



**TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI**  
HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY

# Bài tập lớn Cơ sở dữ liệu Education

Nhóm 4

Viện Toán ứng dụng và Tin học

Ngày 20 tháng 5 năm 2021

Advisor: Nguyễn Danh Tú

# Nội dung chính trình bày

Thông tin về cơ  
sở dữ liệu

Truy vấn dữ liệu

Cập nhật dữ liệu

## 1 Thông tin về cơ sở dữ liệu

- Diagram
- Thông tin và số lượng các bảng dữ liệu
- Đánh giá mức độ chuẩn hóa

## 2 Truy vấn dữ liệu

- Các câu lệnh truy vấn
- Tối ưu thời gian truy vấn
- Đánh giá hiệu năng

## 3 Cập nhật dữ liệu

- Transaction
- Tạo một CSDL y hệt cấu trúc



# Các thành viên trong nhóm

## Thông tin về cơ sở dữ liệu

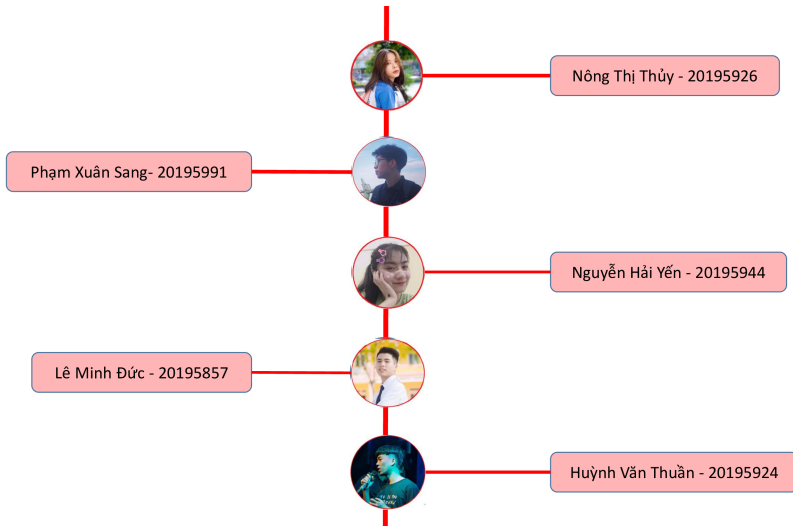
Diagram

Thông tin và số lượng các bảng dữ liệu

Đánh giá mức độ chuẩn hóa

## Truy vấn dữ liệu

## Cập nhật dữ liệu



### Thông tin về cơ sở dữ liệu

Diagram

Thông tin và số lượng các bảng dữ liệu

Đánh giá mức độ chuẩn hóa

Truy vấn dữ liệu

Cập nhật dữ liệu

### Education là gì?



### Thông tin về cơ sở dữ liệu

Diagram

Thông tin và số lượng các bảng dữ liệu

Đánh giá mức độ chuẩn hóa

Truy vấn dữ liệu

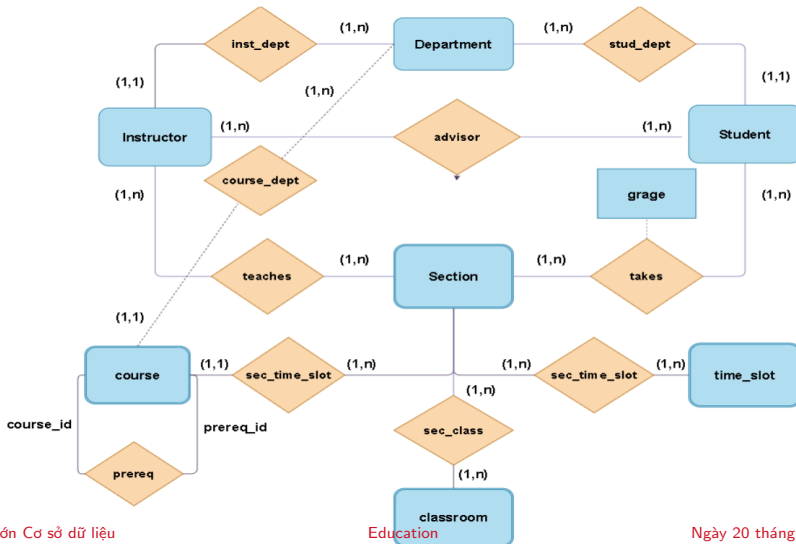
Cập nhật dữ liệu

Nguồn: db-book.com





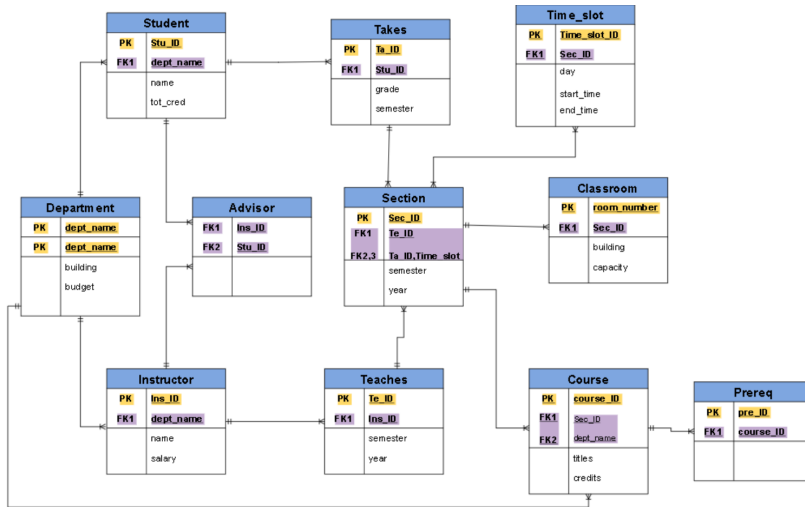
## Mô hình ER



# Thông tin về cơ sở dữ liệu

## R-E Diagram

### Mô hình RE



Thông tin về cơ  
sở dữ liệu

Diagram

Thông tin và số lượng  
các bảng dữ liệu

Đánh giá mức độ  
chuẩn hóa

Truy vấn dữ liệu

Cập nhật dữ liệu



# Thông tin về cơ sở dữ liệu

## Thông tin và số lượng các bảng dữ liệu

### Thông tin về cơ sở dữ liệu

Diagram

Thông tin và số lượng các bảng dữ liệu

Đánh giá mức độ chuẩn hóa

### Truy vấn dữ liệu

### Cập nhật dữ liệu

| Name       | Engine | Version | Row Format | Rows    | Avg Row Length | Data Length |
|------------|--------|---------|------------|---------|----------------|-------------|
| advisor    | InnoDB | 10      | Dynamic    | 232742  | 115            | 25.5 MiB    |
| classroom  | InnoDB | 10      | Dynamic    | 12000   | 132            | 1.5 MiB     |
| course     | InnoDB | 10      | Dynamic    | 200     | 81             | 16.0 KiB    |
| department | InnoDB | 10      | Dynamic    | 20      | 819            | 16.0 KiB    |
| instructor | InnoDB | 10      | Dynamic    | 199141  | 55             | 10.5 MiB    |
| prereq     | InnoDB | 10      | Dynamic    | 100     | 163            | 16.0 KiB    |
| section    | InnoDB | 10      | Dynamic    | 316957  | 61             | 18.5 MiB    |
| student    | InnoDB | 10      | Dynamic    | 1199738 | 53             | 61.1 MiB    |
| takes      | InnoDB | 10      | Dynamic    | 2993381 | 58             | 165.9 MiB   |
