--Product

CREATE PROCEDURE getProductByCatalog

@catId CHAR(5),

@numProduct INT = NULL,

@isAdmin BIT = 0

AS

BEGIN

IF(@isAdmin IS NULL OR @isAdmin = 0)

IF (@numProduct IS NULL)

SELECT \* FROM Product WHERE Status = 1 AND CatalogId = @catId

ELSE

SELECT TOP(@numProduct) \* FROM Product WHERE Status = 1 AND CatalogId = @catId

ELSE

IF (@numProduct IS NULL)

SELECT \* FROM Product WHERE CatalogId = @catId

ELSE

SELECT TOP(@numProduct) \* FROM Product WHERE CatalogId = @catId

END

GO

CREATE PROCEDURE getLastestProduct

@numProduct INT = NULL

AS

BEGIN

IF (@numProduct IS NULL)

SELECT \* FROM Product WHERE Status = 1 ORDER BY Created DESC

ELSE

SELECT TOP(@numProduct) \* FROM Product WHERE Status = 1 ORDER BY Created DESC

END

GO

CREATE PROCEDURE getSaleProduct

@numProduct INT = NULL

AS

BEGIN

IF (@numProduct IS NULL)

SELECT \* FROM Product WHERE Status = 1 AND Discount > 0 ORDER BY Discount DESC

ELSE

SELECT TOP(@numProduct) \* FROM Product WHERE Status = 1 AND Discount > 0 ORDER BY Discount DESC

END

GO

CREATE PROCEDURE getHotProduct

@numProduct INT = NULL

AS

BEGIN

IF (@numProduct IS NULL)

SELECT \* FROM Product WHERE Status = 1 AND [View] > 0 ORDER BY [View] DESC

ELSE

SELECT TOP(@numProduct) \* FROM Product WHERE Status = 1 AND [View] > 0 ORDER BY [View] DESC

END

GO

CREATE PROCEDURE getHotSellingProduct

@numProduct INT = NULL

AS

BEGIN

IF (@numProduct IS NULL)

SELECT \* FROM Product WHERE Status = 1 AND Sold > 0 ORDER BY Sold DESC

ELSE

SELECT TOP(@numProduct) \* FROM Product WHERE Status = 1 AND Sold > 0 ORDER BY Sold DESC

END

GO

CREATE PROCEDURE getProductById

@proId CHAR(7)

AS

BEGIN

SELECT \* FROM Product WHERE Status = 1 AND ProductId = @proId

END

GO

CREATE PROCEDURE getProductBySpecification

@specKey NVARCHAR(20),

@specValue NVARCHAR(20),

@isAdmin BIT = 0

AS

BEGIN

IF(@isAdmin IS NULL OR @isAdmin = 0)

SELECT p.\* FROM Specification s RIGHT JOIN Product p ON s.ProductId = p.ProductId WHERE s.Status = 1 AND s.SpecificationKey = @specKey AND s.Specification = @specValue

else

SELECT p.\* FROM Specification s RIGHT JOIN Product p ON s.ProductId = p.ProductId WHERE s.SpecificationKey = @specKey AND s.Specification = @specValue

END

GO

CREATE PROCEDURE getProductBetweenPrice

@fromPrice FLOAT,

@toPrice FLOAT

AS

BEGIN

SELECT \* FROM Product WHERE Status = 1 AND PriceOutput BETWEEN @fromPrice AND @toPrice

END

GO

CREATE PROCEDURE getNumberProduct

AS

BEGIN

SELECT COUNT(\*) FROM Product

END

GO

CREATE PROCEDURE getNumberProductInCatalog

@catId CHAR(5)

AS

BEGIN

SELECT COUNT(\*) FROM Product WHERE CatalogId = @catId

END

GO

CREATE PROCEDURE getProductByName

@name NVARCHAR(100)

AS

BEGIN

SELECT \* FROM Product WHERE ProductName = @name

END

GO

CREATE PROCEDURE getAllProduct

@isAdmin BIT = 0

AS

BEGIN

IF(@isAdmin = 0)

SELECT \* FROM Product WHERE Status = 1

ELSE

SELECT \* FROM Product

END

GO

CREATE PROCEDURE createProduct

@productId CHAR(7),

@proName NVARCHAR(100),

@priceOutput FLOAT,

@discount INT,

@quantity INT,

@quality INT,

@title NVARCHAR(200),

@guarantee INT,

@oneSwitchOne INT,

@description NTEXT,

@status BIT,

@catalogId CHAR(5),

@result BIT OUTPUT

AS

BEGIN

BEGIN try

INSERT Product VALUES(@productId,@proName,@priceOutput,@discount,@quantity,@quality,@title,@guarantee,@oneSwitchOne,@description,NULL,NULL,NULL,@status,@catalogId)

