

1. Project description: explain what your database project is about.

Our database system is produced for The Wholesale Clothing Company and its goal is to maintain the relationships between The Wholesale Clothing Company and its departments, employees, inventories, clothes, shops working with and logistic companies which are responsible for transportation of clothes between shops and our company.

2. Scope: what is included/excluded? Which processes are supported, which ones are not?

Included Business Processes

- Creating shipments from Producer to us and Us from Shops
- Storing clothes in inventory (by ShipmentID FK column in CLOTHING and INVENTORY table)
- Categorization of Clothes (type column in CLOTHING)
- Sale records are kept in SHIPMENT table
- Logistics company information are stored
- Storing employee and department info

Excluded Business Processes

- All clothing returning functionality (a.k.a change requests) are excluded
- Exporting to EXCEL will be done in web interface part

3. Data and requirements analysis for the database and business processes

Functional Requirements

- Creating, reading, updating and deletion of all entities.
- Filtering of clothes by their attributes.
- Exporting data to excel.
- Return and change features.
- Storage of sale records.

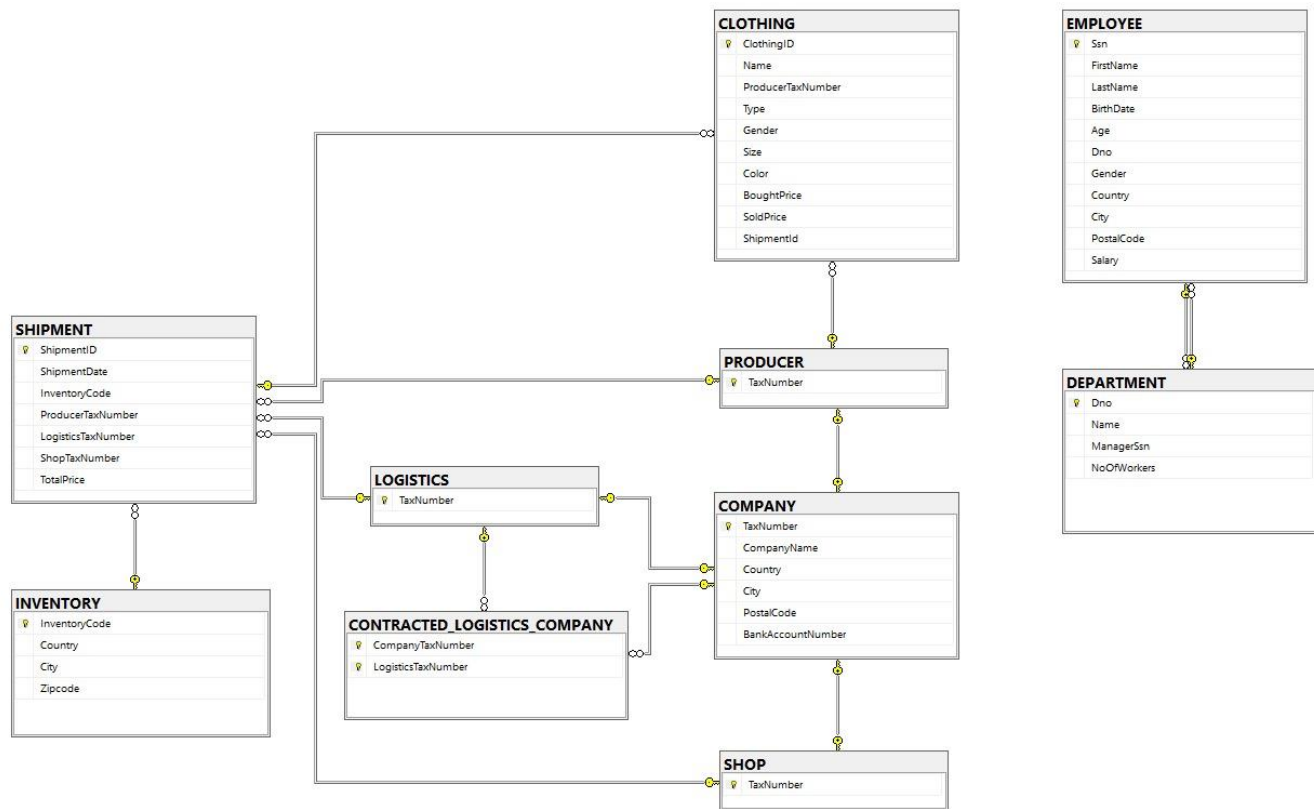
Non-Functional Requirements

- Mobile compliant web interface
- Flask framework along with Python will be for web interface.
- Pyodbc driver will be used.
- Microsoft SQL server will be used as database engine