| teaches    | InnoDB | 10      | Dynamic    | 498928  | 53             | 25.6 MiB    |
| time_slot  | InnoDB | 10      | Dynamic    | 20      | 819            | 16.0 KiB    |

Table Count:  
Database Size:

**11**  
**354.9 MiB**





# Đánh giá mức độ chuẩn hóa

## Thông tin về cơ sở dữ liệu

Diagram

Thông tin và số lượng các bảng dữ liệu

Đánh giá mức độ chuẩn hóa

## Truy vấn dữ liệu

## Cập nhật dữ liệu



- Tất cả các bảng đều đạt chuẩn **BCNF (Boyce–Codd normal form)**.

## Chứng minh

Xét quan hệ (bảng) **section** có:

- Phụ thuộc hàm (course\_id, sec\_id, semester, year) → (building, room\_number, time\_slot\_id).
- Các thuộc tính khóa: course\_id, sec\_id, semester, year.
- Các thuộc tính không khóa: building, room\_number, time\_slot\_id.
- Dễ dàng thấy quan hệ **section** đạt chuẩn **BCNF**.

# Truy vấn dữ liệu

## Các câu lệnh truy vấn

Thông tin về cơ sở dữ liệu

Truy vấn dữ liệu

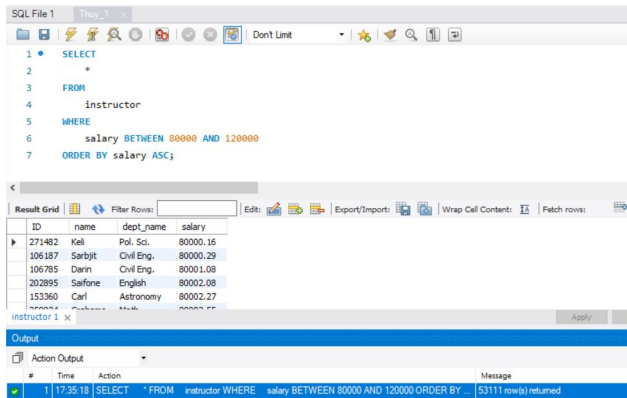
Các câu lệnh truy vấn

Tối ưu thời gian truy vấn

Đánh giá hiệu năng

Cập nhật dữ liệu

1. Sắp xếp danh sách người hướng dẫn có lương từ 80000 đến 120000 theo thứ tự lương tăng dần.



The screenshot shows the SQL Developer interface. The top pane displays the following SQL query:

```
1 SELECT
2 *
3 FROM
4 instructor
5 WHERE
6 salary BETWEEN 80000 AND 120000
7 ORDER BY salary ASC;
```

The bottom pane shows the 'Result Grid' with the following data:

| ID     | name    | dept_name  | salary   |
|--------|---------|------------|----------|
| 271482 | Keli    | Pol. Sci.  | 80000.16 |
| 106187 | Sarbjit | Civil Eng. | 80000.29 |
| 106785 | Darin   | Civil Eng. | 80001.08 |
| 202895 | Salfone | English    | 80002.08 |
| 153360 | Carl    | Astronomy  | 80002.27 |
| 80004  | Carl    | Math       | 80002.55 |

Below the result grid, the 'Output' pane shows the 'Action Output' table:

| # | Time     | Action  | Message               |
|---|----------|---|-----------------------|
| 1 | 17:35:18 | SELECT * FROM instructor WHERE salary BETWEEN 80000 AND 120000 ORDER BY ... | 53111 row(s) returned |

# Truy vấn dữ liệu

## Các câu lệnh truy vấn

Thông tin về cơ sở dữ liệu

Truy vấn dữ liệu

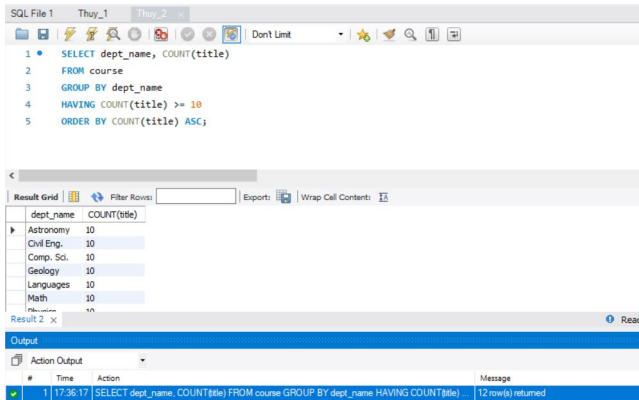
Các câu lệnh truy vấn

Tối ưu thời gian truy vấn

Đánh giá hiệu năng

Cập nhật dữ liệu

2. Lấy ra thông tin những phòng ban có số tiêu đề  $\geq 10$  sắp xếp theo thứ tự tăng dần của tiêu đề.



The screenshot shows the SQL Developer interface with a query window titled 'Thuy\_2'. The query is as follows:

```
1 SELECT dept_name, COUNT(title)
2 FROM course
3 GROUP BY dept_name
4 HAVING COUNT(title) >= 10
5 ORDER BY COUNT(title) ASC;
```

