2.a

ALTER PROCEDURE usp\_AddArticle

@artnamn varchar(30),

@antal smallint,

@pris decimal(6,2),

@rabatt decimal(6,2),

@plats char(10)

AS

BEGIN

BEGIN TRY

INSERT INTO artikel(artnamn, antal, pris, rabatt, plats)

VALUES(@artnamn, @antal, @pris, @rabatt, @plats)

END TRY

BEGIN CATCH

RAISERROR('Fel värde i artikel', 16, 1)

END CATCH

END

EXEC usp\_AddArticle 'Dator', 30, 6500.0, 0.0, 'Förråd 10';

ALTER PROCEDURE usp\_UppdateArticle

@artikelID int = 0,

@artnamn varchar(30),

@antal smallint,

@pris decimal(6,2),

@rabatt decimal(6,2),

@plats char(10)

AS

BEGIN

IF EXISTS(SELECT artikelID FROM artikel WHERE artikelID=@artikelID)

BEGIN TRY

UPDATE artikel

SET artnamn=@artnamn, antal=@antal, pris=@pris, rabatt=@rabatt, plats=@plats

WHERE artikelID = @artikelID

END TRY

BEGIN CATCH

RAISERROR('Något gick fel',16,1)

END CATCH

ELSE

RAISERROR('ArtikelID finns inte',16,1)

END

EXEC usp\_UppdateArticle 8, 'laptop', 30, 4500.0, 0.0, 'Förråd 10';

ALTER PROCEDURE usp\_DeleteArticle

@artikelID int = 0

AS

BEGIN

IF EXISTS(SELECT artikelID FROM artikel WHERE artikelID=@artikelID)

BEGIN

DELETE FROM artikel

WHERE artikelID=@artikelID

END

ELSE

BEGIN

RAISERROR('artikelid fanns inte',16,1)

END

END

EXEC usp\_DeleteArticle 9;

2.b

ALTER PROCEDURE usp\_ListArticle

@artikelID int = 0

AS

BEGIN

IF EXISTS(SELECT artikelID FROM artikel WHERE artikelID = @artikelID)

BEGIN

SELECT artikelID, artnamn, antal, pris, antal\*pris AS Lagervärde FROM artikel

WHERE artikelID = @artikelID

ORDER BY artnamn ASC

END

ELSE IF(@artikelID = 0)

BEGIN

SELECT artikelID, artnamn, antal, pris, antal\*pris AS Lagervärde FROM artikel

ORDER BY artnamn ASC

END

ELSE

BEGIN

RAISERROR('artikelid finns inte',16,1)

END

END

EXEC usp\_ListArticle 2

2.c

ALTER PROCEDURE usp\_PhoneCustomer

@kundID int = 0

AS

BEGIN

CREATE TABLE #Temp

(

namn varchar(30),

ort varchar(30),

telenr varchar(15),

teltypID int

)

IF EXISTS(SELECT kundID FROM kund WHERE kundID = @kundID)

BEGIN

INSERT INTO #Temp (namn, ort, telenr, teltypID)

SELECT namn, ort, telenr, teltypID FROM kund k

LEFT JOIN telefon t

ON k.kundID=t.kundID

WHERE k.kundID = @kundID

END

ELSE IF(@kundID = 0)

BEGIN

INSERT INTO #Temp (namn, ort, telenr, teltypID)

SELECT namn, ort, telenr, teltypID FROM kund k

LEFT JOIN telefon t

ON k.kundID=t.kundID

ORDER BY teltypID ASC

END

ELSE

BEGIN

RAISERROR('kundid finns inte',16,1)

END

SELECT \* FROM #Temp

DROP TABLE #Temp

END

EXEC usp\_PhoneCustomer 0

2.d

ALTER PROCEDURE usp\_SaleDate

@datumstart DATE,

@datumslut DATE

AS

BEGIN

IF(@datumstart > @datumslut)

BEGIN

RAISERROR('Startdatum kan inte vara störe än slutdatum',16,1)

END

ELSE

BEGIN

SELECT fr.ArtikelID, Artnamn, datum, fr.antal, fr.pris, totalt FROM artikel a

INNER JOIN fakturarad fr

ON a.artikelID=fr.artikelID

INNER JOIN faktura f

ON fr.fakturaID=f.fakturaID

WHERE datum BETWEEN @datumstart AND @datumslut

ORDER BY artnamn ASC

END

END

EXEC usp\_SaleDate '2015-01-01', '2015-02-16'

