**1.**      **Các thành phần chức năng.**

**1.1.**           **V\_LIST\_NewAccount.**

-         Cho xem danh sách 10 tài khoản được tạo gần đây nhất gồm các thông tin : ID, Username, Email.

GO

CREATE VIEW V\_LIST\_NewAccount

AS( SELECT TOP 10\*

FROM dbo.TaiKhoan

ORDER BY idTK DESC )

-         Giải thuật.

+Sắp xếp bảng tài khoản theo thứ tự id giảm dần

+Lấy 10 phần từ đầu tiên.

**1.2.**           **V\_LIST\_Account.**

-         Cho xem danh sách của tất cả tài khoản có trong cơ sở dữ liệu, gồm các thông tin : ID\_Tài Khoản, Tên Đăng Nhập, Chức vụ,Họ Tên, Địa Chỉ, Năm Sinh, Ngày tạo, Block

GO

CREATE VIEW V\_LIST\_Account

AS( SELECT \*

FROM dbo.TaiKhoan )

-         Giải thuật:

+Lấy tất cả dữ liệu trong bảng TaiKhoan

**1.3.**           **V\_LIST\_Laptop**

-         Xem danh sách các máy có trong cơ sở dữ liệu:

+Mã Hàng Hóa, Tên Hàng Hóa, Giá Gốc, Giá Bán, idNhaSX, Số Lượng và các mô tả về Laptop

-         Giải thuật:

GO

CREATE VIEW V\_LIST\_Laptop

AS( SELECT \*

FROM dbo.HangHoa )

1.5   V\_LIST\_GiỏHangTấtCả.

-         Cho xem danh sách tất cả những giỏ hàng được đặt, gồm các thông tin : ID, Tên SP, Số Lượng, Tổng Tiền, Ngày đặt, Ngày giao Và Các thông tin Sản phẩm trong giỏ hàng

GO

CREATE PROC V\_LIST\_Bill

AS( SELECT \*

FROM dbo.Bill

FULL OUTER JOIN dbo.Bill\_info

ON Bill\_info.idBill = Bill.idBill )

-         Giải thuật.

**1.5.**           **V\_R\_Authorization.**

-         Đọc danh sách các quyền.

GO

CREATE VIEW V\_R\_PhanQuyen

AS( SELECT\*

FROM dbo.PhanQuyen )

-         Giải thuật.

**1.9.**           **V\_Report\_DonHang.**

-         Lấy tất cả đơn hàng để xuất phiếu report.

GO

CREATE PROC V\_LIST\_Bill

AS( SELECT \*

FROM dbo.Bill

FULL OUTER JOIN dbo.Bill\_info

ON Bill\_info.idBill = Bill.idBill )

-         Giải thuật.

**1.10.**       **V\_List\_PhươngThứcThanhToan**

-Lấy tất cả phương thức thanh tóán

CREATE VIEW V\_LIST\_PhuongThucThanhToan

AS( SELECT\*

FROM dbo.PT\_ThanhToan )

-Giải thuật:

**1.11.**       **V\_List\_NhaSanXuat**

-Lấy tất cả thông tin nhà sản xuất

GO

CREATE VIEW V\_ListNSX

AS SELECT \*

FROM dbo.NSX

-Giải thuật:

**1.12.**       **Pro\_List\_ThongTinNhậpKho(Không có đối tượng địa chỉ -.-)**

*(câu này coi lại dùm t thấy sai sai)*

-         Lấy tất cả thời gian, số lượng, địa chỉ  của đơn nhập.

GO

CREATE PROCEDURE pro\_List\_ThongTinNhapKho

AS( SELECT NgayGiaoDich,SoLuong,TongTien

FROM dbo.DonNhap,dbo.Nhap

WHERE DonNhap.idGiaoDich=Nhap.idGiaoDich )

-         Giải Thuật

**1.13.**      

**1.14.**       **Pro\_KhuyenMaiCuaThang**

-         Lấy tất cả thông tin khuyên mãi trong tháng Với dữ liệu ban đầu là Date(Tháng- Năm)

*(trong tháng là lấy như này đúng ko?)*

GO

CREATE PROCEDURE pro\_KhuyenMaiCuaThang(@thang INT, @nam INT)