Business Processes

- Producer produces the Clothes and ships them to our company by a Logistics company.
- Quality control Department checks whether the shipped clothes are deformed or not. If so, they are returned to Producer. Cost of deformed clothes are sent back to our bank account.
- Shipped clothes are kept in an Inventory and they are categorized by Employees that are working in operation Department.
- Marketing department arranges the shops demands and sales department sells them.
- In each sale, a sale record is kept.
- Clothes are sent to Shops by a Logistics company.
- If a return request is issued from a Shop, the Clothes are sent back to Inventory by Logistic company. Each return request is stored in database. The cost of returned clothes are sent back to shops bank account.
- If a change request is issued from a Shop, the Clothes are sent back to Inventory by Logistic company. Each change request is stored in database.

4. Diagram of whole database.



c) TABLES

1) CLOTHING TABLE

ii) Table which contains information of clothes which are produced by producers.

i, iii) Name Of the fields/columns and their data types

- ClothingID (int)
- Name (nvarchar(50))
- ProducerTaxNumber (int)
- Type (nvarchar(25))
- Gender (char(1))
- Size (nvarchar(5))
- Color (nvarchar(25))
- BoughtPrice (int)
- SoldPrice (int)
- ShipmentId (int)

iv)

Indexes = None

Primary Key = ClothingID

Foreign Keys are = CLOTHING_PRODUCER

CLOTHING_SHIPMENT

CLOTHING is connected with PRODUCER by ProducerTaxNumber,

CLOTHING is connected with SHIPMENT by ShipmentId

v)

Unique; None

Identity; ClothingID is identity

Check constraints; Gender is should be 'F' for Female and 'M' for

Male, others invalid

Defaults; None

Computed columns; None

vi) No Triggers.

2) COMPANY TABLE

ii) Table which contains information about companies that are producers, logistics and shops.

i,iii) Name Of the fields/columns and their data types

- TaxNumber (int)
- CompanyName (nvarchar(50))
- Country (nvarchar(25))
- City (nvarchar(25))
- PostalCode (nchar(5))
- BankAccountNumber(nvarchar(50))

iv)

Indexes = None

Primary Key = TaxNumber

Foreign Key = None

v)

Unique = BankAccountNumber

Identity = None

Check constraints = None

Defaults = None

Computed columns = None

vi) No Triggers.

3) CONTRACTED LOGISTICS COMPANY TABLE

ii) This table keeps which logistics work with which company

i, iii) Name Of the fields/columns and their data types

- CompanyTaxNumber (int)
- LogisticsTaxNumber (int)

iv)

Indexes = None

Primary Keys are = CompanyTaxNumber

LogisticsTaxNumber

Foreign Keys are = CONTRACTED_LOGISTICS_COMPANY_COMPANY

CONTRACTED_LOGISTICS_COMPANY_LOGISTICS

CONTRACTED_LOGISTICS_COMPANY is connected with company with
COMPANY by CompanyTaxNumber

CONTRACTED_LOGISTICS_COMPANY is connected with LOGISTICS by
LogisticsTaxNumber

v)

Unique = None

Identity = None

Check constraints = None

Defaults = None

Computed columns = None

vi) No Triggers.

4) DEPARTMENT TABLE

ii) Table which contains information about departments.

i,iii) Name Of the fields/columns and their data types

- Dno (int)
- Name (nvarchar(50))
- ManagerSsn (int)
- NoOfWorkers (int)

iv)

Indexes = None

Primary Key = Dno

Foreign Key = DEPARTMEN_EMPLOYEE

DEPARTMENT is connected with EMPLOYEE by ManagerSsn.

v)

Unique = None

Identitiy = None

Check constraints = None

Defaults = None

Computed columns = None

vi) No Triggers.

