

NGÔN NGỮ SQL

- Tổng quan SQL (Structured query language)
 - Truy vấn cơ bản
 - Truy vấn lồng
 - Truy vấn sử dụng hàm kết gộp
 - Truy vấn gom nhóm
 - Truy vấn con ở mệnh đề from
 - Truy vấn kết ở mệnh đề from
-

TỔNG QUAN VỀ SQL

- Ngôn ngữ cấp cao thân thiện hướng khai báo
 - Được phát triển bởi IBM
 - Được ANSI công nhận và phát triển thành chuẩn
 - SQL-86
 - SQL-92
 - SQL-99
 -
-

TRUY VẤN CƠ BẢN

- SQL và ĐSQH

$\pi_L(\sigma_C(r)) \longleftrightarrow \begin{array}{l} \text{SELECT } L \\ \text{FROM } r \\ \text{WHERE } C \end{array}$

TRUY VẤN CƠ BẢN

- Ví dụ

MANV	HONV	TENLO	TENNV	NGSINH	DCHI	PHAI	LUONG	MA_NQL	PHG
33344555	Nguyen	Thanh	Tung	12/08/55	Lam Dong	Nam	40000	88866555	5
98798798	Nguyen	Manh	Hung	09/15/62	Ba Ria VT	Nam	38000	33344555	5

SELECT *

FROM NHANVIEN

WHERE MA_DV=5

$\sigma_{MA_DV=5}(NHANVIEN)$

TRUY VẤN CƠ BẢN

- Ví dụ

```
SELECT MA_NV, HONV, TENLOT, TENNV  
FROM NHANVIEN  
WHERE MA_DV=5 AND PHAI='Nam'
```

$\pi_{MA_NV, HONV, TENLOT, TENNV}(\sigma_{MA_DV=5 \wedge PHAI='Nam'}(NHANVIEN))$

MA_NV	HONV	TENLOT	TENNV
333445555	Nguyen	Thanh	Tung
987987987	Nguyen	Manh	Hung

TRUY VẤN CƠ BẢN

- Đặt tên cột cho kết quả truy xuất

```
SELECT MA_NV, HONV AS HO, TENLOT AS 'TEN LOT', TENNV AS TEN  
FROM NHANVIEN  
WHERE MA_DV=5 AND PHAI='Nam'
```

$$\rho_{MA_NV, HO, TEN\ LOT, TEN}(\pi_{MA_NV, HONV, TENLOT, TENNV}(\sigma_{MA_DV=5 \wedge PHAI='Nam'}(NHANVIEN)))$$

MA_NV	HO	TEN LOT	TEN
333445555	Nguyen	Thanh	Tung
987987987	Nguyen	Manh	Hung

TRUY VẤN CƠ BẢN

- Ví dụ

```
SELECT MA_NV, HONV + ' ' + TENLOT + ' ' + TENNV AS 'HO TEN'  
FROM NHANVIEN  
WHERE MA_DV=5 AND PHAI='Nam'
```

MA_NV	HO TEN
333445555	Nguyen Thanh Tung
987987987	Nguyen Manh Hung

$$\rho_{MA_NV, HO\ TEN}(\pi_{MA_NV, HONV+TENLOT+TENNV}$$
$$(\sigma_{MA_DV=5 \wedge PHAI='Nam'}(NHANVIEN)))$$

TRUY VẤN CƠ BẢN

- Ví dụ

```
SELECT MA_NV, LUONG*1.1 AS 'LUONG10%'  
FROM NHANVIEN  
WHERE MA_DV=5 AND PHAI='Nam'
```

MA_NV	LUONG10%
333445555	33000
987987987	27500

$$\rho_{MA_NV, LUONG10\%}(\pi_{MA_NV, LUONG*1.1}(\sigma_{MA_DV=5 \wedge PHAI='Nam'}(NHANVIEN)))$$

TRUY VẤN CƠ BẢN

- Ví dụ
 - Loại bỏ các hàng giống nhau

SELECT LUONG

FROM NHANVIEN

WHERE MA_DV=5 AND PHAI='Nam'

LUONG
30000
25000
25000
38000

SELECT DISTINCT LUONG

FROM NHANVIEN

WHERE MA_DV=5 AND PHAI='Nam'