AS( SELECT \*

FROM dbo.KhuyenMai

WHERE MONTH(NgayBatDau)=@thang

AND YEAR(NgayBatDau)=@nam

AND MONTH(NgayKetThuc)=@thang

AND YEAR(NgayKetThuc)=@nam )

-         Giải thuật:

**1.16.**    **Pro\_DanhSachCacTaiKhoanBiKhoa**

-         Lấy tất cả các thông tin tài khoản bị khóa để Report

-         Giải thuật

GO

SELECT \*

FROM dbo.TaiKhoan

WHERE Chan='1'

**1.18.**       **TR\_D\_Account. (Khánh)**

-         Tạo 1 cái bẫy khi xóa dữ liệu trên bảng Account.

Create trigger TR\_D\_Account

On dbo.TaiKhoan

For Delete

As

Begin

delete NhanVien

from deleted inner join NhanVien

on NhanVien.idNV = deleted.idTK

delete PhanQuyenTK

from deleted inner join PhanQuyenTK

on PhanQuyen.idTK = deleted.idTK

End

Go

delete TaiKhoan where idTK = "123"

Go

Hi

**1.19.**       **TR\_D\_Authorization. (Khánh)**

-         Tạo 1 cái bẫy khi xóa dữ liệu trên bảng Quyền try cập.

Create trigger TR\_D\_Authorization

On dbo.PhanQuyenTK

For Delete

As

Begin

delete PhanQuyen

from deleted inner join PhanQuyen

on PhanQuyen.idPQ = deleted.idPQ

delete TaiKhoan

from deleted inner join TaiKhoan

on TaiKhoan.idTK = deleted.idTK

End

Go

delete PhanQuyenTK where idPQ = "456"

delete PhanQuyenTK where idTK = "345"

Go

**1.20.**       **TR\_D\_PhuongThucThanhToan. (Khánh)**

-         Tạo 1 cái bẫy khi xóa dữ liệu trên bảng PhuongThucThanhToan.

-         Giải thuật.

Create trigger TR\_D\_PhuongThucThanhToan

On dbo.PT\_ThanhToan

For Delete

As

Begin

delete Bill

from deleted inner join Bill

on Bill.idPT\_ThanhToan = deleted.idPT\_ThanhToan

End

Go

delete PT\_ThanhToan where idPT\_ThanhToan = "01"

Go

**1.21.**       **TR\_D\_SanPham. (Khánh)**

-         Tạo 1 cái bẫy khi xóa dữ liệu trên bảng SanPham.

-         Giải thuật.

Create trigger TR\_D\_SanPham

On dbo.Hanghoa

For Delete

As

Begin

delete HangHoaImage

from deleted inner join HangHoaImage

on HangHoaImage.idSP = deleted.idSP

End

Go

delete HangHoa where idP = "001"

Go

**1.22.**       **TR\_D\_NhaSanXuất. (Khánh)**

-         Tạo 1 cái bẫy khi xóa dữ liệu trên bảng NhaSX.

-         Giải thuật.

Create trigger TR\_D\_NhaSanXuat

On dbo.NSX

For Delete

As

Begin

delete Hanghoa

from deleted inner join HangHoa

on HangHoa.idSP = deleted.idNSX

End

**1.23.**       **TR\_D\_Bill\_InFo. (Khánh)**

-         Tạo 1 cái bẫy khi xóa dữ liệu trên bảng GioHangInfo(Khi chúng ta xóa dữ liệu số lượng trong bảng)

-         Giải thuật.

**1.24.**       **TR\_D\_GioHang. (Khánh)**

-         Tạo 1 cái bẫy khi xóa dữ liệu trên bảng GioHang.

-         Giải thuật.

**1.25.**       **GetAccountByID.**

-         Hàm lấy dữ liệu 1 Account thông qua ID.

GO

CREATE FUNCTION GetAccountByID (@Id int)

RETURNS TABLE

RETURN( SELECT \*

FROM dbo.TaiKhoan

WHERE idTK=@Id )

-         Giải thuật.

**1.26.**       **GetAccountByUserName.**

-         Hàm lấy dữ liệu 1 Account thông qua UserName.

GO

CREATE FUNCTION GetAccountByUsername (@Username INT)

RETURNS TABLE

RETURN( SELECT \*

FROM dbo.TaiKhoan

WHERE users=@Username )

-         Giải thuật.