5) EMPLOYEE TABLE

ii) Table which contains information about Employees.

i,iii)) Name Of the fields/columns and their data types

- Ssn (int)
- FirstName (nvarchar(50))
- LastName (nvarchar(50))
- BirthDate (smalldatetime)
- Age (int)
- Dno (int)
- Gender (char(1))
- Country (nvarchar(25))
- City (nvarchar(25))
- PostalCode (nchar(5))
- Salary (int)

iv)

Indexes = BirthDate

Primary Key = Ssn

Foreign Key = EMPLOYEE_DEPARTMENT

EMPLOYEE is connected with DEPARTMENT by Dno.

v)

Unique = None

Identity = None

Check constraints = Gender should be 'F' for Female and 'M' for Male, others invalid

Defaults = Country is Turkey

Computed columns = Age

vi)

- tg_DecrementNoOfWorkers = this trigger automatically decrease NoOfWorkers in department table when employee is deleted.
- tg_IncrementNoOfWorkers = this trigger automatically increase NoOfWorkers in department table when employee is added to the employee table.

6) INVENTORY TABLE

ii) Table which contains information about where are our inventories.

i,iii) Name Of the fields/columns and their data types

- InventoryCode (int)
- Country (nvarchar(25))
- City (nvarchar(25))
- ZipCode (nvarchar(25))

iv)

Indexes = None

Primary Key = InventoryCode

Foreign Key = None

v)

Unique = None

Identity = None

Check constraints = None

Defaults = None

Computed columns = None

vi) No Trigger.

7) LOGISTICS TABLE

ii) Table which contains information about companies Tax Numbers.

i,iii)) Name Of the fields/columns and their data types

- TaxNumber (int)

iv)

Indexes = None

Primary Key = TaxNumber

Foreign Key = LOGISTICS_COMPANY

LOGISTICS is connected with COMPANY by TaxNumber

v)

Unique = None

Identitiy = None

Check constraints = None

Defaults = None

Computed columns = None

vi) No Trigger.

8) PRODUCER TABLE

ii) Table which contains information of producers.

i, iii) Name Of the fields/columns and their data types

- TaxNumber (int)

iv)

Indexes = None

Primary Key = TaxNumber

Foreign Keys are = PRODUCER_COMPANY

PRODUCER is connected with COMPANY by TaxNumber

v)

Unique; None

Identitiy; None

Check constraints; None

Defaults; None

Computed columns; None

vi) No triggers.

9) SHIPMENT TABLE

ii) Table which contains information of shipment which made by producer to our inventory or our inventory to shop.

i,iii) Name Of the fields/columns and their data types

- ShipmentID (int)
- ShipmentDate (smalldatetime)
- InventoryCode (int)
- ProducerTaxNumber (int)
- LogisticsTaxNumber (int)
- ShopTaxNumber (int)
- TotalPrice (int)

iv)

Indexes = None

Primary Key = ShipmentID

Foreign Keys are = SHIPMENT_INVENTORY, SHIPMENT_PRODUCER, SHIPMENT_LOGISTICS, SHIPMENT_SHOP

SHIPMENT is connected with INVENTORY by InventoryCode,

SHIPMENT is connected with PRODUCER by ProducerTaxNumber,

SHIPMENT is connected with LOGISTICS by LogisticsTaxNumber,

SHIPMENT is connected with SHOP by ShopTaxNumber

v)

Unique = None

Identity = ShipmentID

Check constraints = None

Defaults = None

Computed columns = None

vi) No Triggers.

10) SHOP TABLE

ii) Table which contains information of shops.

i, iii) Name Of the fields/columns and their data types

- TaxNumber (int)

iv)

Indexes = None

Primary Keys are = TaxNumber

Foreign Keys are = SHOP_COMPANY

SHOP_COMPANY is connected with COMPANY by TaxNumber

v)

Unique = None

Identity = None

Check constraints = None

Defaults = None

Computed columns = None

vi) No triggers.

d) Views

- 1) **ClothingInventory**: Table which contains id, name of clothes and their inventory codes.

```
Create View [dbo].[ClothingInventory]
As
Select c.ClothingID, c.Name, i.InventoryCode
From CLOTHING c inner join SHIPMENT s on c.ShipmentId=s.ShipmentID
inner join INVENTORY i on s.InventoryCode=i.InventoryCode
```