LUONG
30000
25000
38000

TRUY VẤN CƠ BẢN

- Ví dụ

SELECT MA_NV, TENNV

FROM NHANVIEN NV, DONVI DV

WHERE TENDV='Nghien cứu' AND NV.MA_DV=DV.MA_DV

Biểu thức logic

↓
TRUE

↓
TRUE

TRUY VẤN CƠ BẢN

- Ví dụ
 - Mức độ ưu tiên trong biểu thức logic

```
SELECT MA_NV, TENNV  
FROM NHANVIEN NV, DONVI DV  
WHERE (DV.TENDV='Nghien cuu' OR DV.TENDV='Quan ly')  
      AND NV.MA_DV=DV.MA_DV
```

TRUY VẤN CƠ BẢN

- Ví dụ
 - Toán tử BETWEEN

```
SELECT MA_NV, TENNV  
FROM NHANVIEN  
WHERE LUONG>20000 AND LUONG<30000
```

```
SELECT MA_NV, TENNV  
FROM NHANVIEN  
WHERE LUONG BETWEEN 20000 AND 30000
```

TRUY VẤN CƠ BẢN

- Ví dụ
 - Toán tử NOT BETWEEN

```
SELECT MA_NV, TENNV  
FROM NHANVIEN  
WHERE LUONG NOT BETWEEN 20000 AND 30000
```

TRUY VẤN CƠ BẢN

- Ví dụ
 - Ngày giờ trong biểu thức điều kiện

```
SELECT MA_NV, TENNV
```

```
FROM NHANVIEN
```

```
WHERE NGSINH BETWEEN '1955-12-08' AND '1966-07-19'
```

TRUY VẤN CƠ BẢN

- Giá trị NULL
 - Sử dụng trong trường hợp
 - ✓ Không biết
 - ✓ Không thể áp dụng
 - ✓ Không tồn tại
 - Những biểu thức tính toán liên quan đến giá trị NULL sẽ cho ra **kết quả là NULL**
 - Những biểu thức so sánh liên quan đến giá trị NULL sẽ cho ra kết quả là **UNKNOWN**
-

NGÔN NGỮ TRUY VẤN DỮ LIỆU

- Ví dụ
 - NULL

```
SELECT MA_NV, TENNV  
FROM NHANVIEN  
WHERE MA_LD IS NULL
```

```
SELECT MA_NV, TENNV  
FROM NHANVIEN  
WHERE MA_LD IS NOT NULL
```

TRUY VẤN CƠ BẢN

- Mệnh đề ORDER BY
 - Dùng để hiển thị kết quả truy vấn theo một thứ tự nào đó
 - Cú pháp

SELECT <danh sách các cột>

FROM <danh sách các bảng>

WHERE <điều kiện>

ORDER BY <danh sách các cột>

✓ **ASC**: tăng (mặc định)

✓ **DESC**: giảm

TRUY VẤN CƠ BẢN

- Ví dụ
 - Sắp xếp nhân viên trong đơn vị số 4 theo thứ tự lương tăng dần

```
SELECT MA_NV,HO_NV,TEN_NV,LUONG  
FROM NHANVIEN  
WHERE MA_DV=4  
ORDER BY LUONG ASC
```

TRUY VẤN CƠ BẢN

- Phép toán tập hợp trong SQL
 - SQL có cài đặt các phép toán
 - ✓ Hội (UNION)
 - ✓ Giao (INTERSECT)
 - ✓ Trừ (EXCEPT)
 - Kết quả trả về là tập hợp
 - ✓ (Tự động) Loại bỏ các bộ trùng nhau
 - ✓ Để giữ lại các bộ trùng nhau, dùng kèm từ khóa ALL
 - ❖ UNION ALL
 - ❖ INTERSECT ALL
 - ❖ EXCEPT ALL
-

TRUY VẤN CƠ BẢN

- Phép toán tập hợp trong SQL
 - UNION
 - ✓ Cú pháp

SELECT <ds cột> FROM <ds bảng> WHERE <điều kiện>

UNION [ALL]

