
Nguyen Tat Thanh Institute of International Education (NIIE)

# DATABASE MANAGEMENT SYSTEMS

## (Creadits 3)

**MSc. Luong Tran Ngoc Khiet**

**May - 2021**

# Opening Course

MSc. Luong Tran Ngoc Khiet

NTT Institute of International Education (NIIE)

# Introduction

- Learning outcome (LO) :
  - Understand DB environment and DB development process.
  - Analysis Database model: Modeling data; Enhanced E-R model and business
  - Design Database: Logical database model and Relational model, physical database design.
  - Setting Database system: add constraint, relational method.

Database Management System

# Introduction

## ❖Learning Resources

[1] Satinder Bal Gupta & Aditya Mittal (2017). Introduction to Database Management System, 2nd Edition.

[2] Edward Pollack (2019). Dynamic SQL: Applications, Performance, and Security in Microsoft SQL Server.

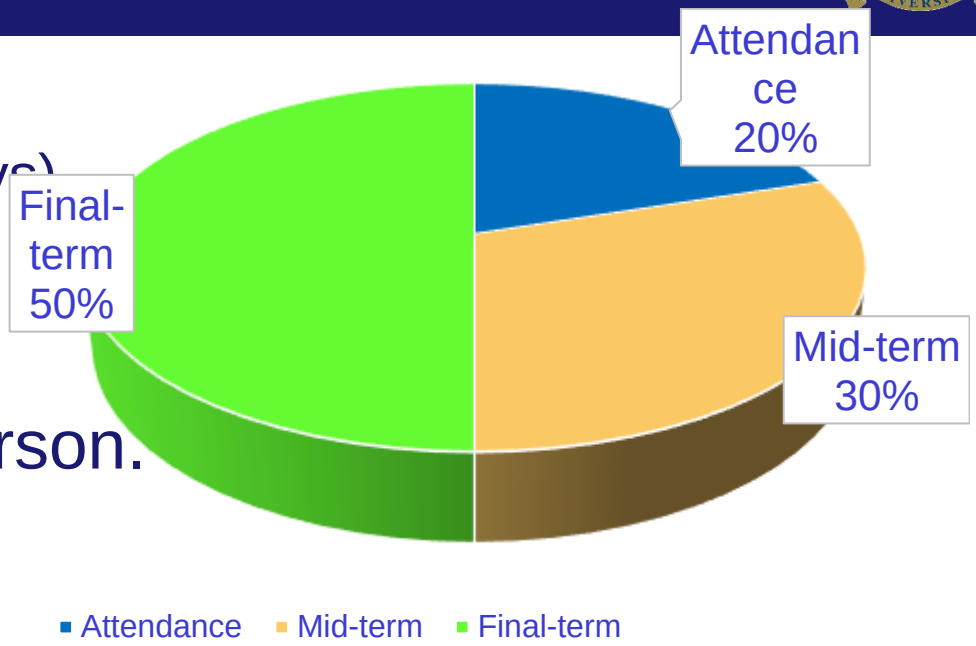
[3] John Wiley & Sons (2019), Database Management System. John Wiley & Sons.

[4] Anthony DeBarros (2018). Practical SQL. William Pollock.

[5] Rex Hogan (2018). A Practical Guide to Database Design Second Edition. Taylor & Francis Group

# Introduction (cont.)

- Scheduler: 48h (12 days)
- Prepare:
  - Group with 2 – 4 person.
  - Tools: SQL Server
- Email: khietltn@gmail.com with subject [NIIE_DBMS]
- Need to: Install sample database QLBongDa, QLSinhVien, QLDeAn.



# Discussion

# Programming with T_SQL

- Define variable:

  DECLARE @ten_bien datatype

For example:

  DECLARE @tuoi int

  DECLARE @mssv varchar(5)

  DECLARE @numCount int

- Rule: variable name has begin character @
- Commonly used data types: Using data types system except text, ntext, image

# Assign value to variable

- Using keyword Set:

  SET @ten_bien = value

For example:

  DECLARE @hoten nvarchar(20)

  SET @hoten = N‘Kim Hoàng Lộc’

Why start with the letter N?