Below the query window, the 'Result Grid' displays the following data:

| dept_name  | COUNT(title) |
|------------|--------------|
| Astronomy  | 10           |
| Civil Eng. | 10           |
| Comp. Sci. | 10           |
| Geology    | 10           |
| Languages  | 10           |
| Math       | 10           |
| Physics    | 10           |

The 'Output' window shows the execution message:

| # | Time     | Action  | Message            |
|---|----------|---|--------------------|
| 1 | 17:36:17 | SELECT dept_name, COUNT(title) FROM course GROUP BY dept_name HAVING COUNT(title) ... | 12 row(s) returned |

Thông tin về cơ sở dữ liệu

Truy vấn dữ liệu

Các câu lệnh truy vấn

Tối ưu thời gian truy vấn

Đánh giá hiệu năng

Cập nhật dữ liệu



### 3. Thống kê ngân sách các tòa nhà của phòng ban.

SQL File 1 Thuy\_1 Thuy\_2 Thuy\_3 x

Don't Limit

```

1 SELECT
2     building,
3     SUM(budget) TongNgansach,
4     AVG(budget) TrungbinhNgansach,
5     MAX(budget) NgansachLonnhath,
6     MIN(budget) NgansachNhonhat
7 FROM
8     department
9 GROUP BY building;

```

Result Grid

| building    | TongNgansach | TrungbinhNgansach | NgansachLonnhath | NgansachNhonhat |
|-------------|--------------|-------------------|------------------|-----------------|
| Saucon      | 441840.92    | 441840.920000     | 441840.92        | 441840.92       |
| Taylor      | 1711446.54   | 570482.180000     | 699140.86        | 395051.74       |
| Bronfman    | 734550.70    | 734550.700000     | 734550.70        | 734550.70       |
| Candlestick | 1514442.30   | 757221.150000     | 866831.75        | 647610.55       |
| Chandler    | 255041.46    | 255041.460000     | 255041.46        | 255041.46       |
| Lamberton   | 106378.69    | 106378.690000     | 106378.69        | 106378.69       |

Result 2 x

Output

Action Output

| # | Time     | Action   | Message            |
|---|----------|--|--------------------|
| 1 | 17:37:11 | SELECT building, SUM(budget) TongNgansach, AVG(budget) TrungbinhNgansach, M... | 16 row(s) returned |

# Truy vấn dữ liệu

## Các câu lệnh truy vấn

Thông tin về cơ sở dữ liệu

Truy vấn dữ liệu  
Các câu lệnh truy vấn

Tối ưu thời gian truy vấn

Đánh giá hiệu năng


Cập nhật dữ liệu



4. Tìm tên của tất cả những người hướng dẫn có mức lương lớn hơn ít nhất một giáo viên hướng dẫn trong bộ môn Sinh học.

```
Thuy_5 Thuy_4 x
SELECT DISTINCT
  T.name
FROM
  instructor AS T,
  instructor AS S
WHERE
  T.salary > S.salary
  AND S.dept_name = 'Biology';
```

Result Grid

Filter Rows:  Export:  Wrap Cell Content: ☐

| name       |
|------------|
| Bawa       |
| Yazdi      |
| Wieland    |
| D'Agostino |
| Liley      |
| Atanassov  |
| Moreira    |
| Gustafsson |

Result 2 x [Read](#)

Output

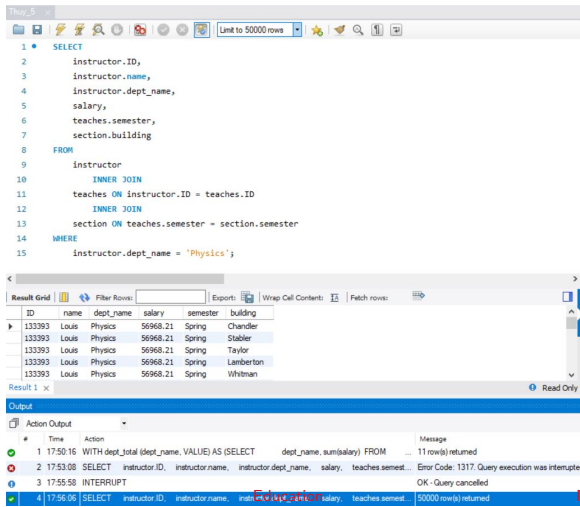
Action Output

| # | Time     | Action  | Message            |
|---|----------|---|--------------------|
| 1 | 17:57:47 | SELECT DISTINCT T.name FROM instructor AS T, instructor AS S WHERE T.salary > S.salary AND S.dept_name = 'Biology'; | 42 row(s) returned |

# Truy vấn dữ liệu

## Các câu lệnh truy vấn

5. Hiển thị thông tin gồm: ID, tên người hướng dẫn, tiền lương, học kỳ, tên tòa nhà với tên phòng ban là Vật lý.



The screenshot shows a SQL query in the 'Thay\_5' window. The query is a SELECT statement that joins the 'instructor', 'teaches', and 'section' tables to retrieve information about instructors in the Physics department. The query is as follows:

```
1 SELECT
2     instructor.ID,
3     instructor.name,
4     instructor.dept_name,
5     salary,
6     teaches.semester,
7     section.building
8 FROM
9     instructor
10    INNER JOIN
11    teaches ON instructor.ID = teaches.ID
12    INNER JOIN
13    section ON teaches.semester = section.semester
14 WHERE
15     instructor.dept_name = 'Physics';
```

Below the query, the 'Result Grid' shows the results of the query. The results are as follows:

| ID     | name  | dept_name | salary   | semester | building  |
|--------|-------|-----------|----------|----------|-----------|
| 133393 | Louis | Physics   | 56968.21 | Spring   | Chandler  |
| 133393 | Louis | Physics   | 56968.21 | Spring   | Stabler   |
| 133393 | Louis | Physics   | 56968.21 | Spring   | Taylor    |
| 133393 | Louis | Physics   | 56968.21 | Spring   | Lamberton |
| 133393 | Louis | Physics   | 56968.21 | Spring   | Whitman   |

The 'Output' window shows the execution of the query. The first row shows the query was executed successfully, returning 11 row(s). The second row shows the query was interrupted, with an error message: 'Error Code: 1317. Query execution was interrupted'. The third row shows the query was cancelled, with an error message: 'OK - Query cancelled'. The fourth row shows the query was executed successfully, returning 50000 row(s).