**1.28.**       **GetImageByLapID.**

-         Hàm lấy hình ảnh  thông qua LapID.

GO

CREATE FUNCTION GetLapImageByID (@Id INT)

RETURNS IMAGE

BEGIN

RETURN( SELECT dbo.HangHoaImage.Hinh

FROM dbo.HangHoaImage

WHERE idSP=@Id)

END

-         Giải thuật.

**1.29.**       **GetEmployeeByID.**

-         Hàm lấy dữ liệu 1 Customer thông qua ID.

CREATE FUNCTION GetEmployeeByID (@Id INT)

RETURNS TABLE

RETURN( SELECT\*

FROM dbo.NhanVien

WHERE idNV=@Id )

-         Giải thuật.

**1.30.**       **GetEmployeeByName.**

-         Hàm lấy dữ liệu 1 Customer thông qua UserName của Account tương ứng.

GO

CREATE FUNCTION GetEmployeeByName (@Name INT)

RETURNS TABLE

RETURN( SELECT idNV,HoTen,NgaySinh,GioiTinh,DiaChi,SDT,ChucVu

FROM dbo.NhanVien,dbo.TaiKhoan

WHERE users=@Name

AND idNV=idTK )

-         Giải thuật.

**1.31.**       **GetDetailsAccount.**

-         Hàm lấy chi tiết của 1 người dùng.

-         Giải thuật.

**1.32.**       **GetLapByID.**

-         Hàm lấy 1 SảnPhẩm thông qua ID.

GO

CREATE FUNCTION GetLapByID (@Id INT)

RETURNS TABLE

RETURN( SELECT \*

FROM dbo.HangHoa

WHERE idSP=@Id )

-         Giải thuật.

**1.33.**       **SearchAccount.**

**1.34.**       **SearchLaptop**

**Hàm tìm kiếm sản phẩm. Trả về 1 danh sách các sản phẩm thỏa mãn yêu cầu.(m muốn thõa cái gì mới dc ><) Đoán xem -.- phải biết input là gì chứ ><**

-         Giải thuật.

**1.35.**       **CountAccount.**

-         Hàm đếm số lượng tài khoản khả dụng có trong cơ sở dữ liệu.

GO

CREATE FUNCTION CountAccount()

RETURNS INT

BEGIN

DECLARE @count INT

SELECT @count=COUNT(users)

FROM dbo.TaiKhoan

WHERE Chan='0'

RETURN @count

END

-         Giải thuật.

**1.36.**       **CountProduct.**

-         Hàm đếm số loại sản phẩm hiện được bày bán có trong cơ sở dữ liệu.

GO

CREATE FUNCTION CountProduct()

RETURNS INT

BEGIN

DECLARE @count INT

SELECT @count=COUNT(idSP)

FROM dbo.HangHoa

RETURN @count

END

-         Giải thuật.

**1.37.**       **GetQuyềnvsIDTaiKhoan.**

-         Hàm lấy quyền với tài khoản tương ứng.

GO

CREATE PROCEDURE GetAuthorization(@id INT)

AS( SELECT  PhanQuyen.idPQ,LoaiQuyen,TenPQ

FROM dbo.PhanQuyenTK,dbo.PhanQuyen

WHERE idTK=@id

AND PhanQuyen.idPQ=PhanQuyenTK.idPQ )

-         Giải thuật.

**1.38.**       **GetSảnPhẩmPrice.**

-         Hàm lấy giá sản phẩm.

GO

CREATE FUNCTION GetPrice1(@Id INT )

RETURNS INT

BEGIN

DECLARE @result INT

SELECT @result=dbo.HangHoa.GiaBan

FROM dbo.HangHoa

WHERE idSP=@Id

RETURN @result

END

GO

CREATE FUNCTION GetPrice2(@Id INT )

RETURNS INT

BEGIN

DECLARE @result INT

SELECT @result=dbo.HangHoa.GiaGoc

FROM dbo.HangHoa

WHERE idSP=@Id

RETURN @result

END

-         Giải thuật.

**1.39.**       **IsAccountExists.**

-         Hàm kiểm tra tên tài khoản đã được tạo hay chưa.