# Assign value to variable

- Using keyword Select for variable simple:

  SELECT @ten_bien = value

For example:

  DECLARE @hoten nvarchar(20)

  SELECT @hoten = N‘Phan Nguyễn’

# Assign value to variable

- Using keyword Select for variable value in table:

```
SELECT @ten_bien = column_name
FROM   table_name
```

For example:

Find the employee with the maximum salary:

```
DECLARE @MaxSalary decimal(18,2)
SELECT   @MaxSalary = MAX(Luong)
FROM       NhanVien
```

# SQL Server Object – Local Variables

- Variables are used in the query as parameters.
- Using database sample below:

| HocSinh | | | |
|---|---|---|---|
| MaHS | TenHS | NgaySinh | DiaChi |
| 01 | Bảo | 10/10/1977 | 123 |
| 02 | Hải | 11/11/1981 | 456 |

| DiemThi | | | |
|---|---|---|---|
| MaHS | HocKy | NamHoc | Diem |
| 01 | 01 | 2001 | 10 |
| 03 | 01 | 2002 | 8 |

# SQL Server Object – Local Variables

For example:

List to student whose birthday is on '10/10/1977'

```
DECLARE  @NgaySinh datetime
SET      @NgaySinh = '10/10/1977'
SELECT * FROM HocSinh
WHERE NgaySinh = @NgaySinh
```

# SQL Server Object – Local Variables

For example:

Liệt kê danh sách các học sinh có địa chỉ là '123' và điểm thi lớn hơn 7

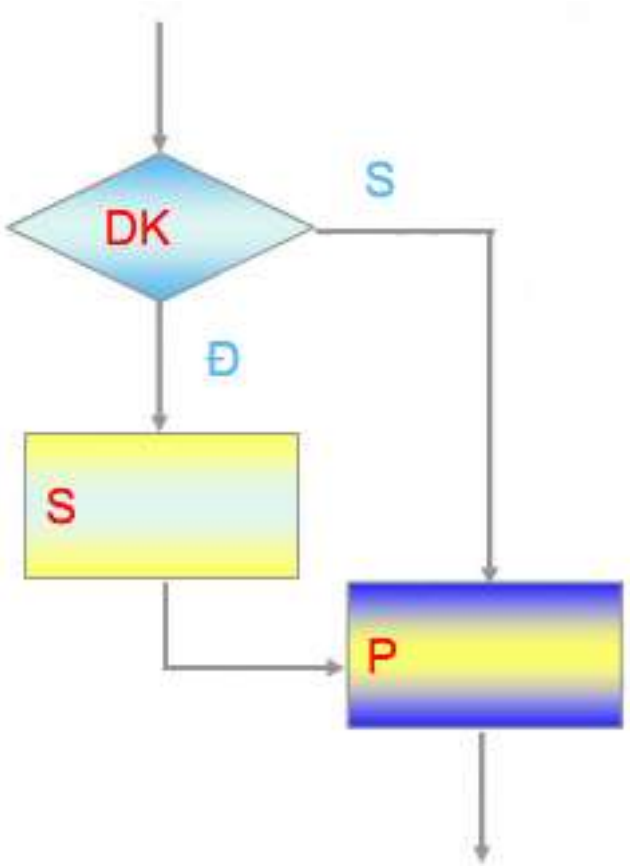
```
DECLARE  @DiaChi nvarchar(50), @Diem
   Decimal
SELECT@DiaChi='123', @Diem = 7
SELECT * FROM HocSinh JOIN DiemThi ON
                HocSinh.MaHS = DiemThi.MaHS
WHERE DiaChi = @DiaChi AND DiemThi >
   @Diem
```

# Conditional Statement

- Type simple

```
IF (conditional expression)
BEGIN
        command_true S
        or SQL Statement
END

command_continue P
or SQL Statement
```

