



# BTH2

📅 Ngày học @October 2, 2023

## C1

```
-- a. Thống kê tổng doanh thu bán hàng trong từng năm trong khoảng thời gian từ năm
-- @fromYear cho đến năm @toYear. Số liệu thống kê phải đầy đủ tất cả các năm
-- trong khoảng thời gian này, những năm không có doanh thu thì hiển thị số liệu doanh thu là 0.
DECLARE @fromYear int = 2018,
        @toYear int = 2020;

CREATE table #tblNam
(
    Nam int
)

DECLARE @tmpYear int = @fromYear;
WHILE (@tmpYear <= @toYear)
    BEGIN
        INSERT INTO #tblNam VALUES (@tmpYear);
        SET @tmpYear += 1;
    END
SELECT t1.Nam, ISNULL(t2.DoanhThu, 0) as DoanhThu
FROM #tblNam as t1
    LEFT JOIN
    (
        select YEAR(o.OrderDate) as Nam,
            SUM(od.Quantity * od.SalePrice) AS DoanhThu
        FROM Orders AS o JOIN OrderDetails AS od
        on o.OrderId = od.OrderId
        GROUP BY YEAR(o.OrderDate)
    ) as t2
    ON t1.Nam = t2.Nam
DROP TABLE #tblNam
GO

-- b. Thống kê tổng doanh thu bán hàng trong từng tháng của năm @year. Số liệu
-- thống kê phải đầy đủ 12 tháng của năm, những tháng không có doanh thu thì hiển
-- thị số liệu doanh thu là 0.
DECLARE @year int = 2017,
        @month int = 1;
DECLARE @tblThang Table
(
    Thang int
)
WHILE (@month <= 12)
```

```

BEGIN
    INSERT INTO @tblThang VALUES (@month);
    SET @month += 1;
END

SELECT t1.Thang, ISNULL(t2.DoanhThu, 0) as DoanhThu
FROM @tblThang as t1
LEFT JOIN
(
    select MONTH(o.OrderDate) as thang,
        SUM(od.Quantity * od.SalePrice) AS DoanhThu
    FROM Orders AS o JOIN OrderDetails AS od
    on o.OrderId = od.OrderId
    WHERE YEAR(o.OrderDate) = @year
    GROUP BY MONTH(o.OrderDate)
) as t2
ON t1.Thang = t2.thang
GO

-- c. Thống kê tổng doanh thu bán hàng trong từng ngày trong khoảng thời gian từ ngày
-- @fromDate cho đến ngày @toDate. Số liệu thống kê phải đầy đủ các ngày trong
-- khoảng thời gian này, những ngày không có doanh thu thì hiển thị số liệu doanh thu là 0.

-- DATEFROMPARTS(year, month, day)
DECLARE @fromDate date = DATEFROMPARTS(2017, 06, 12),
        @toDate date = DATEFROMPARTS(2017, 07, 22);

DECLARE @tblNgày Table
(
    ngay date
)

DECLARE @tmp date = DATEFROMPARTS(YEAR(@fromDate), MONTH(@fromDate), DAY(@fromDate))
WHILE (DATEDIFF(dd, @tmp, @toDate) != 0)
BEGIN
    insert into @tblNgày values (@tmp);
    set @tmp = DATEADD(day, 1, @tmp);
END

select t1.ngay, ISNULL(t2.revenue, 0) as Revenue
from @tblNgày AS t1
LEFT JOIN
(
    select o.OrderDate, SUM(od.Quantity * od.SalePrice) as Revenue
    from Orders as o join OrderDetails as od
    ON o.OrderId = od.OrderId
    group by o.OrderDate
) as t2
ON t1.ngay = t2.OrderDate
GO

```

## C2:

```

-- 2. Nhằm phân tích số liệu kinh doanh của các năm trong khoảng thời gian từ năm
-- @fromYear cho đến năm @toYear, ta cần bảng kết quả thống kê theo mẫu sau:

```

```

-- Năm      Doanh thu      Doanh thu lũy kế      Mức tăng giảm
-- 2010      1000           1000                   0
-- 2011      500           1500                   -500
-- 2012      700           2200                   200
-- ...      ...           ...                   ...
-- Yêu cầu số liệu thống kê phải đầy đủ các năm trong khoảng thời gian cần thống kê, trong đó:
--      - Doanh thu lũy kế = Doanh thu năm hiện tại + Doanh thu lũy kế năm trước
--      (Doanh thu lũy kế năm đầu tiên chính là doanh thu của năm đó)
--      - Mức tăng giảm = Doanh thu năm hiện tại - Doanh thu năm trước
--      (Mức tăng giảm của năm đầu tiên là 0)

DECLARE @fromYear INT = 2017,
        @toYear INT = 2020;
DECLARE @tmpYear INT, @revenue MONEY, @cumulativeRevenue MONEY, @change MONEY;

DECLARE @tbldoanhThu TABLE (
    Nam INT,
    DoanhThu MONEY,
    DoanhThuLuyKe MONEY,
    MucTangGiam MONEY
)

SET @revenue = ( SELECT SUM(Quantity * SalePrice)
    FROM OrderDetails AS od JOIN Orders AS o
    ON od.OrderId = o.OrderId
    WHERE YEAR(OrderDate) = @fromYear
    GROUP BY YEAR(OrderDate));
-- SELECT @revenue

INSERT INTO @tbldoanhThu (Nam, DoanhThu, DoanhThuLuyKe, MucTangGiam)
VALUES      (@fromYear, @revenue, @revenue, 0);

SET @tmpYear = @fromYear + 1;

WHILE (@tmpYear <= @toYear)
BEGIN
    -- Tính toán doanh thu cho năm hiện tại và năm trước
    SET @revenue = (SELECT SUM(od.Quantity * od.SalePrice)
        FROM Orders AS o JOIN OrderDetails AS od
        ON o.OrderId = od.OrderId
        WHERE YEAR(o.OrderDate) = @tmpYear
        GROUP BY YEAR(o.OrderDate));

    SET @cumulativeRevenue = @revenue + (SELECT DoanhThuLuyKe
        FROM @tbldoanhThu
        WHERE Nam = @tmpYear - 1);

    SET @change = @revenue - (SELECT SUM(od.Quantity * od.SalePrice)
        FROM Orders AS o JOIN OrderDetails AS od
        ON o.OrderId = od.OrderId
        WHERE YEAR(o.OrderDate) = @tmpYear - 1
        GROUP BY YEAR(o.OrderDate));

    -- Lưu kết quả vào bảng
    INSERT INTO @tbldoanhThu (Nam, DoanhThu, DoanhThuLuyKe, MucTangGiam)
    VALUES (@tmpYear, @revenue, @cumulativeRevenue, @change);

```

```

SET @tmpYear = @tmpYear + 1;
END;

SELECT * FROM @tbldoanhThu;