GO

CREATE FUNCTION IsAccountExists (@username NVARCHAR(20))

RETURNS BIT

BEGIN

DECLARE @tmp INT, @result BIT

SELECT @tmp=COUNT(user)

FROM dbo.TaiKhoan

WHERE users=@username

IF @tmp>0

SET @result=1

ELSE

SET @result=0

RETURN @result

END

-         Giải thuật.

**1.40.**       **IsLoginSuccess.**

-         Hàm kiểm tra tài khoản và mật khẩu người dùng dùng để đăng nhập có chính xác hay không.

GO

CREATE FUNCTION IsLoginSuccess(@username NVARCHAR(20),@pass NVARCHAR(20))

RETURNS BIT

BEGIN

DECLARE @result BIT

IF EXISTS ( SELECT \*

FROM dbo.TaiKhoan

WHERE users=@username

AND pass=@pass )

SET @result = 1

ELSE

SET @result = 0

RETURN @result

END

-         Giải thuật.

**1.41.**       **RevenueOfMonth.**

-         Hàm thống kê doanh thu hàng tháng.

-         Giải thuật.

**1.42.**       **RevenueOfYear.**

-         Hàm thống kê doanh thu hàng năm.

-         Giải thuật.

**1.43.**       **C\_Account.**

-         Thủ tục thêm mới một Account.

GO

CREATE PROCEDURE Add\_Account( @idTK INT,

@users NVARCHAR(20),

@pass NVARCHAR(20),

@Chan NVARCHAR(2) )

AS

INSERT INTO dbo.TaiKhoan

        (idTK, users, pass, Chan )

VALUES  (@idTK, @users, @pass, @Chan)

-         Giải thuật.

**1.44.**       **C\_AccountAuthorization.**

-         Thủ tục thêm mới quyền sử dụng cho 1 tài khoản.

GO

CREATE PROCEDURE Add\_AccountAuthorization( @idTK INT,

@idPQ INT )

AS

INSERT INTO dbo.TaiKhoan

VALUES  (@idTK, @idPQ)

-         Giải thuật.

**1.45.**       **C\_Customer.**

-         Thủ tục thêm mới một Customer.

GO

CREATE PROCEDURE Add\_Customer( @idNV INT,

@HoTen NVARCHAR(50),

@NgaySinh DATE,

@GioiTinh NVARCHAR(10),

@DiaChi NVARCHAR(200),

@SDT NVARCHAR(20),

@ChucVu NVARCHAR(50) )

AS

INSERT INTO dbo.NhanVien

VALUES  (@idNV, @HoTen, @NgaySinh, @GioiTinh, @DiaChi, @SDT, @ChucVu)

-         Giải thuật.

**1.46.**       **C\_Laptop.**

-         Thủ tục tạo mới một Laptop.

GO

CREATE PROCEDURE Add\_Laptop( @idSP INT,

@idNSX INT,

@TenSP NVARCHAR(50),

@CPU NVARCHAR(100),

@RAM NVARCHAR(50),

@ManHinh NVARCHAR(50),

@PIN NVARCHAR(20),

@GiaBan INT,

@GiaGoc INT,

@SoLuong INT

)

AS

INSERT INTO dbo.NhanVien

VALUES  (@idSP, @idNSX, @TenSP, @CPU, @RAM, @ManHinh, @PIN, @GiaBan, @GiaGoc, @SoLuong)

-         Giải thuật.

**1.47.**       **C\_PhotoCuaMotSanPham.**

-         Thủ tục tạo mới một Photo.

GO

CREATE PROCEDURE Add\_ProductImage( @idSP INT,

@Hinh IMAGE )

AS

INSERT INTO dbo.HangHoaImage

VALUES  (@idSP, @Hinh)

-         Giải thuật.

**1.48.**       **C\_Bill.**

-         Thủ tục tạo mới một PurchaseOrders.

GO

CREATE PROCEDURE Add\_Bill( @idBill INT,

@TongTien INT,

@NgayDatHang DATE,

@idKH INT,

@idPT\_ThanhToan INT,

@idKhuyenMai INT,

@DiaChiGiaoHang NVARCHAR(200),

@TinhTrangGiaoHang NVARCHAR(100) )

AS

INSERT INTO dbo.Bill

        ( TongTien ,

          NgayDatHang ,

          idKH ,

          idPT\_ThanhToan ,

          idKhuyenMai ,

          DiaChiGiaoHang ,

          TinhTrangGiaoHang

        )

VALUES  ( @idBill, @TongTien, @NgayDatHang, @idKH, @idPT\_ThanhToan, @idKhuyenMai, @DiaChiGiaoHang, @TinhTrangGiaoHang)

-         Giải thuật.

