

## 6. Bài Tập Tuần 06

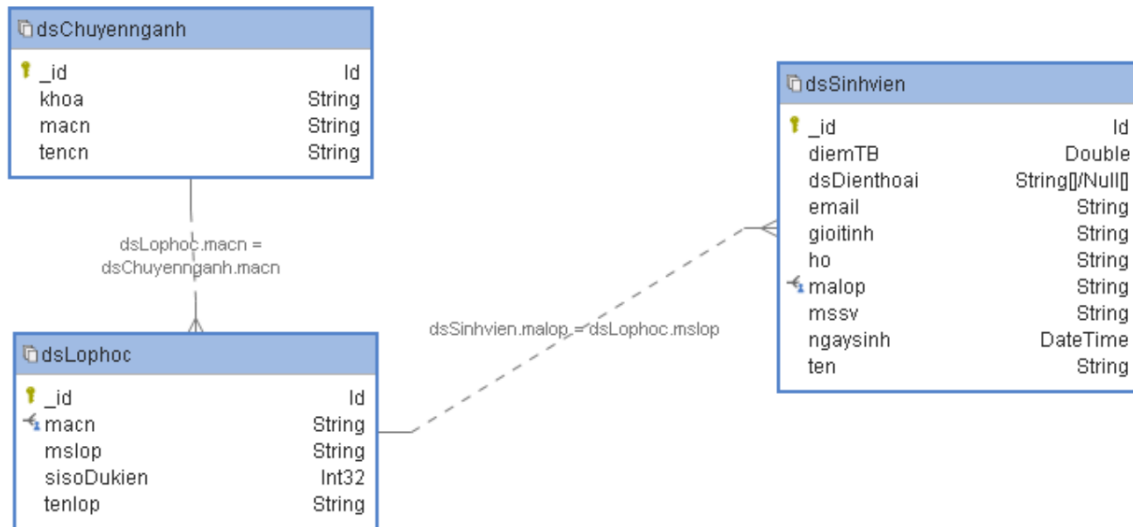
**6.1 Import collection zipcodes.json, tìm hiểu ý nghĩa của document sau đó thực hiện yêu cầu:**

```
{
  _id : "01001",
  city : "AGAWAM",
  loc : [ -72.622739, 42.070206 ],
  pop : 15338,
  state : "MA"
}
```

1. Hiển thị  $n$  documents từ document thứ  $k$ . ( $n, k$  tùy ý).
2. Tìm dân số của thành phố (*city*) PALMER, hiển thị city và dân số (*pop*).
3. Tìm các document có dân số  $>70,000$ .
4. Tìm danh sách tất cả các bang (*state*) không trùng.
5. Tìm tất cả các thành phố của bang MA có dân số trên 500.
6. Tìm tất cả các bang có ít nhất một thành phố có dân số trên 65,000.
7. Tính tổng dân số theo từng bang.
8. Tìm bang có tổng dân số lớn nhất.
9. Tìm tất cả các bang có tổng dân số trên 10,000,000.
10. Tính dân số trung bình của các thành phố theo từng bang.
11. Tìm số lượng thành phố duy nhất của bang “WA”.
12. \*Tính số thành phố của mỗi bang (*không trùng*), sắp xếp giảm dần theo số lượng.
13. \*Tìm thành phố có dân số lớn nhất và nhỏ nhất.
14. \*\*Tìm thành phố có tổng dân số (*pop*) lớn nhất trong mỗi bang.
15. \*\*\*Xuất danh sách những document có dân số nhỏ hơn dân số trung bình của mỗi bang.
16. \*\*\*Phân loại các thành phố theo dân số (*sử dụng \$bucket*):
  - Chia nhỏ thành phố theo 3 nhóm:
    - i. Nhỏ (  $< 10,000$  dân );
    - ii. Trung bình: (  $10,000 - 100,000$  dân );
    - iii. Lớn (  $> 100,000$  dân )

## 6.2 Query (dữ liệu sinhviendb3)

Thông tin sinh viên gồm: Mã số sinh viên, họ, tên, giới tính, ngày sinh, email, các số điện thoại và điểm trung bình.



1. Đếm số sinh viên thực tế theo từng lớp học.
2. Liệt kê danh sách sinh viên không có số điện thoại hoặc email.
3. Liệt kê danh sách sinh viên có từ 2 số điện thoại trở lên.
4. Tìm lớp học có tổng số sinh viên thực tế cao nhất
5. Đếm số sinh viên có điểm trung bình từ 9.0 trở lên theo từng lớp.
6. Xuất danh sách sinh viên giỏi ( $diemTB \geq 9.0$ ) theo từng lớp ra collection `dsSinhVienGioi`.
7. Tính tổng số sinh viên theo từng chuyên ngành.
8. Tính điểm trung bình theo từng lớp, sắp xếp giảm dần.
9. Tìm sinh viên có điểm cao nhất trong từng lớp.
10. \*Tìm lớp học có số sinh viên nhiều hơn `sisoDukien`.
11. \*\*Tiếp theo câu số 6, bổ sung thêm thông tin tên chuyên ngành vào collections `dsSinhVienGioi`.
  1. Nếu sinh viên đã tồn tại thì update thêm thông tin;
  2. Nếu sinh viên chưa tồn tại thì thực hiện thêm mới dữ liệu.