Thông tin về cơ sở dữ liệu

Truy vấn dữ liệu  
Các câu lệnh truy vấn

Tối ưu thời gian truy vấn

Đánh giá hiệu năng

Cập nhật dữ liệu



6. Tìm phòng ban có mức lương trung bình cao nhất.

```
SQL File 1  Thuy_1  Thuy_2  Thuy_3  Thuy_4  Thuy_5 x
[Icons] Dont Limit [Icons]
1 SELECT
2   dept_name
3 FROM
4   instructor
5 GROUP BY dept_name
6 HAVING AVG(salary) >= ALL (SELECT
7   AVG(salary)
8   FROM
9     instructor
10  GROUP BY dept_name);
```

Result Grid [Icons] Filter Rows: [ ] Export: [ ] Wrap Cell Content: [ ]

| dept_name |            |
|-----------|------------|
| ▶         | Accounting |

```
instructor 2 x [ ] Read
```

Output

Action Output [ ]

| #   | Time     | Action  | Message           |
|-----|----------|---|-------------------|
| ✓ 1 | 17:38:47 | SELECT dept_name FROM instructor GROUP BY dept_name HAVING AVG(salary) >= ALL | 1 row(s) returned |

# Truy vấn dữ liệu

## Các câu lệnh truy vấn

Thông tin về cơ sở dữ liệu

Truy vấn dữ liệu  
Các câu lệnh truy vấn

Tối ưu thời gian truy vấn

Đánh giá hiệu năng

Cập nhật dữ liệu



7. Tìm tổng số sinh viên (riêng biệt) đã tham gia các phần khóa học do giảng viên có ID 100001 giảng dạy.

Thuy\_4 Thuy\_7

```
1 SELECT
2   COUNT(DISTINCT ID)
3 FROM
4   takes
5 WHERE
6   EXISTS( SELECT
7     course_id, sec_id, semester, year
8   FROM
9     teaches
10  WHERE
11    teaches.ID = "100001"
12    AND takes.course_id = teaches.course_id
13    AND takes.sec_id = teaches.sec_id
14    AND takes.semester = teaches.semester
15    AND takes.year = teaches.year);
```

Result Grid

|    | COUNT(DISTINCT ID) |
|----|--------------------|
| 17 |                    |

Result 4

Output

| # | Time     | Action   | Message           |
|---|----------|--|-------------------|
| 4 | 17:42:33 | SELECT COUNT(DISTINCT ID) FROM takes WHERE EXISTS( SELECT course_id, sec_id, semester, year FROM teaches WHERE teaches.ID = "100001" AND takes.course_id = teaches.course_id AND takes.sec_id = teaches.sec_id AND takes.semester = teaches.semester AND takes.year = teaches.year); | 1 row(s) returned |
| 1 | 17:42:58 | SELECT COUNT(DISTINCT ID) FROM takes WHERE EXISTS( SELECT course_id, sec_id, semester, year FROM teaches WHERE teaches.ID = "100001" AND takes.course_id = teaches.course_id AND takes.sec_id = teaches.sec_id AND takes.semester = teaches.semester AND takes.year = teaches.year); | 1 row(s) returned |



# Truy vấn dữ liệu

## Các câu lệnh truy vấn

8. Sắp xếp số lượng giảng viên trong mỗi bộ môn giảng dạy một khóa học vào học kỳ mùa xuân năm 2008 theo thứ tự giảm dần.

Thuy\_4 Thuy\_7 Thuy\_8

```

1 • SELECT
2     instructor.dept_name, COUNT(instructor.ID) AS instr_count
3 FROM
4     instructor,
5     teaches
6 WHERE
7     instructor.ID = teaches.ID
8     AND year = 2008
9     AND semester = 'Spring'
10 GROUP BY instructor.dept_name

```

Result Grid

| dept_name   | instr_count |
|-------------|-------------|
| Marketing   | 22          |
| Astronomy   | 21          |
| Math        | 21          |
| Mech. Eng.  | 19          |
| Cybernetics | 19          |
| Civil Eng.  | 18          |
| Pol. Sci.   | 18          |
| Accounting  | 17          |

Result 1 x

Output

Action Output

| # | Time     | Action  | Message                    |
|---|----------|---|----------------------------|
| 4 | 17:42:33 | SELECT COUNT(DISTINCT ID) FROM takes WHERE EXISTS( SELECT                               | cours... 1 row(s) returned |
| 1 | 17:42:58 | SELECT COUNT(DISTINCT ID) FROM takes WHERE EXISTS( SELECT                               | cours... 1 row(s) returned |
| 2 | 17:44:42 | SELECT instructor.dept_name, COUNT(instructor.ID) AS instr_count FROM instructor, te... | 20 row(s) returned         |

Thông tin về cơ sở dữ liệu

Truy vấn dữ liệu  
Các câu lệnh truy vấn

Tối ưu thời gian truy vấn

Đánh giá hiệu năng

Cập nhật dữ liệu



# Truy vấn dữ liệu

## Các câu lệnh truy vấn

9. Đối với mỗi phần khóa học được cung cấp trong năm 2001, tìm tổng số tín chỉ trung bình (tính theo tín chỉ) của tất cả sinh viên đăng ký trong phần đó, nếu phần đó có ít nhất 2 sinh viên..

```

1 SELECT
2   takes.course_id, takes.semester, year, sec_id, AVG(tot_cred)
3 FROM
4   student,
5   takes
6 WHERE
7   student.ID = takes.ID AND year = 2001
8 GROUP BY course_id, semester, year, sec_id
9 HAVING COUNT(student.ID) >= 2;

```

| course_id | semester | year | sec_id | AVG(tot_cred) |
|-----------|----------|------|--------|---------------|
| 857       | Fall     | 2001 | 1197   | 641.1250      |
| 659       | Fall     | 2001 | 1197   | 531.3333      |
| 960       | Fall     | 2001 | 1197   | 317.2857      |
| 991       | Fall     | 2001 | 1197   | 732.3333      |
| 604       | Fall     | 2001 | 1197   | 585.0000      |

| # | Time     | Action  | Message                    |
|---|----------|---|----------------------------|
| 4 | 17:42:33 | SELECT COUNT(DISTINCT ID) FROM takes WHERE EXISTS( SELECT                               | cours... 1 row(s) returned |
| 1 | 17:42:58 | SELECT COUNT(DISTINCT ID) FROM takes WHERE EXISTS( SELECT                               | cours... 1 row(s) returned |
| 2 | 17:44:42 | SELECT instructor.dept_name, COUNT(instructor.ID) AS instr_count FROM instructor, te... | 20 row(s) returned         |
| 3 | 17:48:04 | SELECT takes.course_id, takes.semester, year, sec_id, AVG(tot_cred) FROM student, t...  | 800 row(s) returned        |