**1.49.**       **C\_ Bill\_Info.**

-         Thủ tục tạo mới một Bill\_Info.

GO

CREATE PROCEDURE Add\_BillInfo( @idBill INT,

@idSP INT,

@SoLuong INT)

AS

INSERT INTO dbo.Bill\_info

VALUES  (@idBill, @idSP, @SoLuong)

-         Giải thuật.

**AddNSX**

**1.50.**       **D\_Account.**

-         Thủ tục xóa bỏ 1 Account.

GO

CREATE PROCEDURE D\_Account( @Id INT)

AS

DELETE dbo.TaiKhoan WHERE idTK=@Id

-         Giải thuật.

**1.51.**       **D\_ AccountAuthorization.**

-        Xóa quyền trên bảng AccountAuthorization.

-         Giải thuật.

GO

CREATE PROCEDURE D\_AccountAuthorization( @Id INT)

AS

DELETE dbo.PhanQuyenTK WHERE idPQ=@Id

**1.53.**       **TRAN\_CreateAccount.**

-         Thủ tục tạo mới khi người dùng đăng kí tài khoản.

-         Giải thuật.

CREATE PROC InsertKhachHang

@idKH INT,

@HoTen CHAR(50),

@GioiTinh CHAR(10),

@DiaChi CHAR(200),

@SDT CHAR(15)

AS BEGIN

INSERT dbo.KhanhHang

(

    idKH,HoTen,GioiTinh,DiaChi,SDT

)

VALUES

(

@idKH,@HoTen,@GioiTinh,@DiaChi,@SDT

)

END

GO

**1.54.**       **TRAN\_UpdateAccount.**

-         Thủ tục cập nhật khi quản lý cập nhật lại 1 tài khoản.

-         Giải thuật.

**1.55.**       **TRAN\_UpdateInfoCustomer.**

-         Hàm cập nhật khi cập nhật thông tin cá nhân.

-         Giải thuật.

CREATE PROC UpdateKhachHang

@idKH INT,

@HoTen CHAR(50),

@GioiTinh CHAR(10),

@DiaChi CHAR(200),

@SDT CHAR(15)

AS

BEGIN

    UPDATE dbo.KhachHang SET

SDT = @SDT, HoTen = @HoTen, GioiTinh = @GioiTinh, DiaChi = @DiaChi

WHERE idKH = @idKH

END

**1.56.**       **U\_Account.**

-         Thủ tục cập nhật Account.

CREATE PROCEDURE U\_Account(@id INT,@newUsername NVARCHAR(20),@newPass NVARCHAR(20),@newChan NVARCHAR(2))

AS UPDATE dbo.TaiKhoan

SET users=@newUsername,

pass=@newPass,

Chan=@newChan

WHERE idTK=@id

-         Giải thuật.

**1.57.**       **U\_Phân quyền.**

-         Thủ tục cập nhật AccoutAuthorization.

GO

CREATE PROCEDURE U\_Authorization(@id INT,@loaiQuyen INT),@tenQuyen NVARCHAR(50))

AS UPDATE dbo.PhanQuyen

SET LoaiQuyen=@loaiQuyen,

TenPQ=@tenQuyen

WHERE idPQ=@id

-         Giải thuật.

**1.58.**       **U\_Password.**

-         Thủ tục cập nhật mật khẩu.

GO

CREATE PROCEDURE U\_Password( @username NVARCHAR(20),@newpass NVARCHAR(20))

AS UPDATE dbo.TaiKhoan

SET pass=@newpass

WHERE users=@username

-         Giải thuật.

**1.59.**       **U\_HangHoa.**

-         Thủ tục cập nhật Hàng Hóa.