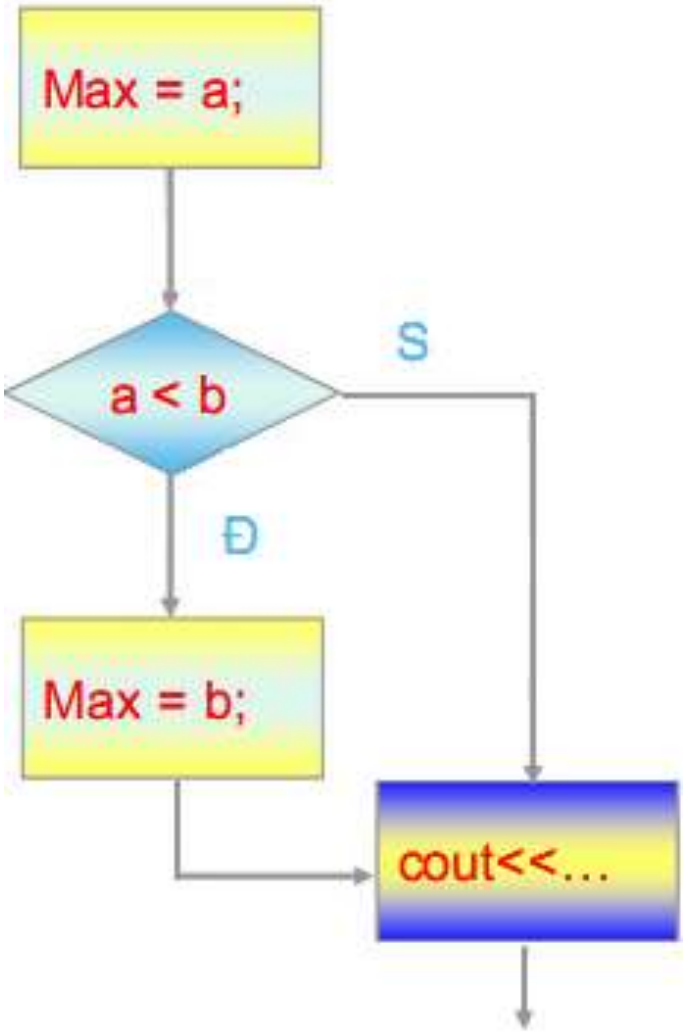


# Conditional Statement

❖For example:

Find Max 2 number input

```
DECLARE @a, @b, @Max int
SET @Max = @a
IF (@a < @b)
BEGIN
        SET @Max = @b
END
Print @Max
```

