**Types of Condition in MsSql**

1. **Case Statement Condition**

CASE

WHEN condition1 THEN result1

WHEN condition2 THEN result2

...

WHEN conditionN THEN resultN

ELSE default\_result

END

**\_\_\_\_\_\_\_\_ Table \_\_\_\_\_\_\_\_\_\_**

create table conditionTable(

id int primary key identity, f\_name varchar(25), l\_name varchar(25),

price int, quantity int,totalmarks int,obtainMarks int,

)

insert into conditionTable values ('f\_name','l\_name',1500,5,550,320),

('saqib','Ali',4440,25,11100,720),('a','b',2500,12,350,320),('f\_name','l\_name',1500,5,550,320),

('f\_name','l\_name',1500,5,550,320),('f\_name','l\_name',1500,5,550,320),('f\_name','l\_name',1500,5,550,320);

insert into conditionTable values ('x','y',130,4,550,320),('llll','mmmmm',1500,0,550,320),('f\_name','l\_name',1500,0,550,320)

**\_\_\_\_\_\_\_\_\_ 1. Single Case Statement \_\_\_\_\_\_\_**

select

f\_name,

price,

quantity,

case

when quantity = 0 then 'OutofStock'

else 'InStock'

end as 'stock\_Alert'

from conditionTable

**\_\_\_\_\_\_\_\_\_ 2. Multiple Case Statement \_\_\_\_\_\_\_**

//\_\_\_\_\_\_\_\_\_\_\_ 1. Quantity \_\_\_\_\_\_\_\_\_\_\_\_\_

select

f\_name,

price,

quantity,

case

when quantity = 0 then 'OutofStock'

when quantity < 5 then 'LowStock'

when quantity < 10 then 'GoodStock'

else 'InStock'

end as 'stock\_Alert'

from conditionTable

//\_\_\_\_\_\_\_\_\_\_\_ 2. Price \_\_\_\_\_\_\_\_\_\_\_\_\_

select

f\_name,

price,

case

when price < 2000 then 'Sastaa Product'

when price between 2000 and 10000 then 'Mid Prouct'

when price between 10000 and 200000 then 'Expansive Product'

else 'Not Found'

end as 'Rate'

from conditionTable

//\_\_\_\_\_\_\_\_\_\_\_ 3. Student Percentage and Grand \_\_\_\_\_\_\_\_\_\_\_\_\_

--\_\_\_ 1. WithOut subQuery \_\_\_\_

select

totalmarks,

obtainMarks,

(obtainMarks\*100)/totalmarks 'percentage',

case

when (obtainMarks\*100)/totalmarks >= 90 and (obtainMarks\*100)/totalmarks<=100 then 'A+1'

when (obtainMarks\*100)/totalmarks >= 80 and (obtainMarks\*100)/totalmarks<90 then 'A+'

when (obtainMarks\*100)/totalmarks >= 70 and (obtainMarks\*100)/totalmarks<80 then 'A'

when (obtainMarks\*100)/totalmarks >= 60 and (obtainMarks\*100)/totalmarks<70 then 'B'

when (obtainMarks\*100)/totalmarks >= 50 and (obtainMarks\*100)/totalmarks<60 then 'C'

when (obtainMarks\*100)/totalmarks >= 40 and (obtainMarks\*100)/totalmarks<50 then 'D'

else 'Fail'

end as 'Grad'

from conditionTable

CASE

WHEN condition1 THEN result1

WHEN condition2 THEN result2

...

WHEN conditionN THEN resultN

ELSE default\_result

END

--\_\_\_ 2. With subQuery \_\_\_\_

select

totalmarks,

obtainMarks,

per,

case

when per >= 90 and per<=100 then 'A+1'

when per >= 80 and per<90 then 'A+'

when per >= 70 and per<80 then 'A'

when per >= 60 and per<70 then 'B'

when per >= 50 and per<60 then 'C'

when per >= 40 and per<50 then 'D'

else 'Fail'

end as 'Grad'

from (

select

\*,

(obtainMarks\*100)/totalmarks per

from

conditionTable

) as studentRecord

CASE

WHEN condition1 THEN result1

WHEN condition2 THEN result2

...

WHEN conditionN THEN resultN

ELSE default\_result

END

1. **IF else Condition**

* **If Condition is Not Used in Select Query**
* **You can use with values and Condition**
* **Used in Triggers, Procedure, Views…**

**Syntax**

---\_\_\_\_ Syntax \_\_\_\_

if condition

begin

-- code of lines

end

else

begin

-- error message

end

***Example 1.***

Declare @variableValue int = 5;

if @variableValue > 10

Begin

print 'Account Varified'

end

else

begin

RAISERROR('Pin code not match', 16,1);

End

**\_\_\_\_\_\_ Exist & Not Exist \_\_\_\_\_\_\_**

CREATE TABLE students (

id INT PRIMARY KEY,

name VARCHAR(50),

age INT,

gender CHAR(1)

);

INSERT INTO students (id, name, age, gender)

VALUES

(1, 'Alice', 18, 'F'),

(2, 'Bob', 19, 'M'),

(3, 'Charlie', 20, 'M'),

(4, 'David', 19, 'M'),

(5, 'Emily', 21, 'F'),

(6, 'Frank', 22, 'M'),

(7, 'Grace', 20, 'F'),

(8, 'Henry', 19, 'M'),

(9, 'Isabel', 18, 'F'),

(10, 'John', 21, 'M');

---\_\_\_\_\_\_\_ Createing Variables \_\_\_\_\_\_\_\_\_\_\_

DECLARE @id INT = 1; -- the ID of the student to update

DECLARE @age INT = 20; -- the new age for the student

---\_\_\_\_\_\_\_ Checking Variables (id Exist Then update Age)\_\_\_\_\_\_\_\_\_\_\_

IF EXISTS (SELECT \* FROM students WHERE id = @id AND gender = 'F')

BEGIN

UPDATE students SET age = @age WHERE id = @id;

PRINT 'Age updated successfully.';

END

ELSE

BEGIN

PRINT 'Age cannot be updated for non-female student.';

END

**Example**

if exists(

select

grade\_of\_employee

from (

select

employee\_id,

salary\_amount,

case

when salary\_amount <= 60000 then 'Enternee'

else 'Heighly paid'

end as grade\_of\_employee

from salaries

) as empSalary

where grade\_of\_employee = 'Enternee'

)

begin

print 'Enternee Are Work'

end

else

begin

print 'noooo'

end