GO

CREATE PROCEDURE U\_Product( @id INT, @idNSX INT, @tenSP NVARCHAR(50), @CPU NVARCHAR(100), @RAM NVARCHAR(50), @ManHinh NVARCHAR(50),

@PIN NVARCHAR(20), @GiaBan INT, @GiaGoc INT, @SoLuong INT)

AS UPDATE dbo.HangHoa

SET idNSX=@idNSX,

TenSP=@tenSP,

CPU=@CPU,

RAM=@RAM,

ManHinh=@ManHinh,

PIN=@PIN,

GiaBan=@GiaBan,

GiaGoc=@GiaGoc,

SoLuong=@SoLuong

WHERE idSP=@id

-         Giải thuật.

**1.61.**       **U\_Bill.**

-         Thủ tục cập nhật đơn hàng.

GO

CREATE PROCEDURE U\_Bill(@idBill INT,

@TongTien INT,

@NgayDatHang DATE,

@idKH INT,

@idPT\_ThanhToan INT,

@idKhuyenMai INT,

@DiaChiGiaoHang NVARCHAR(200),

@TinhTrangGiaoHang NVARCHAR(100))

AS UPDATE dbo.Bill

SET TongTien=@TongTien,

NgayDatHang=@NgayDatHang,

idKH=@idKH,

idPT\_ThanhToan=@idPT\_ThanhToan,

idKhuyenMai=@idKhuyenMai,

DiaChiGiaoHang=@DiaChiGiaoHang,

TinhTrangGiaoHang=@TinhTrangGiaoHang

WHERE idBill=@idBill

-         Giải thuật.

**1.62.**       **U\_Bill\_Info.**

-         Thủ tục cập nhật billInfo.

GO

CREATE PROCEDURE U\_Bill\_Info(@idBill INT,

@idSP INT,

            @SoLuong INT)

AS UPDATE dbo.Bill\_info

SET idSP=@idBill,

SoLuong=@SoLuong

WHERE idBill=idBill

-         Giải thuật.

**1.63.**       **SendCodeToResetPassword(Đoán xem ai làm)**

-         Thủ tục gửi mã xác nhận quên mật khẩu.

-         Giải thuật :

·        Tạo ngẫu nhiên code 6 số

·        Gửi code qua mail, đồng thời trả code về project để xử lí.

**1.65.**       **Procedure(Thành)**

-         Thủ tục thêm dữ liệu trả lời cho bot chat.

-         Giải thuật:

**1.66.**       **Inventory**

-         Hàm trả về các tổng sản phẩm trong kho sao lại là các tổng

GO

CREATE FUNCTION Inventory()

RETURNS INT

BEGIN

DECLARE @tong INT

SELECT @tong=SUM(SoLuong)

FROM dbo.HangHoa

RETURN @tong

END

-         Giải thuật

**1.68.**       Pro\_**List\_Drop\_Sale\_KhoangThoiGian**

-         Xem những mặt hàng có doanh số đi xuống trong 3 tháng gần nhất

-         Giải thuật

**1.69.**       Pro\_**Rise\_Sale\_KhoanThoiGian**

-         Xem những mặt hàng có doanh số tăng lên trong 3 tháng gần nhất.

-         Giải thuật

**1.73.**       **V\_List\_KhachHangMuaHangNhieuNhat**

-         Lấy danh sách khách hàng mua hang nhiều nhất

GO

CREATE VIEW PurchaseTimes

AS( SELECT idKH,COUNT(idBill) AS SoLanMua

FROM dbo.Bill

GROUP BY idKH

ORDER BY SoLanMua DESC )

GO

CREATE VIEW TopKH

AS( SELECT \*

FROM dbo.KhachHang

WHERE idKH IN( SELECT TOP 5(idKH)

FROM dbo.TopPurchaseTimes ))

-         Giải thuật

**1.76.**   **Func\_DoiHoaThanhThuong(Thành)**

-         Đổi chuỗi thành kí tự thường để so sánh

SELECT LOWER('QUANTRIMANG.COM');

Result: ‘quantrimang.com’

cái này cx cần làm????

-         Giải Thuật

**1.77.**   **MaHoaKiTu(Thành)**

-         Đổi chuỗi thường thành các mã hóa

-         Giải Thuật

**1.78.**   **Report\_SảnPhẩmĐược Mua NhiềuLần**

- Tạo report gồm các sản phẩm được mua lại nhiều lần.