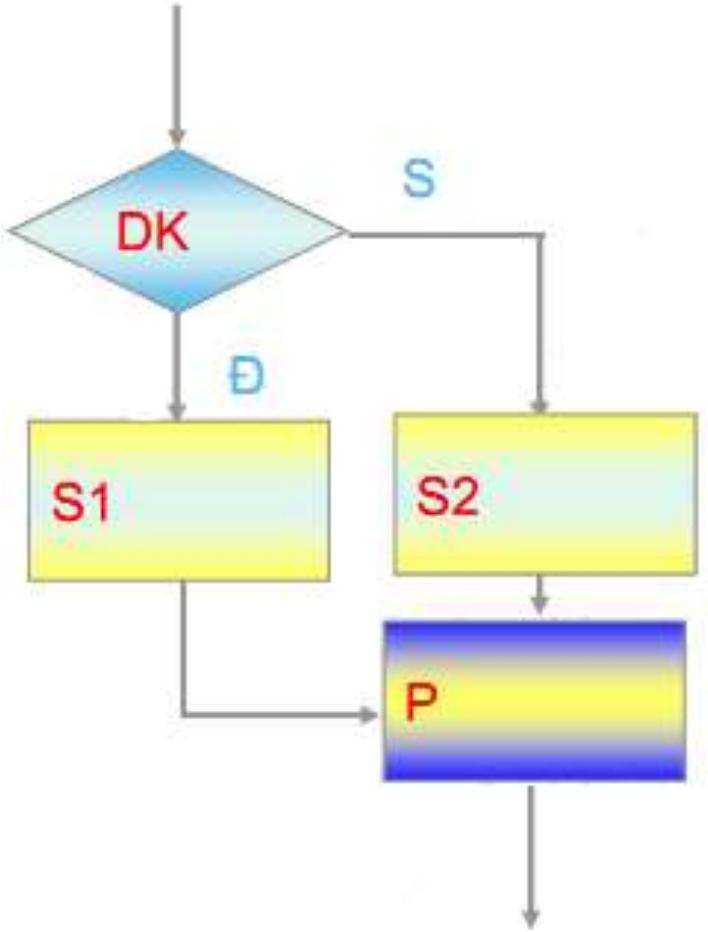


# Conditional Statement

❖ Type 2-way

```
IF (conditional expression)
BEGIN
        command S1
END
ELSE
BEGIN
        command S
END

command_continue P
or SQL Statement
```



# Conditional Statement

- For example:

Find Max 2 number input

```
DECLARE @a, @b, @Max int
IF (@a < @b)
BEGIN
        SELECT @Max = @b
END
ELSE
BEGIN
        SELECT @Max = @a
END
Print @Max
```

# CASE – As same Switch/Case

- Allows checking conditions and exporting information on a case-by-case basis

  Cho phép kiểm tra điều kiện và xuất thông tin theo từng trường hợp

- Syntax 1

```
CASE <tên cột>/<biểu thức>
        WHEN <giá trị> THEN <biểu thức>
        WHEN <giá trị> THEN <biểu thức>
        …
        [ELSE <biểu thức>]
END
```

# CASE

- Syntax 2

```
CASE WHEN <giá trị> THEN <biểu thức>
        WHEN <giá trị> THEN <biểu thức>
        …
        [ELSE <biểu thức>]
END
```

# For example Case

Ex 1:

```
SELECT TENCLB1, 'Kết quả' =
CASE
    WHEN (SOBANTHANG – SOBANTHUA > 0) THEN
    'Thắng'
    WHEN (SOBANTHANG – SOBANTHUA = 0) THEN
    'Hòa'
    WHEN (SOBANTHANG – SOBANTHUA < 0) THEN
    'Thua'
END,
TENCLB2
FROM vKETQUA
```

# For example Case (cont.)

CHITIETBAN(sohd, masach, slban, dgban)

Ví dụ 2:

```
SELECT masach,sum(slban)as tongslban,
    (CASE WHEN sum(slban)>10 THEN
                N‘Bán chạy'
    ELSE
                N‘Bán chậm'
    END) as thongtin
FROM CHITIETBAN
GROUP BY masach
```

# For example Case (cont.)

SACH (masach, tuasach, sotrang, matheloai, slton)

Case study:
Tăng số trang của những sách thuộc thể loại Tin học (1) +10,
Giảm số trang của những sách thuộc thể loại Toán học(4) -10

```
UPDATE sach SET sotrang=sotrang+
CASE WHEN matheloai=1 THEN
                    10
        WHEN matheloai=4 THEN
                   -10
        ELSE
                    0
END
```

# While

```
WHILE (bieuthuc logic)
BEGIN
        //Lenh/Khoi lenh S
END
//Lenh/khoi lenh sau while
```

- Example script using total s, with s = 1 + 2 + .. + n

```
DECLARE @i,@S INT
SELECT @i=1,@s=0
while(@i<=@n)
BEGIN
        SELECT @s = @s+@i;
        SELECT @i = @i+1;
END
PRINT @S
```

# While

- BREAK: Thoát khỏi vòng lặp WHILE
- CONTINUE: Thực hiện lần lặp mới

# Convert data type

- Get number to string:

  STR( số_thực, số_ký_tự [, Số lẻ])

*Ví dụ :*

- *SELECT STR(123);* => ***Result: '123'***
- *SELECT STR(123.5);* => ***Result: '124'***

  *(result is rounded because decimal places defaults to 0)*
- *SELECT STR(123.5, 5);* => ***Result: '124'***

