**3.A**

// Lägger in ny kundinformation i tabellen kund

INSERT INTO Kund (Namn, Adress, Postnr, Ort, Rabatt, KategoriID)

VALUES ('Danielssons Elektriska AB', 'Storgatan', '123 56', 'Stockholm', 0.03, 1)

SELECT \* FROM Kategori

SELECT \* FROM Kund

// Lägger in telefon numer och typ

INSERT INTO Telefon (Telenr, TeltypID, KundID)

VALUES ('08-897 02 00', 4, 5), ('070-547 02 87', 5, 5)

SELECT \* FROM Teltyp

SELECT \* FROM Telefon

**3.B**

// Lägger in 3 nya artiklar mitt alla info i artikeltabellen

INSERT INTO Artikel (Artnamn, Antal, Pris, Rabatt)

VALUES ('Bildskärm, platt 10ms', 47, 2176.00, 0), ('Tangentbord', 36, 280.00, 0), ('Nätkabel, TP kat 5', 1020, 2.50, 0)

SELECT \* FROM Artikel

**3.C**

// Lägger in nya faktura i fakturatabellen

SELECT \* FROM Faktura

INSERT INTO Faktura (Datum, Betvillkor, KundID)

VALUES ('2012-04-20', 25, 5)

SELECT \* FROM Faktura

// Lägger in faktura information i fakturadtabellen

INSERT INTO Fakturad (FakturaID, ArtikelID, Antal, Pris, Rabatt, MomsID)

VALUES (5, 5, 2, 2176.00, 0.03, 4), (5, 1, 22, 6.70, 0.03, 4)

SELECT \* FROM Fakturad

**3.D**

// Lägger in ny Kategori i Kategoritabellen

INSERT INTO Kategori

(Kategori)

VALUES

('Standard');

SELECT \* FROM Kategori

**4.A**

// uppdaterar kund med ID 1 får nu kategori 4

SELECT \* FROM Kategori

SELECT \* FROM Kund

UPDATE Kund

SET KategoriID = 4

WHERE KundID = 1

**4.B**

// uppdaterar kund med TelID 4, den får ny telefonnummer

SELECT \* FROM Telefon

SELECT \* FROM Kund

UPDATE Telefon

SET Telenr = '0480-492239'

WHERE TelID = 4

**4.C**

// uppdaterar alla priser med 8%

SELECT \* FROM Artikel

UPDATE Artikel

SET Pris = Pris + (Pris \* 0.08)

SELECT \* FROM Artikel

**4.D**

//ny kolumn i tabellen i Artikel

SELECT \* FROM Artikel

ALTER TABLE Artikel

ADD Plats char(10)

**4.E**

//söker i kolumn Artnamn, hittat stringen Bildskärm och ändrar då på värdet i plats kolumnen

UPDATE Artikel

SET Plats = 'HPL 25'

WHERE Artnamn LIKE '%Bildskärm%

SELECT \* FROM Artikel

**4.F**

//söker i kolumn Plats, hittat alla null och ändrar då på värdet i kolumnen

SELECT \* FROM Artikel

UPDATE Artikel

SET Plats = 'Förråd 10'

WHERE Plats IS NULL

**4.G**

//ändra total priset i fakturan men dem nya priset , söker i kolumn Artikel, hittar alla ArtikelID 2 och sen ger det värdet till pris i fakturatabellen som har ArtikelID = 2

SELECT \* FROM Artikel

UPDATE Fakturad

SET Pris = antal \* (SELECT Pris FROM Artikel

WHERE ArtikelID = 2)

WHERE ArtikelID = 2

**5.A**

//tar bort rad som har KundID 5 och TeltypID4 i tabellen Telefon

DELETE FROM Telefon

WHERE KundID = 5 AND TeltypID = 4

SELECT \* FROM Telefon

**5.B**

//tar bort alla rader som har ArtikelID 2 i Fakturad tabellen

DELETE FROM Fakturad

WHERE ArtikelID = 2

SELECT \* FROM Fakturad

**5.C**

//tar bort rad som har KategoriID D i Kategori tabellen, men den misslyckas eftersom RI är satt till No Action, alltså får man inte ta bort en kategori som någon befintlig kund tillhör.

