# Đề bài

1. Liệt kê danh sách n sinh viên
2. Tìm kiếm sinh viên khi biết mã số
3. Tìm danh sách khóa học thuộc của một khoa nào đó khi biết mã khoa
4. Liệt kê tất cả các khoa
5. Liệt kê tên của tất cả các trưởng khoa
6. Tìm tên của trưởng khoa CS
7. Liệt kê tất cả các khóa học của CS và IE
8. Liệt kê danh sách các tên của các sinh viên đăng ký học khóa học CS101
9. Tổng số sinh viên đăng ký học của mỗi khoa
10. Tổng số sinh viên đăng ký học của mỗi khoa, kết quả sắp xếp theo mã khoa
11. Tổng số sinh viên đăng ký học của mỗi khoa, kết quả sắp xếp theo số sinh viên
12. Liệt kê danh sách tên của các trưởng khoa mà các khoa này không có sinh viên đăng ký học
13. Danh sách khoa có số sinh viên đăng ký học nhiều nhất
14. Danh sách sinh viên có gpa >= 3.2, kết quả sắp xếp giảm dần theo gpa

# Đáp án

// 1.   Liệt kê danh sách n sinh viên

MATCH(n:Student) return n

// 2.   Tìm kiếm sinh viên khi biết mã số

MATCH (n:Student)

where n.student\_id = "22"

return n

// 3.   Tìm danh sách khóa học thuộc của một khoa nào đó khi biết mã khoa

MATCH (n:Course)-[r:BELONGS\_TO]->(m:Department)

WHERE m.dept\_id ="Math"

RETURN \*

// 4.   Liệt kê tất cả các khoa

match(n:Department)

return n

// 5.   Liệt kê tên của tất cả các trưởng khoa

MATCH (n:Department)

RETURN n.dean

// 6.   Tìm tên của trưởng khoa CS

MATCH (n:Department)

WHERE n.dept\_id = "CS"

RETURN n.dean

// 7.   Liệt kê tất cả các khóa học của CS và IE

MATCH (m:Course)-[r:BELONGS\_TO]->(n:Department)

WHERE n.dept\_id = "CS" or n.dept\_id = "IE"

RETURN \*

// 8.   Liệt kê danh sách các tên của các sinh viên đăng ký học khóa học CS101

MATCH (n:Student)-[r:ENROLLED]->(m:Course)

WHERE m.course\_id = "CS101"

RETURN \*

// 9.   Tổng số sinh viên đăng ký học của mỗi khoa

MATCH (s:Student)-[e:ENROLLED]->(c:Course)-[b:BELONGS\_TO]->(d:Department)

RETURN d.name, count(s) as total\_students

// 10.  Tổng số sinh viên đăng ký học của mỗi khoa, kết quả sắp xếp theo mã khoa

MATCH (s:Student)-[e:ENROLLED]->(c:Course)-[b:BELONGS\_TO]->(d:Department)

RETURN d.dept\_id, count(s) as total\_students

ORDER BY d.dept\_id

// 11.  Tổng số sinh viên đăng ký học của mỗi khoa, kết quả sắp xếp theo số sinh viên

MATCH (s:Student)-[e:ENROLLED]->(c:Course)-[b:BELONGS\_TO]->(d:Department)

RETURN d.dept\_id, count(s) as total\_students

ORDER BY total\_students

// 12.  Liệt kê danh sách tên của các trưởng khoa mà các khoa này không có sinh viên đăng ký học

MATCH (d:Department)

WHERE NOT EXISTS {match (s:Student)-[e:ENROLLED]->(c:Course)-[b:BELONGS\_TO]->(d)}

RETURN d.dean

// 13.  Danh sách khoa có số sinh viên đăng ký học nhiều nhất

match (n:Student)-[r:ENROLLED]->(c:Course)-[r2:BELONGS\_TO]->(d:Department)

WITH d.dept\_id as deparmentId, count(n) as total\_students

WITH COLLECT(deparmentId) as deparmentsId, total\_students

ORDER BY total\_students DESC

LIMIT 1

UNWIND deparmentsId as deparmentId

RETURN deparmentId

// 15.  Danh sách sinh viên có gpa >= 3.2, kết quả sắp xếp giảm dần theo gpa

MATCH (n:Student)

WHERE n.gpa >= 3.2

RETURN n

ORDER BY n.gpa desc