SELECT <ds cột> FROM <ds bảng> WHERE <điều kiện>

TRUY VẤN CƠ BẢN

- Phép toán tập hợp trong SQL

- UNION

- ✓ Cho biết các mã dự án có

- ❖ Nhân viên có họ là Nguyen tham gia, hoặc

- ❖ Trưởng phòng quản lý dự án đó họ là Nguyen

```
SELECT MA_DA FROM NHANVIEN NV, THAMGIA_DA TG
WHERE NV.MA_NV=TG.MA_NV AND NV.HO_NV='Nguyen'
UNION
```

```
SELECT MA_DA FROM NHANVIEN NV, DONVI DV, DUAN DA
WHERE NV.MA_NV=DV.MA_LD AND DV.MA_DV=DA.MA_DV
AND NV.HO_NV='Nguyen'
```

TRUY VẤN CƠ BẢN

- Phép toán tập hợp trong SQL
 - INTERSECT
 - ✓ Cú pháp

SELECT <ds cột> FROM <ds bảng> WHERE <điều kiện>

INTERSECT [ALL]

SELECT <ds cột> FROM <ds bảng> WHERE <điều kiện>

TRUY VẤN CƠ BẢN

- Phép toán tập hợp trong SQL
 - EXCEPT
 - ✓ Cú pháp

SELECT <ds cột> FROM <ds bảng> WHERE <điều kiện>

EXCEPT [ALL]

SELECT <ds cột> FROM <ds bảng> WHERE <điều kiện>

TRUY VẤN LỒNG

- Truy vấn lồng (Embedded Query)
 - ✓ Các câu lệnh SELECT có thể lồng nhau ở nhiều mức
 - ✓ Câu truy vấn con trả về **một tập hợp các giá trị**
 - ✓ Các câu truy vấn con trong cùng một mệnh đề WHERE được kết hợp bằng **phép nối logic**
 - ✓ Mệnh đề WHERE của câu truy vấn cha
 - ❖ **<biểu thức> <so sánh tập hợp> <truy vấn con>**
 - ❖ So sánh tập hợp thường đi kèm với các toán tử
 - IN, NOT IN
 - ALL, ANY, SOME
 - ❖ Kiểm tra sự tồn tại
 - EXISTS, NOT EXISTS

TRUY VẤN LỒNG

- Ví dụ
 - Tìm những nhân viên không có thân nhân nào

```
SELECT *  
FROM NHANVIEN  
WHERE MA_NV NOT IN (  
    SELECT MA_NV  
    FROM THANNHAN)
```

```
SELECT *  
FROM NHANVIEN  
WHERE MA_NV <> ALL (  
    SELECT MA_NV  
    FROM THANNHAN)
```

TRUY VẤN LÒNG

- Ví dụ
 - Tìm những nhân viên có lương nhiều hơn ít nhất một nhân viên trong đơn vị mã số 4

```
SELECT *  
FROM NHANVIEN  
WHERE LUONG > ANY (  
    SELECT LUONG  
    FROM NHANVIEN  
    WHERE MA_DV=4 )
```

```
SELECT NV1.*  
FROM NHANVIEN NV1,  
      NHANVIEN NV2  
WHERE NV1.LUONG > NV2.LUONG  
      AND NV2.PHG=4
```

TRUY VẤN LỒNG

- Ví dụ
 - Tìm nhân viên có lương cao hơn lương của tất cả nhân viên đơn vị mã số 4

```
SELECT *  
FROM NHANVIEN  
WHERE LUONG > ALL (  
    SELECT LUONG  
    FROM NHANVIEN  
    WHERE MA_DV=4 )
```

TRUY VẤN LÒNG

- Ví dụ
 - Tìm những trưởng phòng có ít nhất một thân nhân

```
SELECT *  
FROM NHANVIEN  
WHERE MA_NV IN (SELECT MA_NV FROM THANNHAN)  
AND MA_NV IN (SELECT MA_LD FROM DONVI)
```

TRUY VẤN LÒNG

- Ví dụ
 - Tìm các nhân viên trong phòng Nghiên cứu

```
SELECT MA_NV, TEN_NV
```

```
FROM NHANVIEN NV, DONVI DV
```

```
WHERE DV.TEN_DV='Nghien cuu' AND NV.MA_DV=DV.MA_DV
```

```
SELECT MA_NV, TEN_NV
```

```
FROM NHANVIEN NV
```

```
WHERE EXISTS (
```

```
    SELECT *
```

```
    FROM DONVI DV
```

```
    WHERE DV.TEN_DV='Nghien cuu' AND DV.MA_DV=NV.MA_DV )
```