3.a

ALTER PROCEDURE usp\_MyFirstTransaktion

@fakturaID int, @artikelID int, @antal int, @pris decimal(6,2), @rabatt decimal(6,2), @momsID int

AS

BEGIN

IF EXISTS(SELECT artikelID FROM artikel WHERE artikelID=@artikelID)

BEGIN

BEGIN TRY

DECLARE @errormsg varchar(40)

BEGIN TRAN

SET @errormsg ='fakturarad kunde inte skapas'

INSERT INTO fakturarad(fakturaID, artikelID, antal, pris, rabatt, momsID)

VALUES(@fakturaID, @artikelID, @antal, @pris, @rabatt, @momsID)

SET @errormsg ='artikel kunde inte uppdateras'

UPDATE artikel

SET antal = antal-@antal

WHERE artikelID = @artikelID

COMMIT TRAN

END TRY

BEGIN CATCH

RAISERROR(@errormsg,16,1)

END CATCH

END

END

EXEC usp\_MyFirstTransaktion 5, 2, 301, 13.50, 0.0, 1

3.b

ALTER PROCEDURE usp\_NewPhone

@kundID int = @@IDENTITY,

@telenr varchar(15),

@teltypID int

AS

BEGIN

BEGIN TRY

INSERT INTO telefon(kundID, telenr, teltypID)

VALUES(@kundID, @telenr, @teltypID)

END TRY

BEGIN CATCH

RAISERROR('telefon kund inte läggas till',16,1)

END CATCH

END

ALTER PROCEDURE usp\_DoubleSproc

@namn varchar(40), @adress varchar(30), @postnr varchar(6), @ort varchar(25), @rabatt decimal(6, 2), @kategoriID tinyint

AS

BEGIN

BEGIN TRY

BEGIN TRAN

INSERT INTO kund (namn, adress, postnr, ort, rabatt, kategoriID)

VALUES(@namn, @adress, @postnr, @ort, @rabatt, @kategoriID)

EXEC usp\_NewPhone @@IDENTITY, '070-1234567', 5

COMMIT TRAN

END TRY

BEGIN CATCH

ROLLBACK TRAN

RAISERROR('Kunden kunde inte läggas till',16,1)

END CATCH

END

EXEC usp\_DoubleSproc 'statoil', 'storavägen 2', '352 50', 'Växjö', 0.00, 2

3.c

ALTER PROCEDURE usp\_DeleteInvoice

@fakturaID int

AS

BEGIN

DECLARE @artikelID int, @antal smallint, @errormsg varchar(50)

SELECT @artikelID = artikelID, @antal = antal

FROM fakturarad

WHERE radid = @fakturaID

BEGIN TRY

BEGIN TRAN

SET @errormsg = 'Kunde inte uppdatera antalet i artikel'

UPDATE artikel

SET antal = antal + @antal

WHERE artikelid = @artikelID

SET @errormsg = 'Kunde inte tabort fakturarad'

DELETE FROM fakturarad

WHERE fakturaID = @fakturaID

SET @errormsg = 'Kunde inte tabort faktura'

DELETE FROM faktura

WHERE fakturaID = @fakturaID

COMMIT TRAN

END TRY

BEGIN CATCH

ROLLBACK TRAN

RAISERROR('Något gick fel',16,1)

END CATCH

END

EXEC usp\_DeleteInvoice 5

4.a

CREATE TABLE Logging

(

ID int NOT NULL IDENTITY(1,1) PRIMARY KEY,

Tbl varchar(25),

TblID int,

Kol varchar(25),

Old varchar(100),

Datum datetime DEFAULT getdate(),

Usr varchar(10)

)

4.b

CREATE TRIGGER ut\_artikel ON artikel

AFTER UPDATE

AS

BEGIN

IF UPDATE(artnamn)

INSERT INTO Logging (tblID, tbl, kol, old, new, datum, usr)

SELECT

i.artikelID,

'artikel',

'namn',

d.artnamn,

i.artnamn,

getdate(),

system\_user

FROM inserted i INNER JOIN Deleted d

ON i.artikelID=d.artikelID;

END