Point',  
    coordinates: [  
      { $toDouble: '$Longitude' },  
      { $toDouble: '$Latitude' }  
    ]  
  },  
  iAQI: { $toInt: '$AQI' }  
}
```

下一個stage用  
\$out 輸出到另  
外一個資料表

geometry欄位  
要設定2D  
sphere索引

# 查詢五公里範圍內的資料

\$geoNear內容

```
{
  near: {
    type: 'Point',
    coordinates: [121.5466, 25.15532]
  },
  distanceField: 'distance',
  maxDistance: 5000,
  includeLocs: 'geometry',
  query: {
    County: { $in: ['臺北市', '新北市'] }
  }
}
```

單位：公尺

query 可省略



# 練習

請根據使用者所在位置，查出該地區的 AQI 值

提示：會使用 `$limit stage`