TRUY VẤN LỒNG

- Ví dụ
 - Tìm nhân viên có người thân trùng tên và cùng giới tính

```
SELECT *  
FROM NHANVIEN NV  
WHERE EXISTS (  
    SELECT *  
    FROM THANNHAN TN  
    WHERE NV.MA_NV=TN.MA_NV  
    AND NV.TEN_NV=TN.TEN_TN  
    AND NV.PHAI=TN.PHAI )
```

TRUY VẤN LỒNG

- Ví dụ
 - Tìm nhân viên không có thân nhân nào

```
SELECT *  
FROM NHANVIEN NV  
WHERE NOT EXISTS (  
    SELECT *  
    FROM THANNHAN TN  
    WHERE NV.MA_NV=TN.MA_NV)
```

TRUY VẤN LỒNG

- Ví dụ
 - Tìm nhân viên có lương cao hơn ít nhất một nhân viên thuộc đơn vị mã số 4

```
SELECT *  
FROM NHANVIEN NV1  
WHERE EXISTS (  
    SELECT *  
    FROM NHANVIEN NV2  
    WHERE NV2.MA_DV=4  
    AND NV1.LUONG>NV2.LUONG)
```

TRUY VẤN LỒNG

- Ví dụ
 - Tìm trưởng phòng có ít nhất một thân nhân

```
SELECT * FROM NHANVIEN NV
WHERE EXISTS (
    SELECT *
    FROM THANNHAN TN
    WHERE NV.MA_NV=TN.MA_NV)
AND EXISTS (
    SELECT *
    FROM DONVI DV
    WHERE NV.MANV=DV.MA_LD )
```

TRUY VẤN LỒNG

- Phép toán tập hợp trong SQL
 - IN
 - ✓ <tên cột> **IN** <câu truy vấn con>
 - ✓ Thuộc tính ở mệnh đề SELECT của truy vấn con **phải cùng kiểu dữ liệu** với thuộc tính ở mệnh đề WHERE của truy vấn cha
 - EXISTS
 - ✓ Không cần **thuộc tính, hằng số hay biểu thức** nào khác đứng trước
 - ✓ Không cần liệt kê tên thuộc tính ở mệnh đề SELECT của truy vấn con
 - ✓ Những câu truy vấn có = **ANY hay IN** đều có thể chuyển sang câu truy vấn có **EXISTS**
-

TRUY VẤN VỚI CÁC HÀM KẾT GỘP

- Hàm kết gộp
 - Các hàm kết gộp được đặt trong **mệnh đề SELECT**
 - COUNT
 - ✓ COUNT(*): Đếm số hàng
 - ✓ COUNT(tên_thuộc_tính): Đếm số giá trị **khác NULL**
 - ✓ COUNT(DISTINCT tên_thuộc_tính): Đếm số giá trị **khác nhau và khác NULL**
 - MIN
 - MAX
 - SUM
 - AVERAGE
-

TRUY VẤN VỚI CÁC HÀM KẾT GỘP

- Ví dụ
 - Tính tổng lương, lương cao nhất, lương thấp nhất, lương trung bình của các nhân viên

```
SELECT SUM(LUONG), MAX(LUONG), MIN(LUONG), AVG(LUONG)  
FROM NHANVIEN
```

TRUY VẤN VỚI CÁC HÀM KẾT GỘP

- Ví dụ
 - Cho biết số lượng nhân viên của phòng Nghiên cứu

```
SELECT COUNT(*) AS SL_NV  
FROM NHANVIEN NV, DONVI DV  
WHERE NV.MA_DV=DV.MA_DV AND DV.TEN_DV='Nghiên cứu'
```

TRUY VẤN VỚI CÁC HÀM KẾT GỘP

- Ví dụ
 - Cho biết số lượng nhân viên trong từng đơn vị

MANV	HONV	TENLO	TENNV	NGSINH	DCHI	PHAI	LUONG	MA_NQL	PHG
33344555	Nguyen	Thanh	Tung	12/08/1955	638 NVC	Nam	40000	88866555	5
98798798	Nguyen	Manh	Hung	09/15/1962	Ba Ria VT	Nam	38000	33344555	5
45345345	Tran	Thanh	Tam	07/31/1972	543 MTL	Nu	25000	33344555	5
99988777	Bui	Ngoc	Hang	07/19/1968	33 NTH Q1	Nu	38000	98765432	4
98765432	Le	Quynh	Nhu	07/20/195	219 TD Q3	Nu	43000	88866555	4
98798798	Tran	Hong	Quang	04/08/1969	980 LHP	Nam	25000	98765432	4
88866555	Pham	Van	Vinh	11/10/1945	450 TV HN	Nam	55000	NULL	1

