1. **Sử dụng câu SELECT**
2. **Số tiền lương mà công ty cấp cho mỗi nhân viên là:**

SELECT

MANHANVIEN,

HO,

TEN,

LUONGCOBAN + PHUCAP AS TONGLUONG

FROM NHANVIEN;

1. **Trong đơn đặt hàng số 3 đặt mua những mặt hàng nào và số tiền mà khách hàng phải trả cho mỗi mặt hàng là bao nhiêu (số tiền phải trả được tính theo công thức (SOLUONG x GIABAN – SOLUONG x GIABAN x MUCGIAMGIA/100).**

SELECT

CHITIETDATHANG.MAHANG,

MATHANG.TENHANG,

SOLUONG,

GIABAN,

MUCGIAMGIA,

(SOLUONG \* GIABAN) - (SOLUONG \* GIABAN \* MUCGIAMGIA / 100) AS TONGTIENPHAITRA

FROM CHITIETDATHANG

JOIN MATHANG ON CHITIETDATHANG.MAHANG = MATHANG.MAHANG

WHERE CHITIETDATHANG.SOHODON = 3;

1. **Hãy cho biết có những khách hàng nào lại chính là đối tác cung cấp hàng của công ty (tức là có cùng tên giao dịch).**

SELECT

KHACHHANG.MAKHACHHANG,

KHACHHANG.TENGIAODICH AS TENKHACHHANG,

NHACUNGCAP.MACUNGCAP,

NHACUNGCAP.TENGIAODICH AS TENDOITAC

FROM KHACHHANG

JOIN NHACUNGCAP ON KHACHHANG.TENGIAODICH = NHACUNGCAP.TENGIAODICH;

1. **Trong công ty có những nhân viên nào có cùng ngày sinh?**

SELECT

A.MANHANVIEN AS NHANVIEN1\_ID,

A.HO AS NHANVIEN1\_HO,

A.TEN AS NHANVIEN1\_TEN,

B.MANHANVIEN AS NHANVIEN2\_ID,

B.HO AS NHANVIEN2\_HO,

B.TEN AS NHANVIEN2\_TEN,

A.NGAYSINH

FROM NHANVIEN A, NHANVIEN B

WHERE

A.MANHANVIEN < B.MANHANVIEN AND

A.NGAYSINH = B.NGAYSINH

ORDER BY A.NGAYSINH;

1. **Tổng số tiền mà khách hàng phải trả cho mỗi đơn đặt hàng là bao nhiêu?**

SELECT

DONDATHANG.SOHODON,

SUM((SOLUONG \* GIABAN) - (SOLUONG \* GIABAN \* MUCGIAMGIA / 100)) AS TONGTIENPHAITRA

FROM DONDATHANG

JOIN CHITIETDATHANG ON DONDATHANG.SOHODON = CHITIETDATHANG.SOHODON

JOIN MATHANG ON CHITIETDATHANG.MAHANG = MATHANG.MAHANG

GROUP BY DONDATHANG.SOHODON

ORDER BY DONDATHANG.SOHODON;

1. **Hãy cho biết mỗi một khách hàng đã phải bỏ ra bao nhiêu tiền để đặt mua hàng của công ty?**

SELECT

KHACHHANG.MAKHACHHANG,

KHACHHANG.TENGIAODICH,

SUM((CHITIETDATHANG.SOLUONG \* MATHANG.GIABAN) - (CHITIETDATHANG.SOLUONG \* MATHANG.GIABAN \* CHITIETDATHANG.MUCGIAMGIA / 100)) AS TONGTIENPHAITRA

FROM KHACHHANG

JOIN DONDATHANG ON KHACHHANG.MAKHACHHANG = DONDATHANG.MAKHACHHANG

JOIN CHITIETDATHANG ON DONDATHANG.SOHODON = CHITIETDATHANG.SOHODON

JOIN MATHANG ON CHITIETDATHANG.MAHANG = MATHANG.MAHANG

GROUP BY KHACHHANG.MAKHACHHANG, KHACHHANG.TENGIAODICH

ORDER BY TONGTIENPHAITRA DESC;

1. **Mỗi một nhân viên của công ty đã lập bao nhiêu đơn đặt hàng (nếu nhân viên chưa hề lập một hoá đơn nào thì cho kết quả là 0).**

SELECT

NHANVIEN.MANHANVIEN,

NHANVIEN.HO,

NHANVIEN.TEN,

COALESCE(COUNT(DONDATHANG.SOHODON), 0) AS SOLUONGDONDATHANG

FROM NHANVIEN

LEFT JOIN DONDATHANG ON NHANVIEN.MANHANVIEN = DONDATHANG.MANHANVIEN

GROUP BY NHANVIEN.MANHANVIEN, NHANVIEN.HO, NHANVIEN.TEN

ORDER BY SOLUONGDONDATHANG DESC;