# Truy vấn dữ liệu

## Các câu lệnh truy vấn

Thông tin về cơ sở dữ liệu

Truy vấn dữ liệu  
Các câu lệnh truy vấn

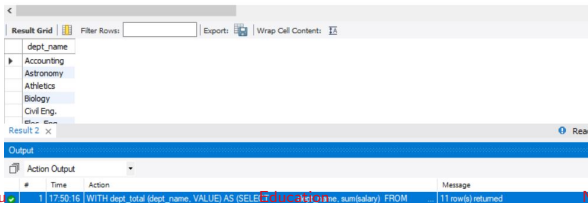
Tối ưu thời gian truy vấn

Đánh giá hiệu năng

Cập nhật dữ liệu

10. Tìm tất cả các phòng ban có tổng mức lương là lớn hơn mức trung bình của tổng tiền lương ở tất cả các bộ phận.

```
Thuy_4 Thuy_9 Thuy_10
WITH dept_total (dept_name, VALUE) AS
  (SELECT
    dept_name, sum(salary)
  FROM
    instructor
  GROUP BY dept_name),
dept_total_avg(VALUE) AS
  (SELECT
    AVG(VALUE)
  FROM dept_total)
SELECT
  dept_name
FROM
  dept_total, dept_total_avg
WHERE dept_total.VALUE > dept_total_avg.VALUE;
```



| dept_name  |
|------------|
| Accounting |
| Astronomy  |
| Athletics  |
| Biology    |
| Civil Eng. |

# Tối ưu thời gian truy vấn

Thông tin về cơ  
sở dữ liệu

Truy vấn dữ liệu

*Các câu lệnh truy vấn*

*Tối ưu thời gian truy  
vấn*

*Đánh giá hiệu năng*

Cập nhật dữ liệu

1. Index
2. Partition



Thông tin về cơ sở dữ liệu

Truy vấn dữ liệu

*Các câu lệnh truy vấn*

*Tối ưu thời gian truy vấn*

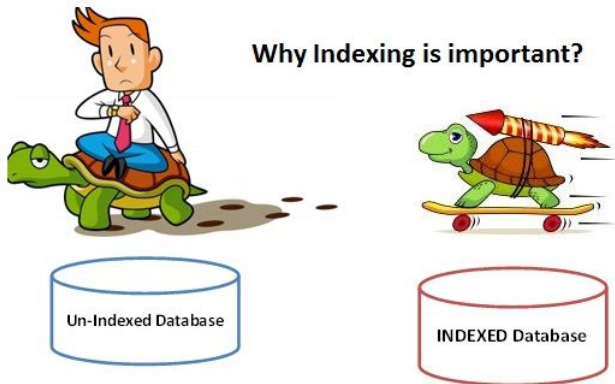
*Đánh giá hiệu năng*

Cập nhật dữ liệu



## Khái niệm

Index là bảng tra cứu đặc biệt có thể sử dụng để tăng nhanh thời gian và hiệu suất thu thập dữ liệu. Index có nhiệm vụ đánh chỉ số cho các bản ghi của bảng.



Thông tin về cơ sở dữ liệu

Truy vấn dữ liệu

*Các câu lệnh truy vấn*

*Tối ưu thời gian truy vấn*

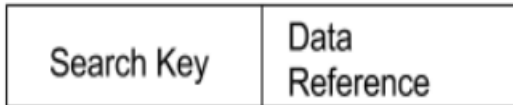
*Đánh giá hiệu năng*

Cập nhật dữ liệu

### Ưu, nhược điểm của Index

| Ưu điểm   | Nhược điểm  |
|---|---|
| <ul style="list-style-type: none"><li>• Đẩy nhanh tốc độ truy vấn xử lý của SELECT, tìm kiếm record theo điều kiện WHERE.</li></ul> | <ul style="list-style-type: none"><li>• Cost của INSERT, UPDATE ,DELETE tăng lên . Hơn nữa, có thể không ảnh hưởng đến tốc độ hoặc tốc độ bị giảm đi bởi index.</li></ul> |

Cấu trúc của index:



Structure of an index

Thông tin về cơ sở dữ liệu

Truy vấn dữ liệu

*Các câu lệnh truy vấn*

*Tối ưu thời gian truy vấn*

*Đánh giá hiệu năng*

Cập nhật dữ liệu

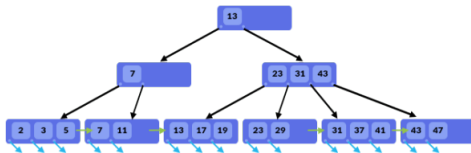


## Cách tạo Index

```
1 CREATE INDEX index_name ON table_name (column_list)
```

## Các kiểu Index

- Kiểu dữ liệu B-Tree

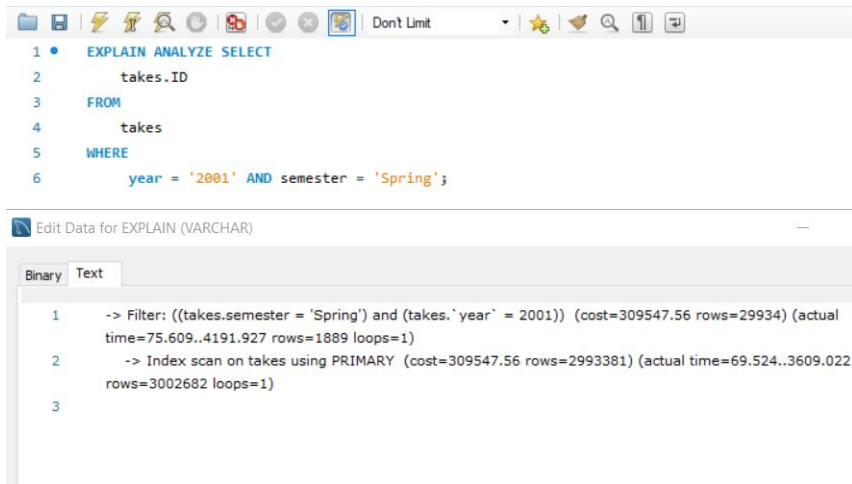


- Các kiểu dữ liệu khác như: Hash, R-Tree, Log - Structure Merge, (full-text, spatial, skiplist)