TRUY VẤN GOM NHÓM

- Cú pháp

SELECT <danh sách các cột>

FROM <danh sách các bảng>

WHERE <điều kiện>

GROUP BY <danh sách các cột gom nhóm>

- Sau khi gom nhóm, các bộ có cùng giá trị tại các thuộc tính gom nhóm sẽ cùng nhóm

TRUY VẤN GOM NHÓM

- Ví dụ
 - Cho biết số lượng nhân viên trong từng đơn vị

```
SELECT MA_DV, COUNT(*) AS SL_NV  
FROM NHANVIEN  
GROUP BY MA_DV
```

```
SELECT DV.TEN_DV, COUNT(*) AS SL_NV  
FROM NHANVIEN NV, DONVI DV  
WHERE NV.MA_DV=DV.MA_DV  
GROUP BY DV.TEN_DV
```


TRUY VẤN GOM NHÓM

- Ví dụ
 - Với mỗi nhân viên cho biết mã số, tên, số lượng dự án và tổng thời gian mà họ tham gia

```
SELECT MA_NV, COUNT(*) AS SL_DA, SUM(THGIAN) AS TONG_TG
FROM THAMGIA_DA
GROUP BY MA_NV
```

```
SELECT NV.HO_NV, NV.TEN_NV, COUNT(*) AS SL_DA,
SUM(TG.THGIAN) AS TONG_TG
FROM THAMGIA_DA TG, NHANVIEN NV
WHERE NV.MA_NV=TG.MA_NV
GROUP BY NV.MA_NV, NV.HO_NV, NV.TEN_NV
```

TRUY VẤN GOM NHÓM

- Điều kiện trên nhóm

- ✓ Cú pháp

SELECT <danh sách các cột>

FROM <danh sách các bảng>

WHERE <điều kiện>

GROUP BY <danh sách các cột gom nhóm>

HAVING <điều kiện trên nhóm>

TRUY VẤN GOM NHÓM

- Ví dụ
 - Cho biết những nhân viên tham gia hai dự án trở lên

```
SELECT MA_NV  
FROM THAMGIA_DA  
GROUP BY MA_NV  
HAVING COUNT(*) >= 2
```

TRUY VẤN GOM NHÓM

- Mệnh đề GROUP BY
 - ✓ Các thuộc tính trong mệnh đề SELECT (trừ các thuộc tính trong hàm kết hợp) phải xuất hiện trong mệnh đề GROUP BY
 - Mệnh đề HAVING
 - ✓ Sử dụng các hàm kết gộp trong mệnh đề SELECT để kiểm tra một số điều kiện
 - ✓ Chỉ kiểm tra điều kiện trên nhóm, không là điều kiện lọc từng bộ
 - ✓ Sau khi gom nhóm, điều kiện trên nhóm mới thực hiện
-

TRUY VẤN GOM NHÓM

- Thứ tự thực hiện GROUP BY ... HAVING ...
 - ✓ Chọn ra **những hàng thỏa mãn** điều kiện WHERE
 - ✓ Gom nhóm kết quả này tương ứng mệnh đề GROUP BY
 - ✓ Áp dụng **các hàm kết hợp cho mỗi nhóm**
 - ✓ Bỏ qua những nhóm không thỏa mãn điều kiện trong mệnh đề HAVING
 - ✓ Rút trích giá trị theo **các cột và hàm kết hợp trong mệnh đề SELECT**
-

TRUY VẤN GOM NHÓM

- Ví dụ
 - Cho biết những đơn vị có lương trung bình nhân viên trên 2.000.000

```
SELECT MA_DV, AVG(LUONG) AS LUONG_TB  
FROM NHANVIEN GROUP BY MA_DV  
HAVING AVG(LUONG) > 20000
```

```
SELECT DV.TEN_DV, AVG(NV.LUONG) AS LUONG_TB  
FROM NHANVIEN NV, DONVI DV  
WHERE DV.MA_DV=NV.MA_DV  
GROUP BY TEN_DV  
HAVING AVG(NV.LUONG) > 20000
```