```

## C3

```

-- 3. Lập bảng thống kê doanh thu, doanh thu lũy kế và mức tăng giảm của từng tháng
-- trong năm @year theo mẫu:
DECLARE @year INT = 2017;
DECLARE @thang INT = 1;
DECLARE @tmpMonth INT, @revenue MONEY, @cumulativeRevenue MONEY, @change MONEY;

DECLARE @tbldoanhThuThang TABLE (
    Thang INT,
    DoanhThu MONEY,
    DoanhThuLuyKe MONEY,
    MucTangGiam MONEY
)

SET @revenue = (SELECT SUM(Quantity * SalePrice)
    FROM OrderDetails AS od JOIN Orders AS o
    ON od.OrderId = o.OrderId
    WHERE MONTH(OrderDate) = @thang
    GROUP BY MONTH(OrderDate));
-- SELECT @revenue

INSERT INTO @tbldoanhThuThang (Thang, DoanhThu, DoanhThuLuyKe, MucTangGiam)
VALUES (@thang, @revenue, @revenue, 0);

SET @tmpMonth = @thang + 1;

WHILE (@tmpMonth <= 12)
BEGIN
    -- Tính toán doanh thu cho năm hiện tại và năm trước
    SET @revenue = (SELECT SUM(od.Quantity * od.SalePrice)
        FROM Orders AS o JOIN OrderDetails AS od
        ON o.OrderId = od.OrderId
        WHERE MONTH(o.OrderDate) = @tmpMonth
        GROUP BY MONTH(o.OrderDate));

    SET @cumulativeRevenue = @revenue + (SELECT DoanhThuLuyKe
        FROM @tbldoanhThuThang
        WHERE Thang = @tmpMonth - 1);

    SET @change = @revenue - (SELECT SUM(od.Quantity * od.SalePrice)
        FROM Orders AS o JOIN OrderDetails AS od
        ON o.OrderId = od.OrderId
        WHERE MONTH(o.OrderDate) = @tmpMonth - 1
        GROUP BY MONTH(o.OrderDate));

    -- Lưu kết quả vào bảng
    INSERT INTO @tbldoanhThuThang (Thang, DoanhThu, DoanhThuLuyKe, MucTangGiam)
    VALUES (@tmpMonth, @revenue, @cumulativeRevenue, @change);

```

```

SET @tmpMonth = @tmpMonth + 1;
END;

SELECT * FROM @tbldoanhThuThang;

```

## C4

```

-- Lập bảng thống kê doanh thu, doanh thu lũy kế và mức tăng giảm của từng ngày trong
-- khoảng thời gian từ ngày @fromDate đến ngày @toDate theo mẫu:
-- 3. Lập bảng thống kê doanh thu, doanh thu lũy kế và mức tăng giảm của từng tháng
-- trong năm @year theo mẫu:
DECLARE @year INT = 2017;
DECLARE @fromDate date = DATEFROMPARTS(2017, 06, 12),
        @toDate date = DATEFROMPARTS(2017, 07, 22);

DECLARE @tmpDate date = DATEFROMPARTS(YEAR(@fromDate), MONTH(@fromDate), DAY(@fromDate)),
        @revenue MONEY, @cumulativeRevenue MONEY, @change MONEY;

DECLARE @tbldoanhThuNgay TABLE (
    Ngay date,
    DoanhThu MONEY,
    DoanhThuLuyKe MONEY,
    MucTangGiam MONEY
)

SET @revenue = (SELECT SUM(Quantity * SalePrice)
    FROM OrderDetails AS od JOIN Orders AS o
    ON od.OrderId = o.OrderId
    WHERE DATEDIFF(dd, OrderDate, @tmpDate) = 0
    GROUP BY DAY(OrderDate));
-- SELECT @revenue

INSERT INTO @tbldoanhThuNgay (Ngay, DoanhThu, DoanhThuLuyKe, MucTangGiam)
VALUES (@tmpDate, @revenue, @revenue, 0);

SET @tmpDate = DATEADD(DAY, 1, @tmpDate);

WHILE (DATEDIFF(dd, @tmpDate, @toDate) != 0)
BEGIN
    -- Tính toán doanh thu cho năm hiện tại và năm trước
    SET @revenue = (SELECT SUM(od.Quantity * od.SalePrice)
        FROM Orders AS o JOIN OrderDetails AS od
        ON o.OrderId = od.OrderId
        WHERE DATEDIFF(dd, OrderDate, @tmpDate) = 0
        GROUP BY DAY(OrderDate));

    SET @cumulativeRevenue = @revenue + (SELECT DoanhThuLuyKe
        FROM @tbldoanhThuNgay
        WHERE Ngay = DATEADD(DAY, -1, @tmpDate))

    SET @change = @revenue - (SELECT SUM(od.Quantity * od.SalePrice)
        FROM Orders AS o JOIN OrderDetails AS od
        ON o.OrderId = od.OrderId
        WHERE o.OrderDate = DATEADD(DAY, -1, @tmpDate)
        GROUP BY DAY(o.OrderDate));