- Giải thuật:

**1.79.**       **Check\_HangHoa(GiaBan): (Khánh)**

-         Dữ liệu GiaBanluôn lớn hơn 0.

-         Giải thuật: Từ làm bên phân tạo bảng rồi :3

**1.79.**       **Check\_HangHoa(GiaGoc): (Khánh)**

-         Dữ liệu GiaGoc luôn lớn hơn 0.

-         Giải thuật:

**1.80.**       **Check\_NhapKho(SoLuong):(Khánh)**

-         Dữ liệu SoLuong luôn phải lớn hơn 0.

-         Giải thuật: Làm rồi thi ghi câu lệnh ra -.- Đừng có ghi là Từ đã làm .-.

**1.81.**       **Check\_NhapKho(NgayGiaoDich):(Khánh)**

-         Dữ liệu NgayGiaoDich luôn nhỏ hơn hoặc bằng ngày hiện tại

-         Giải thuật:

**1.82.**       **Check\_Bil(TongTien):(Khánh)**

-         Dữ liệu TôngTien luôn lớn hơn 0.

-         Giải thuật: Từ làm bên phân tạo bảng rồi :3

1.84. Trigger\_

-Khi nhập số lượng ở bảng Nhâpk thì số lượng ở Hàng Hóa tăng lên tuongw ứng

1.85 Trigger

-Tương tự như 1,84 Thì Tổng Tiền ở Đơn Nhập sẽ tự tăng lên khi nhập hàng hóa vào

Alias: Sử dụng nó để xét các trường hợp đặc biệt, Yêu cầu phải sử dụng ít nhất một lần trong đời.

Anh em cứ làm xong hết cái này đi

Sau ngày 24/11/2020 Quay lại mới có câu mới

Từ Với Khánh Ngày 30/11/2020 Nộp :( Không Thành rút môn)

Với TRAN-BILL

Ta có Hàm đổi chữ hoa thành thường

CREATE FUNCTION [dbo].[fuConvertToUnsign1] ( @strInput NVARCHAR(4000) )

RETURNS NVARCHAR(4000)

AS

BEGIN

IF @strInput IS NULL

RETURN @strInput

IF @strInput = ''

RETURN @strInput

DECLARE @RT NVARCHAR(4000)

DECLARE @SIGN\_CHARS NCHAR(136)

DECLARE @UNSIGN\_CHARS NCHAR (136)

SET @SIGN\_CHARS = N'ăâđêôơưàảãạáằẳẵặắầẩẫậấèẻẽẹéềểễệế ìỉĩịíòỏõọóồổỗộốờởỡợớùủũụúừửữựứỳỷỹỵý ĂÂĐÊÔƠƯÀẢÃẠÁẰẲẴẶẮẦẨẪẬẤÈẺẼẸÉỀỂỄỆẾÌỈĨỊÍ ÒỎÕỌÓỒỔỖỘỐỜỞỠỢỚÙỦŨỤÚỪỬỮỰỨỲỶỸỴÝ' +NCHAR(272)+ NCHAR(208)

SET @UNSIGN\_CHARS = N'aadeoouaaaaaaaaaaaaaaaeeeeeeeeee iiiiiooooooooooooooouuuuuuuuuuyyyyy AADEOOUAAAAAAAAAAAAAAAEEEEEEEEEEIIIII OOOOOOOOOOOOOOOUUUUUUUUUUYYYYYDD'

DECLARE @COUNTER int

DECLARE @COUNTER1 int

SET @COUNTER = 1

WHILE (@COUNTER <=LEN(@strInput))

BEGIN

SET @COUNTER1 = 1

WHILE (@COUNTER1 <=LEN(@SIGN\_CHARS)+1)

BEGIN

IF UNICODE(SUBSTRING(@SIGN\_CHARS, @COUNTER1,1)) = UNICODE(SUBSTRING(@strInput,@COUNTER ,1) )

BEGIN

IF @COUNTER=1

SET @strInput = SUBSTRING(@UNSIGN\_CHARS, @COUNTER1,1) + SUBSTRING(@strInput, @COUNTER+1,LEN(@strInput)-1)

ELSE

SET @strInput = SUBSTRING(@strInput, 1, @COUNTER-1) +SUBSTRING(@UNSIGN\_CHARS, @COUNTER1,1) + SUBSTRING(@strInput, @COUNTER+1,LEN(@strInput)- @COUNTER)

