

DBMS HW6 Report

許致銓 R10625016

1. MongoDB

a. Task1

```
~/De/2024/DB/2024-Spring-Database-Management-System-From-SQL-to-NoSQL/I/HW6 main mongoimport --type=csv --db=hw6
--collection=students --headerline --file='Hw6_file/studentE.csv'
2024-06-04T17:17:51.879+0800 connected to: mongodb://localhost/
2024-06-04T17:17:51.940+0800 103 document(s) imported successfully. 0 document(s) failed to import.

> use hw6
< switched to db hw6
> db.students.findOne({ name: { $regex: /^許致銓/i } });
< {
  _id: 'r10625016',
  position: '學生',
  dept: '森林環資所',
  year: 3,
  name: '許致銓 (CHIH-CHUAN HSU)',
  email: 'r10625016@ntu.edu.tw',
  class: '資料庫系統-從SQL到NoSQL (EE5178)'
}
```

b. Task2

```

> db.students.aggregate([
  { $match: { position: "學生" } }, // Adjust the field and value according to your data schema
  { $group: {
    _id: "$dept", // Group by the 'dept' field
    numberOfStudents: { $sum: 1 } // Count the number of students in each group
  }},
  { $sort: { numberOfStudents: -1 } }, // Sort the groups by 'numberOfStudents' descending
  { $limit: 10 } // Limit to top 10
]);
< {
  _id: '資工所',
  numberOfStudents: 14
}
{
  _id: '電機所',
  numberOfStudents: 11
}
{
  _id: '生物機電所',
  numberOfStudents: 8
}
{
  _id: '經濟系',
  numberOfStudents: 6
}
{
  _id: '電機系',
  numberOfStudents: 6
}
{
  _id: '資訊管理所',
  numberOfStudents: 5
}

```

c. Task3

```

> db.students.updateMany(
  {}, // Filter: empty means apply to all documents
  { $set: { updated: ISODate("2024-05-23") } } // Action: set the 'updated' field
);
< {
  acknowledged: true,
  insertedId: null,
  matchedCount: 103,
  modifiedCount: 103,
  upsertedCount: 0
}

```

d. Task4

```

~/De/2024/DB/2024-Spring-Database-Management-System-From-SQL-to-NoSQL/HW6 main !1 mongoimport --type=csv --db=h
w6 --collection=students --headerline --file='HW6_file/new_studentE.csv'
2024-06-04T17:37:07.764+0800 connected to: mongodb://localhost/
2024-06-04T17:37:07.782+0800 3 document(s) imported successfully. 0 document(s) failed to import.

```

```

> db.students.updateMany(
  { dept: {$regex: /^太空|森林/i } }, // Match the new students
  { $set: { updated: ISODate("2024-06-01") } }
);
< {
  acknowledged: true,
  insertedId: null,
  matchedCount: 4,
  modifiedCount: 3,
  upsertedCount: 0
}

```

(不好意思，在執行時已經執行過一次未 import new_studentE.csv 時的 update，所以 modifiedCount 在這邊只有 3)

e. Task5

```

> db.students.aggregate([
  { $match: {
    updated: {
      $gte: ISODate("2024-01-01T00:00:00.000Z"),
      $lte: ISODate("2024-05-31T23:59:59.999Z")
    }
  }},
  { $group: {
    _id: "$dept",
    count: { $sum: 1 }
  }},
  { $out: "tally" }
]);
<
> db.tally.find();
< {
  _id: '動科系',
  count: 2
}
{
  _id: '資訊管理所',
  count: 5
}
{
  _id: '資工系',
  count: 5
}
{
  _id: '工科海洋所',
  count: 3
}
{

```

```

> db.students.aggregate([
  { $match: {
    updated: {
      $gte: ISODate("2024-06-01T00:00:00.000Z"),
      $lte: ISODate("2024-06-30T23:59:59.999Z")
    }
  }},
  { $group: {
    _id: "$dept",
    count: { $sum: 1 }
  }},
  { $out: "tally" }
]);
<
> db.tally.find();
< {
  _id: '森林環資所',
  count: 1
}
{
  _id: '太空所',
  count: 2
}
{
  _id: '太空系',
  count: 1
}
hw6 >

```

f. Task6

```

~/De/2024/DB/2024-Spring-Database-Management-System-From-SQL-to-NoSQL/i/HW6 main !1 72 mongoimport --type=csv --d
b=hw6 --collection=students --headerline --file='Hw6_file/student_groupE.csv' --mode=merge --upsertFields _id;
2024-06-04T17:42:19.881+0800 connected to: mongodb://localhost/
2024-06-04T17:42:19.910+0800 93 document(s) imported successfully. 0 document(s) failed to import.