# Tối ưu thời gian truy vấn

## Index - So sánh thời gian truy vấn

Kết quả trước khi thêm Index



```
1 • EXPLAIN ANALYZE SELECT
2     takes.ID
3 FROM
4     takes
5 WHERE
6     year = '2001' AND semester = 'Spring';
```

Edit Data for EXPLAIN (VARCHAR)

Binary Text

```
1  -> Filter: ((takes.semester = 'Spring') and (takes.`year` = 2001)) (cost=309547.56 rows=29934) (actual
   time=75.609..4191.927 rows=1889 loops=1)
2  -> Index scan on takes using PRIMARY (cost=309547.56 rows=2993381) (actual time=69.524..3609.022
   rows=3002682 loops=1)
3
```

Thông tin về cơ  
sở dữ liệu

Truy vấn dữ liệu

Các câu lệnh truy vấn

Tối ưu thời gian truy  
vấn

Đánh giá hiệu năng

Cập nhật dữ liệu



# Tối ưu thời gian truy vấn

## Index - So sánh thời gian truy vấn

Kết quả sau khi thêm Index

```
1 CREATE INDEX year_semster_index ON takes(year, semester);
2 EXPLAIN ANALYZE SELECT
3     takes.ID
4 FROM
5     takes
6 WHERE
7     year = '2001' AND semester = 'Spring';
8
```

Edit Data for EXPLAIN (VARCHAR)

Binary Text

```
1 -> Index lookup on takes using year_semster_index (year=2001, semester='Spring') (cost=240.93 rows=1889)
   (actual time=0.052..0.664 rows=1889 loops=1)
2
```

Thông tin về cơ  
sở dữ liệu

Truy vấn dữ liệu

*Các câu lệnh truy vấn*

*Tối ưu thời gian truy  
vấn*

*Đánh giá hiệu năng*

Cập nhật dữ liệu



# Truy vấn dữ liệu

## Đánh giá hiệu năng

Thông tin về cơ sở dữ liệu

Truy vấn dữ liệu

*Các câu lệnh truy vấn*

*Tối ưu thời gian truy vấn*

*Đánh giá hiệu năng*

Cập nhật dữ liệu



| Trước khi thêm Index  | Sau khi thêm Index   | Hiệu năng  |
|---|--|--|
| <ul style="list-style-type: none"><li>Thời gian đọc kết quả đầu tiên: 69,524 (ms)</li></ul>     | <ul style="list-style-type: none"><li>Thời gian đọc kết quả đầu tiên: 0,052 (ms)</li></ul>   | <ul style="list-style-type: none"><li>1337 lần</li></ul> |
| <ul style="list-style-type: none"><li>Thời gian đọc tất cả các kết quả: 3690,022 (ms)</li></ul> | <ul style="list-style-type: none"><li>Thời gian đọc tất cả các kết quả: 0,664 (ms)</li></ul> | <ul style="list-style-type: none"><li>5557 lần</li></ul> |
| <ul style="list-style-type: none"><li>Số hàng phải duyệt: 3002682</li></ul>                     | <ul style="list-style-type: none"><li>Số hàng phải duyệt: 1889</li></ul>                     | <ul style="list-style-type: none"><li>1590 lần</li></ul> |

# Tối ưu thời gian truy vấn

## Index - So sánh thời gian truy vấn

Thông tin về cơ sở dữ liệu

Truy vấn dữ liệu

Các câu lệnh truy vấn


Tối ưu thời gian truy vấn

Đánh giá hiệu năng







Cập nhật dữ liệu



Trước khi thêm Index

| Indexes in Table                    |   |       |        |                            | Index Details |         | Drop Index |
|-------------------------------------|---|-------|--------|----------------------------|---------------|---------|------------|
| Visible                             | Key   | Type  | Uni... | Columns                    | Key Name:     |         |            |
| <input checked="" type="checkbox"/> |  PRIMARY | BTREE | YES    | ID, course_id, sec_id, sem | Index Type:   | Packed: |            |
|                                     |   |       |        |                            | Allows NULL:  | Unique: |            |
|                                     |   |       |        |                            | Cardinality:  |         |            |
|                                     |   |       |        |                            | Comment:      |         |            |
|                                     |   |       |        |                            | User Comment: |         |            |

| Columns in table  |              |          |         |
|---|--------------|----------|---------|
| Column  | Type         | Nullable | Indexes |
|  ID        | varchar(25)  | NO       | PRIMARY |
|  course_id | varchar(8)   | NO       | PRIMARY |
|  sec_id    | varchar(8)   | NO       | PRIMARY |
|  semester  | varchar(6)   | NO       | PRIMARY |
|  year      | decimal(4,0) | NO       | PRIMARY |
|  grade     | varchar(2)   | YES      |         |

# Tối ưu thời gian truy vấn

## Index - So sánh thời gian truy vấn

Sau khi thêm Index

Thông tin về cơ sở dữ liệu

Truy vấn dữ liệu

Các câu lệnh truy vấn

Tối ưu thời gian truy vấn

Đánh giá hiệu năng

Cập nhật dữ liệu



### Indexes in Table

| Visible                             | Key                | Type  | Uni... | Columns                         |
|-------------------------------------|--------------------|-------|--------|---------------------------------|
| <input checked="" type="checkbox"/> | PRIMARY            | BTREE | YES    | ID, course_id, sec_id, semester |
| <input checked="" type="checkbox"/> | year_semster_index | BTREE | NO     | year, semester                  |

### Index Details

Drop Index

Key Name:

Index Type:

Allows NULL:

Cardinality:

Comment:

User Comment:

Packed:

Unique:

### Columns in table

| Column    | Type         | Nullable | Indexes                     |
|-----------|--------------|----------|-----------------------------|
| ID        | varchar(25)  | NO       | PRIMARY                     |
| course_id | varchar(8)   | NO       | PRIMARY                     |
| sec_id    | varchar(8)   | NO       | PRIMARY                     |
| semester  | varchar(6)   | NO       | PRIMARY, year_semster_index |
| year      | decimal(4,0) | NO       | PRIMARY, year_semster_index |
| grade     | varchar(2)   | YES      |                             |

# 5 thao tác dữ liệu từ nghiệp vụ thực tế

Thông tin về cơ  
sở dữ liệu

Truy vấn dữ liệu

Cập nhật dữ liệu

*Transaction*

*Tạo một CSDL y hệt  
cấu trúc*

- Thêm một sinh viên mới.
- Thêm một học phần vào chương trình đào tạo.
- Xóa một sinh viên ra khỏi CSDL.
- Cập nhật điểm học phần chi tiết cho một sinh viên.
- Đăng ký học phần chi tiết cho một sinh viên.