BREAK

END

SET @COUNTER1 = @COUNTER1 +1

END

SET @COUNTER = @COUNTER +1

END

SET @strInput = replace(@strInput,' ','-')

RETURN @strInput

END

Ta có: Hàm cập tài khoản:

GO

CREATE PROC [dbo].[USP\_UpdateAccount]

@userName NVARCHAR(100), @displayName NVARCHAR(100), @password NVARCHAR(100), @newPassword NVARCHAR(100)

AS

BEGIN

DECLARE @isRightPass INT = 0

SELECT @isRightPass = COUNT(\*) FROM dbo.Account WHERE USERName = @userName AND PassWord = @password

IF (@isRightPass = 1)

BEGIN

IF (@newPassword = NULL OR @newPassword = '')

BEGIN

UPDATE dbo.Account SET DisplayName = @displayName WHERE UserName = @userName

END

ELSE

UPDATE dbo.Account SET DisplayName = @displayName, PassWord = @newPassword WHERE UserName = @userName

end

END

Ta có: DECLARE Dùng trong các hàm tạo biến tạm để xét: Cụ thể là sử dụng trong bảng Bill

Dùng Hàm tương tự này trong Bill để khi ta thêm sản phẩm xét xem sản phẩm đó có tồn tại trong bill hay chưa, nếu chưa có chúng ta có thể số lượng trực tiếp hoặc chúng ta có thể cộng dồn.

Cái dưới đây chỉ là Mẫu tham khảo.

CREATE PROC [dbo].[USP\_InsertBillInfo]

@idBill INT, @idFood INT, @count INT

AS

BEGIN

DECLARE @isExitsBillInfo INT

DECLARE @foodCount INT = 1

SELECT @isExitsBillInfo = id, @foodCount = b.count

FROM dbo.BillInfo AS b

WHERE idBill = @idBill AND idFood = @idFood

IF (@isExitsBillInfo > 0)

BEGIN

DECLARE @newCount INT = @foodCount + @count

IF (@newCount > 0)

UPDATE dbo.BillInfo SET count = @foodCount + @count WHERE idFood = @idFood

ELSE

DELETE dbo.BillInfo WHERE idBill = @idBill AND idFood = @idFood

END

ELSE

BEGIN

INSERT dbo.BillInfo

( idBill, idFood, count )

VALUES ( @idBill, -- idBill - int

@idFood, -- idFood - int

@count -- count - int

)

END

END

GO

SQL **Injection Sử dụng hàm để chống các SQL Injection.**

Có hàm xử lý nếu có Mã trống trong bảng mã, Khi đó các sản phẩm

Xét Các Default Với các check thỏa mãn các điều kiện thông thường như số lượng hoặc giá tiền luôn thỏa mãn lớn hon hoặc bằng 0 hoặc các chuỗi phải thỏa mãn điều kiện như giới tính phải nam hay là nữ hoặc vân vân

Xét các **, Auto Increment luôn luôn phải có trong tất cả các mã.**

**Xét các Union: Bao gồm các Ten TK, ? Ae nghĩ kĩ rồi ghi thêm**

Xét với Các trigger

Với Check mỗi kết quả có sẽ được thực hiện hay không ta sử dụng Exist thay vì dùng Slect \* from \* để check khác null.

Yêu cầu: Order by tên và giá theo desc và asc Với cụm từ Like sử dụng hàm chuyển chữ hoa thành thường và sử dụng thêm Ký tự đại diện.

Ngoài ra ta còn có thể tìm theo khoảng giá để sử dụng Between

Trang Product: Tìm kiếm sản phẩm: Theo 2 tiêu chí:

* Sắp xếp theo tên, Được quyền chọn lựa tang dần hoặc giảm dần
* Sắp xếp theo giá, Tương tự như vậy
* Có đổi chuỗi, Có order by Theo desc Hoạc ASC

Trang NSX: Nhà Sản Xuất Top đầu Với yêu cầu sử dụng Top vs Group by Số SP để tìm ra Top các nhà sản xuất có nhiều hàng hóa nhất bằng các sử dụng Sum để sắp xếp đưa lên đầu