```

	<pre> _id: "b09507021" position: "學生" dept: "材料系" year: 4 name: "王禹翔 (WANG YU-HSIANG)" email: "b09507021@ntu.edu.tw" class: "資料庫系統-從SQL到NoSQL (EE5178)" updated: 2024-05-23T00:00:00.000+00:00 group: 10 </pre>	
▶	<pre> _id: "r10521707" position: "學生" dept: "土木所營管組" year: 3 name: "林志瑋 (ZHI-XUAN, LIN)" email: "r10521707@ntu.edu.tw" class: "資料庫系統-從SQL到NoSQL (EE5178)" updated: 2024-05-23T00:00:00.000+00:00 group: 19 </pre>	<div>r10521707@ntu.edu.tw</div> <div>✎ 📎 📧 🗑</div>
	<pre> _id: "b09502132" position: "學生" dept: "機械系" year: 4 name: "周哲瑋 (CHOU, CHE-WEI)" email: "b09502132@ntu.edu.tw" class: "資料庫系統-從SQL到NoSQL (EE5178)" updated: 2024-05-23T00:00:00.000+00:00 group: 7 </pre>	

g. Task7

```

> db.students.aggregate([
  { $match: { "group": { $exists: true, $ne: null } } }, // Filters out documents without a group number
  {
    $group: {
      _id: "$group", // Group by group number
      members: { $push: "$name" } // Collects all member names into an array
    }
  }
]);
< {
  _id: 7,
  members: [
    '周哲瑋 (CHOU, CHE-WEI)',
    '王偉力 (WANG, WEI-LI)',
    '吳吉加 (CHI-CHIA WU)',
    '鄧雅文 (TENG, YA-WEN)'
  ]
}
{
  _id: 11,
  members: [
    '邱泓翊 (CHIU, hung-yi)',
    '郭子敬 (TZU-CHING KUO)',
    '陳品文 (CHEN, PIN-WEN)',
    '蔡予恩 ( TSAI, JONAS)'
  ]
}

```

2. Neo4j

a. Task1

```

1 LOAD CSV WITH HEADERS FROM 'file:///Hw6_file/studentE.csv' AS row
2 CREATE (s:Student {
3   id: row.id,
4   position: row.position,
5   dept: row.dept,
6   year: toInteger(row.year),
7   name: row.name,
8   email: row.email,
9   class: row.class
10 });

```

Added 103 labels, created 103 nodes, set 719 properties, completed after 82 ms.

b. Task2

```

1 LOAD CSV WITH HEADERS FROM 'file:///Hw6_file/student_groupE.csv' AS row
2 MERGE (g:Group {id: row.group}) // Ensure each group is created uniquely
3 WITH row, g
4 MATCH (s:Student {id: row.id}) // Find the student node that matches the student_id
5 MERGE (s)-[:BELONGS_TO]-(g) // Create a relationship from the student to the group
6 RETURN s, g;

```

Overview

Node labels

Student (93) Group (24)

Relationship types

BELONGS_TO (93)

Displaying 117 nodes, 0 relationships.

c. Task3

```

1 MATCH (you:Student)-[:BELONGS_TO]-(g:Group)-[:BELONGS_TO]-(groupMate:Student)
2 WHERE you.name =~ '^許致銓.*'
3 RETURN you.name AS YourName, collect(groupMate.name) AS GroupPartners;

```

YourName	GroupPartners
"許致銓 (CHIH-CHUAN HSU)"	["吳建翰 (WU, JIAN-HAN)", "林郁敏 (LIN, YU-MIN)", "陳冠霖 (CHEN, GUAN-LIN)"]

d. Task4

```

1 MATCH (s:Student)
2 MERGE (d:Department {name: coalesce(s.dept, 'NA')})
3 MERGE (s)-[:ENROLLED_IN]-(d)
4 RETURN s, d;

```

Node properties

Department

<elementid> 4:d98d21fc-2275-423c-87e7-04133801bb9a:151

<id> 151

name 資工所

e. Task5

```
1 MATCH (s:Student)
2 WHERE s.position = '學生' AND (s.position <> '旁聽生' OR s.position IS NULL)
3 WITH s.dept AS Department, coalesce(s.name, 'NA') AS StudentName
4 RETURN Department, count(StudentName) AS NumberOfStudents
5 ORDER BY NumberOfStudents DESC
6 LIMIT 10;
```

	Department	NumberOfStudents
1	"資工所"	14
2	"電機所"	11
3	"生物機電所"	8
4	"經濟系"	6
5	"電機系"	6
6	"資訊管理系"	5

f. Task6

```
1 MATCH (g:Group)←[:BELONGS_TO]-(s:Student)-[:ENROLLED_IN]→(d:Department)
2 WITH g, count(DISTINCT d.name) AS deptCount, collect(DISTINCT d.name) AS deptNames
3 ORDER BY deptCount DESC, g.id ASC
4 RETURN g.id AS GroupNumber, deptNames, deptCount
5 LIMIT 5;
```

	GroupNumber	deptNames	deptCount
1	"10"	["材料所應材班", "材料所", "資工所", "材料系"]	4
2	"17"	["電機系", "機械所園力組", "經濟系", "機械所製造組"]	4
3	"9"	["NA", "森林環資所", "資工所", "電機系"]	4
4	"12"	["經濟系", "土木所營管組", "電信所"]	3
5	"13"	["經濟系", "NA", "資工所"]	3

g. Task7

```
1 MATCH (you:Student {dept: '森林環資所'})-[:BELONGS_TO]→(group:Group)←[:BELONGS_TO]-(partner:Student)
2 MATCH (partner)-[:ENROLLED_IN]→(dept:Department)
3 MATCH (student:Student)-[:ENROLLED_IN]→(dept)
4 WITH partner.name AS partner_name, dept.name AS dept_name, count(student) AS dept_size
5 RETURN partner_name, dept_name, dept_size
6 ORDER BY partner_name, dept_name;
```

	partner_name	dept_name	dept_size
1	"吳建翰 (WU, JIAN-HAN)"	"NA"	2
2	"林郁敏 (LIN, YU-MIN)"	"資工所"	15
3	"陳冠霖 (CHEN, GUAN-LIN)"	"電機系"	8

Started streaming 3 records after 23 ms and completed after 28 ms.