Thông tin về cơ sở dữ liệu

Truy vấn dữ liệu

Cập nhật dữ liệu

Transaction

Tạo một CSDL y hệt cấu trúc

- Thêm sinh viên  $A$  thuộc viện  $B$ .

Mã sinh viên  $ID\_new\_student = MAX(ID)$  trong bảng **student**.

- Cài đặt:

```
1 START TRANSACTION;  
2  
3 SELECT ID_new_student = MAX(ID) + 1 FROM student;  
4  
5 INSERT INTO student(ID, name, dept_name, tot_cred)  
6 VALUES(ID_new_student, "A", "B", 0);  
7  
8 COMMIT;
```



# Thêm một học phần vào chương trình đào tạo

Thông tin về cơ  
sở dữ liệu

Truy vấn dữ liệu

Cập nhật dữ liệu

*Transaction*

*Tạo một CSDL y hệt  
cấu trúc*

## Các bước thực hiện

- Bắt đầu transaction.
- Lấy mã học phần  $@course\_id = MAX(course\_id) + 1$  từ bảng **course**.
- Thêm học phần với mã học phần  $@course\_id$  vào bảng **course**.
- Cập nhật bảng **prereq** với  $@course\_d$ .
- Thực hiện Commit.



# Thêm một học phần vào chương trình đào tạo

Thông tin về cơ  
sở dữ liệu

Truy vấn dữ liệu

Cập nhật dữ liệu

*Transaction*

*Tạo một CSDL y hệt  
cấu trúc*

- Thêm học phần ( $p\_title, p\_dept\_name, p\_credits$ ).
- **Cài đặt:**

```
1 START TRANSACTION;  
2  
3 SELECT @course_id:=MAX(course_id) + 1 FROM course;  
4  
5 INSERT INTO course(course_id, title, dept_name, credits)  
6 VALUES(@course_id, p_title, p_dept_name, p_credits);  
7  
8 INSERT INTO prereq(course_id, prereq_id)  
9 VALUES (@course_id, p_course_id1);  
10  
11 INSERT INTO prereq(course_id, prereq_id)  
12 VALUES (p_course_id2, @course_id);  
13  
14 COMMIT;
```





# Xóa một sinh viên ra khỏi CSDL

Thông tin về cơ  
sở dữ liệu

Truy vấn dữ liệu

Cập nhật dữ liệu

*Transaction*

*Tạo một CSDL y hệt  
cấu trúc*

## Các bước thực hiện

- Bắt đầu transaction.
- Kiểm tra có tồn tại sinh viên này không. Nếu không tồn tại. → Rollback.
- Xóa dữ liệu với *ID* sinh viên cần xóa trong các bảng **student**, **takes** và **advisor**.
- Thực hiện Commit.

# Xóa một sinh viên ra khỏi CSDL

Thông tin về cơ sở dữ liệu

Truy vấn dữ liệu

Cập nhật dữ liệu

*Transaction*

*Tạo một CSDL y hệt cấu trúc*

- Xóa sinh viên có mã số  $p\_ID$ .
- **Cài đặt:**

```
1 START TRANSACTION;  
2  
3 SELECT @flag:=COUNT(ID) FROM student WHERE ID = p_ID;  
4  
5 IF (@flag = 0)  
6 THEN ROLLBACK;  
7 END IF;  
8  
9 DELETE FROM takes WHERE ID = p_ID;  
10 DELETE FROM advisor WHERE s_id = p_ID;  
11 DELETE FROM student WHERE ID = p_ID;  
12  
13 COMMIT;
```



# Cập nhật điểm học phần chi tiết cho một sinh viên

Thông tin về cơ  
sở dữ liệu

Truy vấn dữ liệu

Cập nhật dữ liệu

*Transaction*

*Tạo một CSDL y hệt  
cấu trúc*

## Các bước thực hiện

- Bắt đầu transaction.
- Kiểm tra có tồn tại học phần cụ thể như vậy không. Nếu không tồn tại. → Rollback.
- Cập nhật điểm vào bảng **takes**.
- Điểm học phần lớn hơn  $F$ . → Cập nhật **tot\_cred** trong bảng **student**. → Commit.
- Điểm học phần bằng  $F$  → Commit.



# Cập nhật điểm học phần chi tiết cho một sinh viên

Thông tin về cơ sở dữ liệu

Truy vấn dữ liệu

Cập nhật dữ liệu

*Transaction*

*Tạo một CSDL y hệt cấu trúc*



- Cập nhật điểm học phần chi tiết:  
(*p\_ID*, *p\_course\_id*, *p\_sec\_id*, *p\_semester*, *p\_year*, *p\_grade*).

- **Cài đặt:**

```
1 START TRANSACTION;
2
3 SELECT @flag_fi := COUNT(course_id) FROM takes
4 WHERE ID = p_ID AND course_id = p_course_id
5 AND sec_id = p_sec_id AND semester = p_semester AND year = p_year;
6
7 INSERT INTO takes(ID, course_id, sec_id, semester, year, grade)
8 VALUES (p_ID, p_course_id, p_sec_id, p_semester, p_year, grade);
9
10 SELECT @flag_se := grade FROM takes
11 WHERE ID = p_ID AND course_id = p_course_id
12 AND sec_id = p_sec_id AND semester = p_semester AND year = p_year;
13
14 SELECT @cred := credits FROM course WHERE course_id = p_course_id;
15
16 IF (@flag_fi = 0) THEN
17 ROLLBACK;
18 ELSEIF (@flag_se != 'F') THEN
19 UPDATE student SET tot_cred = tot_cred + @cred WHERE ID = p_ID;
20 COMMIT;
21 ELSE COMMIT;
22 END IF;
```

# Đăng ký học phần chi tiết cho một sinh viên

Thông tin về cơ  
sở dữ liệu

Truy vấn dữ liệu

Cập nhật dữ liệu

*Transaction*

*Tạo một CSDL y hệt  
cấu trúc*



## Các bước thực hiện

- Input danh sách các học phần sinh viên đăng ký.
- Mỗi học phần:
- Bắt đầu transaction.
- Kiểm tra:
  1. Học phần này có tồn tại hay không?
  2. Sinh viên đã đăng ký học phần này chưa?
  3. Sinh viên có vượt quá số tín chỉ cho phép trong một kỳ.
- Nếu:
  - ▶ Đủ cả 3 điều kiện → Thực hiện yêu cầu → Commit.
  - ▶ Vi phạm 1 trong 3 điều kiện → Rollback.