SELECT \* FROM Kategori

DELETE FROM Kategori

WHERE KategoriID = 4

**6.A**

//visar allt i kund tabellen

SELECT \* FROM Kund

**6.B**

//visar Namn, Postnr, Ort ur kund tabellen

SELECT Namn, Postnr, Ort FROM Kund

**6.C**

// visar Namn, Postnr, Ort ur kund tabellen, men i fallande ordning från största till minsta postnummer

SELECT Namn, Postnr, Ort FROM Kund

ORDER BY Postnr DESC

**6.D**

// visar Namn, Postnr och Ort läggs ihop till en ny rad och är bokstavsordning

SELECT Namn, Postnr + ' ' + Ort AS Postadress FROM Kund

ORDER BY Ort ASC

**6.E**

// visar artikelnr, namn, antal, pris och artikelvärde är pris \* antal utan decimaler

SELECT ArtikelID, Artnamn, Plats, Antal, FLOOR(Pris \* Antal) AS Artikelvärde FROM Artikel

**6.F**

// byter namn på kolumner och skapar en ny kolumn som presenteras

SELECT ArtikelID AS Artikelnr, Artnamn AS Namn, Plats, Antal, '\_\_\_\_\_\_\_\_\_\_\_\_' AS 'Nytt antal' FROM Artikel

**6.G**

// visar bar varor som har mer 22 eller mera och som finns i förråd 10

SELECT ArtikelID, Artnamn, Plats, Antal, FLOOR(Pris \* Antal) AS Artikelvärde FROM Artikel

WHERE Antal > 22 AND Plats LIKE 'Förråd 10'

**6.H**

// visar de artiklar som har de 5 högsta artikelvärdena

SELECT TOP 5 ArtikelID, Artnamn, Antal, Pris, FLOOR(Pris\*Antal) AS Artikelvärde

FROM Artikel

ORDER BY Artikelvärde DESC

**7.A**

// kombinerar två tabeller för att presentera namn, ort och teflonnummer, den presenterar namn i stigande ordning

SELECT Namn, Ort, Telenr

FROM Kund AS K INNER JOIN Telefon AS T

ON K.KundID = T.KundID

ORDER BY Namn

**7.B**

// kombinerar 3 tabeller för att presentera information och använda information för att räkna total summan med summa + moms - rabatt

SELECT Datum, Betvillkor, Artnamn, Fr.Antal, Fr.Pris, (M.Moms \* 100) AS 'Moms i %', Fr.Rabatt,

M.Moms\* ((Fr.Pris \* Fr.Antal) - ((Fr.Pris \* Fr.Antal) \* Fr.Rabatt)) + (Fr.Pris \* Fr.Antal) - ((Fr.Pris \* Fr.Antal) \* Fr.Rabatt) AS Summa

FROM Faktura AS F

INNER JOIN Fakturad AS Fr

ON F.FakturaID = Fr.FakturaID

INNER JOIN Artikel AS A

ON Fr.ArtikelID = A.ArtikelID

INNER JOIN Moms AS M

ON Fr.MomsID = M.MomsID

**7.C**

// kombinerar tabeller för att presentera information och använda information för att räkna total summan med summa + moms - rabatt, man får också för fals datum på fakturorna

SELECT Datum, Betvillkor, A.Artnamn, Fr.Antal, Fr.Pris, Moms\*100 AS Moms, Fr.Rabatt,

M.Moms\* ((Fr.Pris \* Fr.Antal) - ((Fr.Pris \* Fr.Antal) \* Fr.Rabatt)) + (Fr.Pris \* Fr.Antal) - ((Fr.Pris \* Fr.Antal) \* Fr.Rabatt) AS Summa, DATEADD(DAY, Betvillkor, Datum) AS Förfallodatum

FROM Faktura AS F

INNER JOIN Fakturad AS Fr

ON F.FakturaID = Fr.FakturaID

INNER JOIN Artikel AS A

ON A.ArtikelID = Fr.ArtikelID

INNER JOIN Moms AS M

ON Fr.MomsID = M.MomsID

**7.d**

// kombinerar tabeller för att presentera information

SELECT Namn, Kategori AS Kundkategori, Datum AS Fakturadatum, Betvillkor AS Betalningsvillkor

FROM Kund AS K

INNER JOIN Kategori AS Ka

ON K.KategoriID = Ka.KategoriID

INNER JOIN Faktura AS F

ON K.KundID = F.KundID

**7.E**

// kombinerar tabeller för att presentera information, läger in kontroll som kontrollerar att vi får de datum vi vill ha

SELECT Namn, Kategori AS Kundkategori, Datum AS Fakturadatum, Betvillkor AS Betalningsvillkor FROM Kund AS K

INNER JOIN Kategori AS Ka

ON K.KategoriID = Ka.KategoriID

INNER JOIN Faktura AS F

ON K.KundID = F.KundID

WHERE Datum BETWEEN '2015/02/01' AND '2015/02/27'

**7.F**

// kombinerar tabeller för att presentera information om alla kunder som inte köpt något

SELECT Namn, (Postnr + ' ' + Ort) AS Postadress

FROM KUND LEFT JOIN Faktura

ON Kund.KundID = Faktura.KundID

WHERE Faktura.KundID IS NULL

**7.G**

// kombinerar tabeller för att presentera information, om de artiklar som inte sålts