  *(result is rounded because decimal places defaults to 0)*
- *SELECT STR(123.5, 5, 1);* => ***Result: '123.5'***
- *SELECT STR(123.456, 7, 3);* => ***Result: '123.456'***
- *SELECT STR(123.456, 7, 0);* => ***Result: '123'***

  *(result is rounded because decimal places is set to 0)*
- *SELECT STR(123.456, 7);* => ***Result: '123'***

  *(result is rounded because decimal places defaults to 0)*

# Convert data type

- Only Convert data type - đổi kiểu dữ liệu:

  CAST(Biểu thức AS Kiểu dữ liệu)

*Example: Cast (@tong as varchar(10))*

- Convert datatype and formatting - đổi kiểu dữ liệu và định dạng:

  CONVERT(Kiểu_dữ_liệu, Biểu_thức
  [, Định dạng])

*Example: Convert(char(10), Getdate(),105)*

# Convert data type with formatting datetime

http://www.sql-server-helper.com/tips/date-formats.aspx

| Định dạng | Hiển thị dữ liệu |
|---|---|
| 101 | Mm/dd/yyyy |
| 103 | Dd/mm/yyyy |
| 105 | Dd-mm-yyyy |
| 112 | Yyyymmdd |

# Important key word!

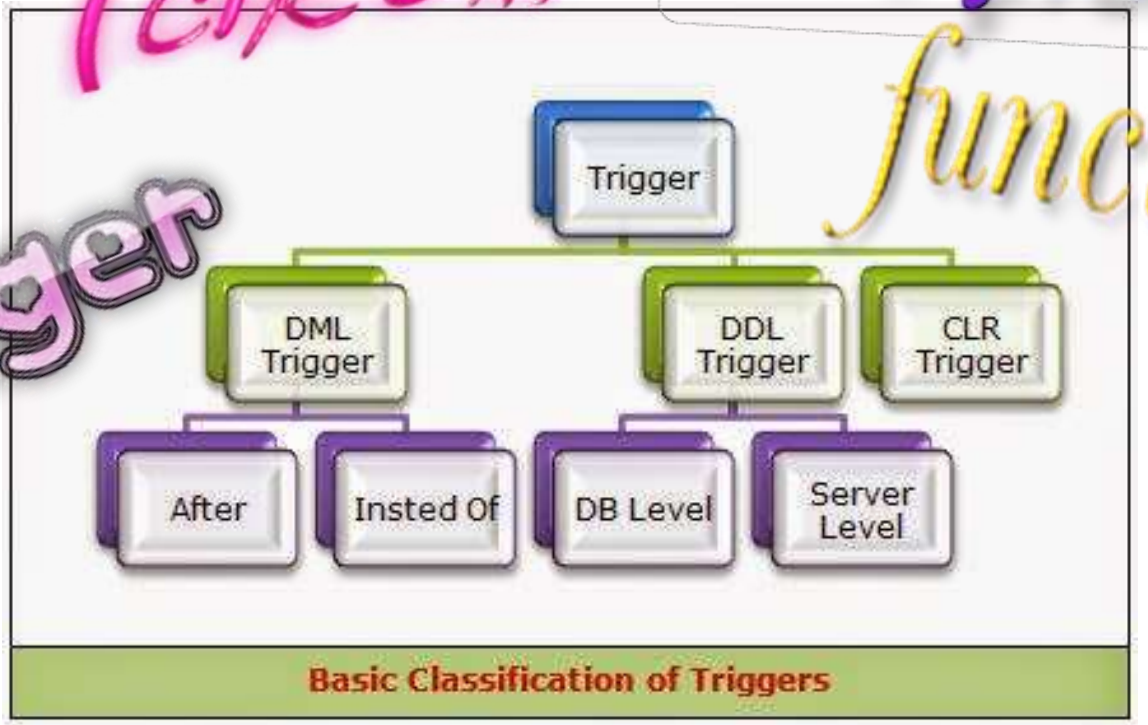
transaction

rule (check)

store procedure

function

trigger



# Discussion