```

```

-- Lưu kết quả vào bảng
INSERT INTO @tbldoanhThuNgay (Ngay, DoanhThu, DoanhThuLuyKe, MucTangGiam)
VALUES (@tmpDate, @revenue, @cumulativeRevenue, @change);

SET @tmpDate = DATEADD(DAY, 1, @tmpDate);
END;

SELECT * FROM @tbldoanhThuNgay;

```

## C5

```

-- Câu 5: Với đầu vào là tháng @month năm @year, hãy lập một bảng thống kê tổng doanh
-- thu bán hàng trong từng ngày của hàng tuần và tổng doanh thu hàng tuần theo mẫu
-- sau (giả sử với @month = 10 và @year = 2023)
-- Trong đó: TuNgay và DenNgay là ngày đầu tuần và ngày cuối tuần (ngày đầu tuần được tính từ thứ hai)
DECLARE @thang int = 2,
        @nam int = 2018;

--SELECT o.OrderDate as Ngay,
--      SUM(od.Quantity * od.SalePrice)
--FROM Orders as o JOIN OrderDetails as od
--ON o.OrderId = od.OrderId
--GROUP BY o.OrderDate

DECLARE @ngayDauThang date = DATEFROMPARTS(@nam, @thang, 1);
DECLARE @ngayCuoiThang date = DATEADD(DAY, -1, DATEADD(MONTH, 1, @ngayDauThang));

DECLARE @ngayDauTuan date = DATEADD(DAY
    , CHOOSE(DATEPART(WEEKDAY, @ngayDauThang), -6, 0, -1, -2, -3, -4, -5)
    , @ngayDauThang);

DECLARE @ngayCuoiTuan date = DATEADD(DAY, 6, @ngayDauTuan);

WITH cte_Tuan AS
(
    SELECT @ngayDauTuan as TuNgay, @ngayCuoiTuan as DenNgay
    UNION ALL
    SELECT DATEADD(DAY, 7, TuNgay), DATEADD(DAY, 7, DenNgay)
    FROM cte_Tuan
    WHERE DATEADD(DAY, 7, TuNgay) <= @ngayCuoiThang
)
, cte_DoanhThuNgay as
(
    SELECT o.OrderDate as Ngay,
           SUM(od.Quantity * od.SalePrice) as DoanhThu
    FROM Orders as o JOIN OrderDetails as od
    ON o.OrderId = od.OrderId
    WHERE MONTH(o.OrderDate) = @thang AND YEAR(o.OrderDate) = @nam
    GROUP BY o.OrderDate
)
SELECT t1.TuNgay, t1.DenNgay,
       SUM(IIF(DATEPART(WEEKDAY, t2.Ngay) = 2, t2.DoanhThu, 0)) as Thu2,
       SUM(IIF(DATEPART(WEEKDAY, t2.Ngay) = 3, t2.DoanhThu, 0)) as Thu3,

```

```

SUM(IIF(DATEPART(WEEKDAY, t2.Ngay) = 4, t2.DoanhThu, 0)) as Thu4,
SUM(IIF(DATEPART(WEEKDAY, t2.Ngay) = 5, t2.DoanhThu, 0)) as Thu5,
SUM(IIF(DATEPART(WEEKDAY, t2.Ngay) = 6, t2.DoanhThu, 0)) as Thu6,
SUM(IIF(DATEPART(WEEKDAY, t2.Ngay) = 7, t2.DoanhThu, 0)) as Thu7,
SUM(IIF(DATEPART(WEEKDAY, t2.Ngay) = 1, t2.DoanhThu, 0)) as ChuNhat,
SUM(t2.DoanhThu) as TongTrongTuan

FROM cte_Tuan as t1
LEFT JOIN cte_DoanhThuNgay AS t2
ON t1.TuNgay < t2.Ngay AND t1.DenNgay >= t2.Ngay
GROUP BY t1.TuNgay, t1.DenNgay

```