1. **Hãy cho biết tổng số tiền lời mà công ty thu được từ mỗi mặt hàng trong năm 2022.**

Ta có công thức tính lợi nhuận là:

Lợi nhuận = (Giá bán – Giá vốn) \* Số lượng bán được

SELECT

MATHANG.MAHANG,

MATHANG.TENHANG,

SUM((MATHANG.GIABAN - MATHANG.GIAVON) \* CHITIETDATHANG.SOLUONG) AS TONGLOINHUAN

FROM MATHANG

JOIN CHITIETDATHANG ON MATHANG.MAHANG = CHITIETDATHANG.MAHANG

JOIN DONDATHANG ON CHITIETDATHANG.SOHODON = DONDATHANG.SOHODON

WHERE YEAR(DONDATHANG.NGAYDATHANG) = 2022

GROUP BY MATHANG.MAHANG, MATHANG.TENHANG

ORDER BY TONGLOINHUAN DESC;

1. **Nhân viên nào của công ty bán được số lượng hàng nhiều nhất và số lượng hàng bán được của những nhân viên này là bao nhiêu?**

WITH EmployeeSales AS (

SELECT

NHANVIEN.MANHANVIEN,

NHANVIEN.HO,

NHANVIEN.TEN,

SUM(CHITIETDATHANG.SOLUONG) AS TONGSOLUONG

FROM NHANVIEN

JOIN DONDATHANG ON NHANVIEN.MANHANVIEN = DONDATHANG.MANHANVIEN

JOIN CHITIETDATHANG ON DONDATHANG.SOHODON = CHITIETDATHANG.SOHODON

GROUP BY NHANVIEN.MANHANVIEN, NHANVIEN.HO, NHANVIEN.TEN

)

SELECT

MANHANVIEN,

HO,

TEN,

TONGSOLUONG

FROM EmployeeSales

WHERE TONGSOLUONG = (SELECT MAX(TONGSOLUONG) FROM EmployeeSales);

1. **Số tiền nhiều nhất mà mỗi khách hàng đã từng bỏ ra để đặt hàng trong các đơn đặthàng là bao nhiêu?**

WITH CustomerOrderTotal AS (

SELECT

DONDATHANG.MAKHACHHANG,

DONDATHANG.SOHODON,

SUM(CHITIETDATHANG.SOLUONG \* CHITIETDATHANG.GIABAN - (CHITIETDATHANG.SOLUONG \* CHITIETDATHANG.GIABAN \* CHITIETDATHANG.MUCGIAMGIA / 100)) AS TOTAL

FROM DONDATHANG

JOIN CHITIETDATHANG ON DONDATHANG.SOHODON = CHITIETDATHANG.SOHODON

GROUP BY DONDATHANG.MAKHACHHANG, DONDATHANG.SOHODON

)

SELECT

MAKHACHHANG,

MAX(TOTAL) AS MAX\_TOTAL

FROM CustomerOrderTotal

GROUP BY MAKHACHHANG;

1. **Hãy cho biết mỗi một loại hàng bao gồm những mặt hàng nào, tổng số lượng hàng của mỗi loại và tổng số lượng của tất cả các mặt hàng hiện có trong công ty là bao nhiêu?**

WITH ProductCount AS (

SELECT

LOAIHANG.TENLOAIHANG,

MATHANG.MAHANG,

MATHANG.TENHANG,

MATHANG.SOLUONG AS SOLUONGHANG

FROM LOAIHANG

JOIN MATHANG ON LOAIHANG.MALOAIHANG = MATHANG.MALOAIHANG

)

SELECT

TENLOAIHANG,

GROUP\_CONCAT(TENHANG) AS MATHANG\_TRONG\_LOAI,

SUM(SOLUONGHANG) AS TONGSOLUONG\_LOAI

FROM ProductCount

GROUP BY TENLOAIHANG

UNION ALL

SELECT

'TONG CONG' AS TENLOAIHANG,

NULL AS MATHANG\_TRONG\_LOAI,

SUM(SOLUONGHANG) AS TONGSOLUONG\_LOAI

FROM ProductCount;

1. **Thống kê xem trong năm 2022, mỗi một mặt hàng trong mỗi tháng và trong cả năm bán được với số lượng bao nhiêu?**