TRUY VẤN GOM NHÓM

- Ví dụ
 - Tìm những đơn vị có lương trung bình cao nhất

```
SELECT MA_DV, AVG(LUONG) AS LUONG_TB
FROM NHANVIEN
GROUP BY MA_DV
HAVING MAX(LUONG) > (
        SELECT AVG(LUONG)
        FROM NHANVIEN
        GROUP BY MA_DV)
```

TRUY VẤN GOM NHÓM

- Ví dụ
 - Tìm các nhân viên tham gia tất cả các dự án

```
SELECT NV.MA_NV, NV.TEN_NV
FROM NHANVIEN NV, THAMGIA_DA TG
WHERE NV.MA_NV=TG.MA_NV
GROUP BY NV.MA_NV, NV.TEN_NV
HAVING COUNT(*) = (
    SELECT COUNT(*)
    FROM DU_AN)
```


TRUY VẤN GOM NHÓM

- Ví dụ
 - Tìm các nhân viên tham gia tất cả các dự án

```
SELECT MA_NV, TEN_NV
FROM NHANVIEN
WHERE (SELECT TG.MA_DA
       FROM NHANVIEN NV, THAMGIA_DA TG
       WHERE NV.MA_NV=TG.MA_NV)
=
(SELECT MA_DA
 FROM THAMGIA_DA)
```

TRUY VẤN CON Ở MỆNH ĐỀ FROM

- Kết quả trả về của câu truy vấn con là **một bảng**
 - ✓ Bảng trung gian trong quá trình truy vấn
 - ✓ Không có lưu trữ thật sự
- Cú pháp

SELECT <danh sách các cột>

FROM R1, R2, (<truy vấn con>) **AS** tên_bảng

WHERE <điều kiện>

TRUY VẤN CON Ở MỆNH ĐỀ FROM

- Ví dụ
 - Cho biết những đơn vị có lương trung bình của nhân viên cao hơn 2.000.000

```
SELECT MA_DV, AVG(LUONG) AS LUONG_TB  
FROM NHANVIEN  
GROUP BY MA_DV  
HAVING AVG(LUONG) > 20000
```

TRUY VẤN CON Ở MỆNH ĐỀ FROM

- Ví dụ
 - Cho biết những đơn vị có lương trung bình của nhân viên cao hơn 2.000.000

```
SELECT DV.MA_DV, DV.TEN_DV, AVG(NV.LUONG) AS LUONG_TB  
FROM NHANVIEN NV, DONVI DV  
WHERE NV.MA_DV=DV.MA_DV  
GROUP BY DV.MA_DV, DV.TEN_DV  
HAVING AVG(NV.LUONG) > 20000
```

TRUY VẤN CON Ở MỆNH ĐỀ FROM

- Ví dụ
 - Cho biết những đơn vị có lương trung bình của nhân viên cao hơn 2.000.000

```
SELECT DV.TEN_DV, TEMP.LUONG_TB
FROM DONVI DV, (SELECT NV.MA_DV, AVG(NV.LUONG) AS LUONG_TB
                FROM NHANVIEN NV
                GROUP BY NV.MA_DV
                HAVING AVG(NV.LUONG)> 20000 ) AS TEMP
WHERE DV.MA_DV=TEMP.MA_DV
```

TRUY VẤN KẾT Ở MỆNH ĐỀ FROM

- Cú pháp

SELECT <danh sách các cột>

FROM R1 [**INNER**] **JOIN** R2 **ON** <biểu thức>

WHERE <điều kiện>

TRUY VẤN KẾT Ở MỆNH ĐỀ FROM

- Ví dụ
 - Tìm mã và tên nhân viên làm việc tại phòng Nghiên cứu

```
SELECT NV.MA_NV, NV.TEN_NV  
FROM NHANVIEN NV, DONVI DV  
WHERE DV.TEN_DV='Nghien cuu' AND DV.MA_DV=NV.MA_DV
```

```
SELECT NV.MA_NV, NV.TEN_NV  
FROM NHANVIEN NV INNER JOIN DONVI DV  
ON NV.MA_DV=DV.MA_DV  
WHERE DV.TEN_DV='Nghien cuu'
```