SET @result = 1

END try

BEGIN catch

SET @result = 0

END catch

END

GO

CREATE PROCEDURE updateProduct

@productId CHAR(7),

@proName NVARCHAR(100),

@priceOutput FLOAT,

@discount INT,

@quantity INT,

@quality INT,

@title NVARCHAR(200),

@guarantee INT,

@oneSwitchOne INT,

@description NTEXT,

@status BIT,

@catalogId CHAR(5),

@result BIT OUTPUT

AS

BEGIN

BEGIN try

UPDATE Product

SET ProductName = @proName,

PriceOutput = @priceOutput,

Discount = @discount,

Quantity = @quantity,

Quality = @quality,

Title = @title,

Guarantee = @guarantee,

OneSwitchOne = @oneSwitchOne,

Description = @description,

Status = @status,

CatalogId = @catalogId

WHERE ProductId = @productId

SET @result = 1

END try

BEGIN catch

SET @result = 0

END catch

END

GO

ALTER PROCEDURE deleteProduct

@productId CHAR(7),

@result BIT OUTPUT

AS

BEGIN

BEGIN try

IF(NOT EXISTS(SELECT 1 FROM Product WHERE ProductId = @productId))

SET @result = 0

ELSE

BEGIN

DELETE Product WHERE ProductId = @productId;

SET @result = 1

END

end try

BEGIN catch

SET @result = 0

END catch

END

GO

ALTER PROCEDURE toggleProduct

@productId CHAR(7),

@result BIT OUTPUT

AS

BEGIN

DECLARE @status BIT

SET @status = (SELECT Status FROM Product WHERE ProductId = @productId)

IF (@status = 1)

BEGIN

UPDATE Product SET Status = 0 WHERE ProductId = @productId

SET @result = 0

END

else

BEGIN

UPDATE Product SET Status = 1 WHERE ProductId = @productId

SET @result = 1

END

END

GO

--\*\*\*\*\*\*Specification

ALTER PROCEDURE insertSpecification

@specKey nvarchar(20),

@spec nvarchar(20),

@status BIT,

@productId CHAR(7),

@result BIT OUTPUT

AS

BEGIN

BEGIN TRY

IF(NOT EXISTS(SELECT \* FROM Specification WHERE SpecificationKey = @specKey AND ProductId = @productId))

BEGIN

INSERT Specification VALUES(@specKey,@spec,@status,@productId)

SET @result = 1

END

ELSE

BEGIN try

UPDATE Specification

SET Specification = @spec,

Status = @status,

WHERE SpecificationKey = @specKey AND ProductId = @productId

SET @result = 1

END try

BEGIN catch

SET @result = 0

end catch

END TRY

BEGIN CATCH

SET @result = 0

END CATCH

END

GO

-- ALTER PROCEDURE updateSpecification

-- @specId INT,

-- @specKey nvarchar(20),

-- @spec nvarchar(20),

-- @status BIT,

-- @productId CHAR(7),

-- @result BIT OUTPUT

-- AS

-- BEGIN

-- BEGIN TRY

-- UPDATE Specification

-- SET SpecificationKey = @specKey,

-- Specification = @spec,

-- Status = @status,

-- ProductId = @productId

-- WHERE SpecificationId = @specId

-- SET @result = 1

-- END TRY

-- BEGIN CATCH

-- SET @result = 0

-- END CATCH

-- END

-- GO

ALTER PROCEDURE updateSpecification

@specKey nvarchar(20),

@spec nvarchar(20),

@status BIT,

@productId CHAR(7),

@result BIT OUTPUT

AS

BEGIN

BEGIN TRY

UPDATE Specification

SET Specification = @spec,

Status = @status

WHERE SpecificationKey = @specKey AND ProductId = @productId

SET @result = 1

END TRY

BEGIN CATCH

SET @result = 0

END CATCH

END

GO

CREATE PROCEDURE deleteSpecificationByProductId

@productId CHAR(7)

AS

BEGIN

DELETE Specification

WHERE ProductId = @productId

END

GO

CREATE PROCEDURE selectSpecificationByKey

@specKey nvarchar(20),

@isAdmin BIT = 0

AS

BEGIN

IF(@isAdmin = 0)

SELECT \* FROM Specification WHERE Status = 1 AND SpecificationKey = @specKey

ELSE

SELECT \* FROM Specification WHERE SpecificationKey = @specKey

END

GO

CREATE PROCEDURE selectSpecificationByProductId

@productId char(7),

@isAdmin BIT = 0

AS

BEGIN

IF(@isAdmin = 0)

SELECT \* FROM Specification WHERE Status = 1 AND ProductId = @productId

ELSE

SELECT \* FROM Specification WHERE ProductId = @productId

END

GO

CREATE PROCEDURE checkSpecificationExist

@specKey nvarchar(20),

@productId char(7),

@result BIT OUTPUT

AS

BEGIN

IF(NOT EXISTS(SELECT \* FROM Specification WHERE SpecificationKey = @specKey AND ProductId = @productId))

SET @result = 0

else

SET @result = 1

END

GO