WITH MonthlySales AS (

SELECT

MATHANG.MAHANG,

MATHANG.TENHANG,

MONTH(DONDATHANG.NGAYDATHANG) AS THANG,

SUM(CHITIETDATHANG.SOLUONG) AS SOLUONG\_BAN

FROM MATHANG

JOIN CHITIETDATHANG ON MATHANG.MAHANG = CHITIETDATHANG.MAHANG

JOIN DONDATHANG ON CHITIETDATHANG.SOHODON = DONDATHANG.SOHODON

WHERE YEAR(DONDATHANG.NGAYDATHANG) = 2022

GROUP BY MATHANG.MAHANG, MATHANG.TENHANG, MONTH(DONDATHANG.NGAYDATHANG)

)

SELECT

MAHANG,

TENHANG,

THANG,

SOLUONG\_BAN

FROM MonthlySales

UNION ALL

SELECT

MAHANG,

TENHANG,

'Tong trong nam 2022' AS THANG,

SUM(SOLUONG\_BAN)

FROM MonthlySales

GROUP BY MAHANG, TENHANG

ORDER BY MAHANG, THANG;

1. **Sử dụng câu lệnh UPDATE để thực hiện các yêu cầu sau**
2. **Cập nhật lại giá trị trường NGAYCHUYENHANG của những bản ghi có NGAYCHUYENHANG chưa xác định (NULL) trong bảng DONDATHANG bằng với giá trị của trường NGAYDATHANG.**

UPDATE DONDATHANG

SET NGAYCHUYENHANG = NGAYDATHANG

WHERE NGAYCHUYENHANG IS NULL;

1. **Tăng số lượng hàng của những mặt hàng do công ty Hải Hoàng cung cấp lên gấp ba.**

UPDATE MATHANG

SET SOLUONG = SOLUONG \* 3

WHERE MACONGTY IN (

SELECT MACONGTY

FROM NHA\_CUNGCAP

WHERE TENCONGTY = 'Hải Hoàng'

);

1. **Cập nhật giá trị của trường NOIGIAOHANG trong bảng DONDATHANG bằng địa chỉ của khách hàng đối với những đơn đặt hàng chưa xác định được nơi giao hàng (giá trị trường NOIGIAOHANG bằng NULL).**

UPDATE DONDATHANG

JOIN KHACHHANG ON DONDATHANG.MAKHACHHANG = KHACHHANG.MAKHACHHANG

SET DONDATHANG.NOIGIAOHANG = KHACHHANG.DIACHI

WHERE DONDATHANG.NOIGIAOHANG IS NULL;

1. **Cập nhật lại dữ liệu trong bảng KHACHHANG sao cho nếu tên công ty và tên giao dịch của khách hàng trùng với tên công ty và tên giao dịch của một nhà cung cấp nào đó thì địa chỉ, điện thoại, fax và e-mail phải giống nhau.**

UPDATE KHACHHANG

JOIN NHACUNGCAP ON KHACHHANG.TENCONGTY = NHACUNGCAP.TENCONGTY AND KHACHHANG.TENGIAODICH = NHACUNGCAP.TENGIAODICH

SET

KHACHHANG.DIACHI = NHACUNGCAP.DIACHI,

KHACHHANG.DIENTHOAI = NHACUNGCAP.DIENTHOAI,

KHACHHANG.FAX = NHACUNGCAP.FAX,

KHACHHANG.EMAIL = NHACUNGCAP.EMAIL;

1. **Tăng lương lên gấp đôi cho những nhân viên bán được số lượng hàng nhiều hơn 50.**

UPDATE NHANVIEN

SET LUONGCOBAN = LUONGCOBAN \* 2

WHERE MANHANVIEN IN (

SELECT MANHANVIEN

FROM CHITIETDATHANG

GROUP BY MANHANVIEN

HAVING SUM(SOLUONG) > 50

);

1. **Tăng phụ cấp lên bằng 50% lương cho những nhân viên bán được hàng nhiều nhất.**

UPDATE NHANVIEN

SET PHUCAP = LUONGCOBAN \* 0.5

WHERE MANHANVIEN IN (

SELECT MANHANVIEN

FROM CHITIETDATHANG

GROUP BY MANHANVIEN

HAVING SUM(SOLUONG) = (

SELECT MAX(totalSOLUONG)

FROM (

SELECT SUM(SOLUONG) AS totalSOLUONG

FROM CHITIETDATHANG

GROUP BY MANHANVIEN

) AS sub

)

);

1. **Giảm 30% lương của những nhân viên không lập được bất kỳ đơn đặt hàng nào.**

UPDATE NHANVIEN

SET LUONGCOBAN = LUONGCOBAN \* 0.7

WHERE MANHANVIEN NOT IN (

SELECT DISTINCT MANHANVIEN

FROM DONDATHANG

);