# Đăng ký học phần chi tiết một sinh viên

Thông tin về cơ sở dữ liệu

Truy vấn dữ liệu

Cập nhật dữ liệu

*Transaction*

*Tạo một CSDL y hệt cấu trúc*



- Đăng ký học phần chi tiết:  
(*p\_ID*, *p\_course\_id*, *p\_sec\_id*, *p\_semester*, *p\_year*, *p\_grade*).

- **Cài đặt:**

```
1 START TRANSACTION;
2
3 SELECT @flag-fi := COUNT(course_id) FROM course WHERE course_id = p_course_id;
4
5 SELECT @flag-se := COUNT(course_id) FROM takes
6 WHERE course_id = p_course_id AND ID = p_ID AND semester = p_semester AND year = p_year;
7
8 INSERT INTO takes(ID, course_id, sec_id, semester, year, grade)
9 VALUES (p_ID, p_course_id, p_sec_id, p_semester, p_year, 0);
10
11 SELECT @flag-th := SUM(credits) FROM takes AS a LEFT JOIN course AS b
12 ON a.course_id = b.course_id WHERE ID = p_ID AND semester = p_semester AND year = p_year;
13
14 IF (@flag-se != 0 OR @flag-fi = 0 OR @flag-th > 24) THEN
15 ROLLBACK;
16 ELSE COMMIT;
17 END IF;
```

# Tạo một CSDL y hệt cấu trúc

Thông tin về cơ  
sở dữ liệu

Truy vấn dữ liệu

Cập nhật dữ liệu

*Transaction*

*Tạo một CSDL y hệt  
cấu trúc*

## Ý tưởng

- Tạo một stored procedure để copy từ data **source** sang **target** với cấu trúc:
  - ▶ Tạo các bảng có cấu trúc như data **source**.
  - ▶ Insert dữ liệu từ các bảng tương ứng trong data **source**.

# Các bước thực hiện

Thông tin về cơ sở dữ liệu

Truy vấn dữ liệu

Cập nhật dữ liệu

Transaction

Tạo một CSDL y hết cấu trúc



- Tạo stored procedure **"copyDB"** với hai tham số đầu vào là **"DBsource"** và **"DBtarget"**.
- Tạo bảng **"Table\_list"** với các cột (**id**, **tbl**) chứa các bảng của **"DBsource"**.
- Tạo bảng **"SQLscript"** với các cột (**id**, **SQLtext**) chứa các câu lệnh cần thực hiện.
- Duyệt bảng **"Table\_list"** và thực hiện:

```
1 INSERT INTO SQLscript (SQLtext)
2 "CREATE TABLE DBtarget.tbl LIKE DBsource.tbl";
```

- Duyệt bảng **"Table\_list"** và thực hiện:

```
1 INSERT INTO SQLscript (SQLtext)
2 "INSERT DBtarget.tbl SELECT * FROM DBsource.tbl"
```

- Thực hiện các câu lệnh trong bảng **"SQLscript"**.



# Đánh giá

Thông tin về cơ  
sở dữ liệu

Truy vấn dữ liệu

Cập nhật dữ liệu

*Transaction*

*Tạo một CSDL y hệt  
cấu trúc*

- Ưu điểm:
  - ▶ Tính tái sử dụng cao.
  - ▶ Dễ thực hiện.
- Nhược điểm:
  - ▶ Thời gian thực hiện procedure chậm đối với database kích cỡ lớn.
  - ▶ Khó kiểm soát dữ liệu copy.
- Cách cải tiến:
  - ▶ Thêm cột **Index** được sắp xếp là khóa chính để tăng tốc độ **Insert**.
  - ▶ Cài đặt thêm **transaction** để kiểm soát dữ liệu.



Thông tin về cơ  
sở dữ liệu

Truy vấn dữ liệu

Cập nhật dữ liệu

*Transaction*

*Tạo một CSDL y hệt  
cấu trúc*

**Sau khi hoàn thành bài tập lớn chúng em đã có thể làm được:**

- Biết cách xây dựng một cơ sở dữ liệu
- Thực hiện truy vấn dữ liệu và tối ưu các truy vấn sử dụng index và partition
- Tạo các transaction để thực hiện các nghiệp vụ thực tế
- Sao chép một Database này sang một Database khác

**Một số hạn chế**

- Chưa tối ưu đc thời gian chạy với CSDL lớn
- Chưa nêu đủ partition



Thông tin về cơ sở dữ liệu

Truy vấn dữ liệu

Cập nhật dữ liệu

*Transaction*

*Tạo một CSDL y hệt cấu trúc*



## Tổng kết

|                                  | Thuần  | Thủy  | Yến  | Sang  | Đức   |
|----------------------------------|--|---|--|---|---|
| Phần 1 - Thông tin cơ sở dữ liệu | <ul style="list-style-type: none"> <li>Thiết kế Diagram</li> <li>Thuyết trình</li> </ul> |   |  |   |   |
| Phần 2 - Truy vấn                |  | <ul style="list-style-type: none"> <li>10 Queries</li> <li>Index</li> <li>Thuyết Trình</li> </ul> | <ul style="list-style-type: none"> <li>Partition</li> </ul>  |   |   |
| Phần 3 - Cập nhật dữ liệu        |  |   | <ul style="list-style-type: none"> <li>30 procedure</li> <li>Sinh tự động procedure</li> <li>Thuyết trình</li> </ul> | <ul style="list-style-type: none"> <li>5 Transactions</li> <li>Copy database bằng procedure.</li> <li>Thuyết trình</li> </ul> |   |
| Slide                            |  |   | <ul style="list-style-type: none"> <li>Làm slide</li> </ul>  |   | <ul style="list-style-type: none"> <li>Làm slide</li> </ul> |

Xin cảm ƠN



Nhóm 4