SELECT Artikel.ArtikelID, Artnamn AS Artikelnamn, Artikel.Pris, Artikel.Antal

FROM Artikel LEFT JOIN Fakturad

ON Fakturad.ArtikelID = Artikel.ArtikelID

WHERE Fakturad.ArtikelID IS NULL

**8.A**

// Skapar lista som visar antal kunder

SELECT COUNT(KundID) AS 'Antal kunder' FROM Kund

**8.B**

// Skapar lista som visar total artikelvärde som finns

SELECT FLOOR(SUM(Pris \* Antal)) AS 'Totala artikelvärdet' FROM Artikel

**8.C**

// Skapar lista som visar total artikelvärde, maxvärde minvärde, medelvärde

SELECT SUM(Antal) AS 'Antal artiklar i lager',

FLOOR(SUM(TotalPris)) AS 'Totala lagervärdet',

FLOOR(MAX(Pris)) AS 'Maxvärdet',

FLOOR(AVG(Pris)) AS 'Medelvärdet',

FLOOR(MIN(Pris)) AS 'Minvärdet'

FROM Artikel

**8.D**

// Skapar lista som visar total artikelvärde, maxvärde minvärde, medelvärde

SELECT SUM(Antal) AS 'Antal artiklar i lager',

FLOOR(SUM(TotalPris)) AS 'Totala lagervärdet',

FLOOR(MAX(Pris)) AS 'Maxvärdet',

FLOOR(AVG(Pris)) AS 'Medelvärdet',

FLOOR (MIN(Pris)) AS 'Minvärdet'

FROM Artikel

WHERE TotalPris > (SELECT AVG(TotalPris) FROM Artikel)

**8.E**

// Skapar lista där summeringen skär utifrån faktura inte fakturarad

SELECT F.FakturaID, MAX(Datum) AS Fakturadatum, SUM(((Fr.Pris \* Fr.Antal) - ((Fr.Pris \* Fr.Antal) \* Fr.Rabatt)) ) AS 'Summa exkl. moms', SUM(M.Moms\* ((Fr.Pris \* Fr.Antal) - ((Fr.Pris \* Fr.Antal) \* Fr.Rabatt)) + (Fr.Pris \* Fr.Antal) - ((Fr.Pris \* Fr.Antal) \* Fr.Rabatt) ) AS 'Summa inkl. moms'

FROM Faktura AS F

INNER JOIN Fakturad AS Fr

ON F.FakturaID = Fr.FakturaID

INNER JOIN Moms AS M

ON Fr.MomsID = M.MomsID

GROUP BY F.FakturaID

**9.A**

// Skapar tabell som omvandlar datum med tid till bara datum

SELECT CONVERT(VARCHAR(10), Datum, 120) Betvillkor, CONVERT(VARCHAR(10), DATEADD(DD, Betvillkor, Datum), 120) AS Förfallodatum,Artnamn,

Fr.Antal, Fr.Pris, (M.Moms \* 100) AS 'Moms i %', Fr.Rabatt, M.Moms\* ((Fr.Pris \* Fr.Antal) - ((Fr.Pris \* Fr.Antal) \* Fr.Rabatt)) + (Fr.Pris \* Fr.Antal) - ((Fr.Pris \* Fr.Antal) \* Fr.Rabatt ) AS Summa FROM Faktura AS F

INNER JOIN Fakturad AS Fr

ON F.FakturaID = Fr.FakturaID

INNER JOIN Artikel AS A

ON Fr.ArtikelID = A.ArtikelID

INNER JOIN Moms AS M

ON Fr.MomsID = M.MomsID

**9.B**

//visar kund tabbelsen, skapar kolumn som heter region och visar postnummer med bara 3 tecken

SELECT \*, LEFT (Postnr, 3) AS Region FROM Kund

**9.C**

//slår ihop två tabeller och sen gör bokstäverna stora och presenterar dem

SELECT UPPER(Artnamn) + ' ' + CAST(TotalPris AS varchar(20)) AS Artikel

FROM Artikel

**9.D**

//Visar min födelsedag, hur många dagar kvar till nästa år och vilken dag min födelsedag kommer vara i år

SELECT DATEDIFF(DAY, CONVERT(date, GETDATE()), CONVERT(date, '2015-12-31')) AS 'Dagar till Nyår',

DATEDIFF(DAY, CONVERT(date, GETDATE()), CONVERT(date, '2015-05-19')) AS 'Dagar till min Födelsedag',

DATENAME(DW, CONVERT(date, '2015-05-19')) AS 'Födelseveckodag'