1. **Giả sử trong bảng DONDATHANG có thêm trường SOTIEN cho biết số tiền mà khách hàng phải trả trong mỗi đơn đặt hàng. Hãy tính giá trị cho trường này.**

UPDATE DONDATHANG

SET SOTIEN = (

SELECT SUM(SOLUONG \* GIABAN - SOLUONG \* GIABAN \* MUCGIAMGIA / 100)

FROM CHITIETDATHANG

WHERE CHITIETDATHANG.SOHOADON = DONDATHANG.SOHOADON

)

1. **Sử dụng câu lệnh DELETE để thực hiện yêu cầu sau**
2. **Xoá khỏi bảng NHANVIEN những nhân viên đã làm việc trong công ty quá 30 năm.**

DELETE FROM NHANVIEN

WHERE DATEDIFF(YEAR, NGAYBATDAU, GETDATE()) > 30;

1. **Xoá khỏi bảng KHACHHANG những khách hàng hiện không có bất kỳ đơn đặt hàng nào cho công ty.**

DELETE FROM KHACHHANG

WHERE MAKHACHHANG NOT IN

(SELECT DISTINCT MAKHACHHANG FROM DONDATHANG);

1. **Xoá khỏi bảng MATHANG những mặt hàng có số lượng bằng 0 và không được đặt mua trong bất kỳ đơn đặt hàng nào.**

DELETE FROM MATHANG

WHERE SOLUONG = 0 AND MAHANG NOT IN

(SELECT DISTINCT MAHANG FROM CHITIETDATHANG);

1. **Thủ tục hàm và Trigger**
2. **Tạo thủ tục lưu trữ để thông qua thủ tục này có thể bổ sung thêm một bản ghi mới cho bảng MATHANG (thủ tục phải thực hiện kiểm tra tính hợp lệ của dữ liệu cần bổ sung: không trùng khoá chính và đảm bảo toàn vẹn tham chiếu).**

CREATE PROCEDURE InsertIntoMathang(

@MAHANG INT,

@TENHANG NVARCHAR(255),

-- Các trường dữ liệu khác của bảng MATHANG nếu có

)

AS

BEGIN

-- Kiểm tra xem MAHANG đã tồn tại chưa

IF EXISTS(SELECT 1 FROM MATHANG WHERE MAHANG = @MAHANG)

BEGIN

RAISERROR('Mã hàng đã tồn tại!', 16, 1)

RETURN

END

-- Thêm dữ liệu mới

INSERT INTO MATHANG(MAHANG, TENHANG /\*, Các trường khác\*/)

VALUES(@MAHANG, @TENHANG /\*, Các giá trị khác\*/)

PRINT 'Thêm dữ liệu thành công!'

END

1. **Tạo thủ tục lưu trữ có chức năng thống kê tổng số lượng hàng bán được của một mặt hàng có mã bất kỳ (mã mặt hàng cần thống kê là tham số của thủ tục).**

CREATE PROCEDURE ThongKeSoLuongBan(

@MAHANG INT

)

AS

BEGIN

-- Kiểm tra xem MAHANG có tồn tại trong bảng MATHANG hay không

IF NOT EXISTS(SELECT 1 FROM MATHANG WHERE MAHANG = @MAHANG)

BEGIN

RAISERROR('Mã hàng không tồn tại!', 16, 1)

RETURN

END

-- Thống kê tổng số lượng hàng bán được

SELECT SUM(SOLUONGBAN) AS TotalSold

FROM BANGBANHANG -- Tên bảng chứa thông tin về số lượng bán có thể khác, hãy thay đổi tương ứng

WHERE MAHANG = @MAHANG

END

1. **Viết hàm trả về một bảng trong đó cho biết tổng số lượng hàng bán được của mỗi mặt hàng. Sử dụng hàm này để thống kê xem tổng số lượng hàng (hiện có và đã bán) của mỗi mặt hàng là bao nhiêu?**

Hàm trả về một bảng cho biết tổng số lượng bán được của mỗi mặt hàng:

CREATE FUNCTION fn\_TongSoLuongBan()

RETURNS TABLE

AS

RETURN

(

SELECT

MAHANG,

SUM(SOLUONGBAN) AS TotalSold

FROM BANGBANHANG -- Tên bảng chứa thông tin về số lượng bán có thể khác, hãy thay đổi tương ứng

GROUP BY MAHANG

)

Sau khi tạo hàm, có thể sử dụng nó để thống kê tổng số lượng hàng (hiện có và đã bán) của mỗi mặt hàng sau:

SELECT

M.MAHANG,

M.TENHANG,

M.SOLUONGHANG as 'Số lượng hiện có',

ISNULL(F.TotalSold, 0) as 'Số lượng đã bán',

M.SOLUONGHANG + ISNULL(F.TotalSold, 0) as 'Tổng số lượng'

FROM

MATHANG M

LEFT JOIN

fn\_TongSoLuongBan() F ON M.MAHANG = F.MAHANG

ORDER BY

M.MAHANG

1. **Viết trigger cho bảng CHITIETDATHANG theo yêu cầu sau:**

**- Khi một bản ghi mới được bổ sung vào bảng này thì giảm số lượng hàng hiện có nếu số lượng hàng hiện có lớn hơn hoặc bằng số lượng hàng được bán ra. Ngược lại thì huỷ bỏ thao tác bổ sung.**

**- Khi cập nhật lại số lượng hàng được bán, kiểm tra số lượng hàng được cập nhật lại có phù hợp hay không (số lượng hàng bán ra không được vượt quá số lượng hàng hiện có và không được nhỏ hơn 1). Nếu dữ liệu hợp lệ thì giảm (hoặc tăng) số lượng hàng hiện có trong công ty, ngược lại thì huỷ bỏ thao tác cập nhật.**

CREATE TRIGGER tr\_UpdateChiTietDatHang

ON CHITIETDATHANG

AFTER INSERT, UPDATE

AS

BEGIN

-- Khi bổ sung bản ghi mới

IF EXISTS(SELECT \* FROM INSERTED) AND NOT EXISTS(SELECT \* FROM DELETED)

BEGIN

DECLARE @SoLuongBan INT, @MaHang INT, @SoLuongHienCo INT

SELECT @MaHang = MAHANG, @SoLuongBan = SOLUONG FROM INSERTED

SELECT @SoLuongHienCo = SOLUONG FROM MATHANG WHERE MAHANG = @MaHang

IF @SoLuongHienCo >= @SoLuongBan

BEGIN

UPDATE MATHANG SET SOLUONG = SOLUONG - @SoLuongBan WHERE MAHANG = @MaHang

END

ELSE

BEGIN

ROLLBACK TRANSACTION

RAISERROR('Số lượng hàng không đủ', 16, 1)

RETURN

END

END

-- Khi cập nhật lại số lượng hàng

IF EXISTS(SELECT \* FROM INSERTED) AND EXISTS(SELECT \* FROM DELETED)

BEGIN

DECLARE @SoLuongBanMoi INT, @SoLuongBanCu INT, @MaHang INT, @SoLuongHienCo INT

SELECT @MaHang = MAHANG, @SoLuongBanMoi = SOLUONG FROM INSERTED

SELECT @SoLuongBanCu = SOLUONG FROM DELETED

SELECT @SoLuongHienCo = SOLUONG FROM MATHANG WHERE MAHANG = @MaHang

IF @SoLuongBanMoi >= 1 AND (@SoLuongHienCo + @SoLuongBanCu - @SoLuongBanMoi) >= 0

BEGIN

UPDATE MATHANG SET SOLUONG = @SoLuongHienCo + @SoLuongBanCu - @SoLuongBanMoi WHERE MAHANG = @MaHang

END

ELSE

BEGIN

ROLLBACK TRANSACTION

RAISERROR('Số lượng hàng không hợp lệ', 16, 1)

RETURN

END

END

END

1. **Viết trigger cho bảng CHITIETDATHANG để sao cho chỉ chấp nhận giá hàng bán ra phải nhỏ hơn hoặc bằng giá gốc (giá của mặt hàng trong bảng MATHANG).**

CREATE TRIGGER trg\_CheckPrice

AFTER INSERT, UPDATE ON CHITIETDATHANG

FOR EACH ROW

BEGIN

DECLARE @GiaBanRa DECIMAL(10, 2);

DECLARE @GiaGoc DECIMAL(10, 2);

-- Lấy giá bán ra từ bản ghi mới trong CHITIETDATHANG

SELECT @GiaBanRa = i.Gia FROM INSERTED i WHERE i.MaMatHang = NEW.MaMatHang;

-- Lấy giá gốc từ bảng MATHANG

SELECT @GiaGoc = m.Gia FROM MATHANG m WHERE m.MaMatHang = NEW.MaMatHang;

-- Kiểm tra giá bán ra so với giá gốc

IF @GiaBanRa > @GiaGoc

BEGIN

RAISERROR('Giá bán ra không được lớn hơn giá gốc', 16, 1);

ROLLBACK TRANSACTION;

END

END;