	ClothingID	Name	InventoryCode
1	1	Oduncu Gomlegi	1
2	2	Boxer	1
3	3	Palazzo	1
4	4	Kaz Tüyü Kaban	2
5	5	Deri Ceket	2
6	6	Genç Çorap	3
7	7	Örme Eşofman Altı	3
8	8	Şişme Mont	1
9	9	Kareli Gömlek	1
10	10	Dik Yaka Şişme Mont	2
11	11	Mini Etek	2
12	12	Polo Yaka T-shirt	2
13	13	Zümrüt Yeşili Elbise	3
14	14	Pijama Altı	3
15	15	Yüksek Bel Pantolon	3
16	16	Oversize Ceket	1
17	17	Kapüşonlu Hırka	1
18	18	Örme Bluz	1
19	19	Basic T-shirt	1
20	20	Bisiklet Yaka T-shirt	1
21	21	Kumaş Pantolon	3
22	22	Sade Gömlek	3
23	23	Spor Çorap	3
24	24	V Yaka T-shirt	3
25	25	Pijama Üstü	3
26	26	Siyah Mont Çocuk	2
27	27	Beyaz Mont Çocuk	2

- 2) **ClothingTypes**: Table which contains clothing types and amount of them.

```
Create VIEW [dbo].[ClothingTypes]
AS
SELECT c.Type, count(*) as Amount
FROM CLOTHING c
Group By c.Type
```

	Type	Amount
1	BLUZ	1
2	CEKET	2
3	ELBİSE	1
4	EŞOFMAN	1
5	ETEK	1
6	GÖMLEK	3
7	HIRKA	1
8	İÇ GİYİM	3
9	KABAN	1
10	MONT	4
11	PANTOLON	3
12	PIJAMA	2
13	T-SHIRT	4

- 3) **DepartmentsOfEmployees:** Table which contains ssnn, full name of employees and their departments' department number and department name.

```
Create VIEW [dbo].[DepartmentsOfEmployees]
AS
SELECT e.Ssn,e.FirstName + ' ' + e.LastName as FullName,e.Dno,d.Name
FROM EMPLOYEE e inner join DEPARTMENT d on e.Dno=d.Dno
```

	Ssn	FullName	Dno	Name
1	101	CEM ANARAL	1	IT
2	102	HASAN ŞENYURT	1	IT
3	103	ALİ YETİM	1	IT
4	104	NİMET ŞAHİN	1	IT
5	105	ABDULLAH PEKMEZ	1	IT
6	201	MELİSA DURMUŞ	2	MARKETING
7	202	AYŞE KESKİN	2	MARKETING
8	203	ZEYNEP TÜRKOĞLU	2	MARKETING
9	204	BİLGE DELEN	2	MARKETING
10	205	KEMALETTİN KURUR	2	MARKETING
11	301	EZGİ MOLA	3	SALES
12	302	ASLAN ASLANKAYA	3	SALES
13	303	MURAT BELEK	3	SALES
14	304	BAHAR YILAR	3	SALES
15	305	SEHEL CAN	3	SALES
16	401	RECEP TORLU	4	OPERATION
17	402	MUHAMMED OLCAN	4	OPERATION
18	403	MUSTAFA ŞENTÜRK	4	OPERATION
19	404	EGE OĞLAK	4	OPERATION
20	405	SİMGE ÖZER	4	OPERATION
21	501	MEHMET DEMİR	5	QUALITY ...
22	502	FATİH ŞİMŞEK	5	QUALITY ...
23	503	MELEK TAŞKIN	5	QUALITY ...
24	504	ALİHAN KONUK	5	QUALITY ...
25	505	İLAYDA GÖREN	5	QUALITY ...

- 4) **IncomingShipments:** Table which contains incoming shipment information (producer to our inventory). Shipment Id, Shipment date, inventory code, producer's tax number, logistics' tax number and total price of shipment are shown in the table.

```
CREATE VIEW [dbo].[IncomingShipments]
AS
SELECT s.ShipmentID,s.ShipmentDate,s.InventoryCode,s.ProducerTaxNumber,s.LogisticsTaxNumber,s.TotalPrice
FROM SHIPMENT s
Where s.ShopTaxNumber is NULL
```

	ShipmentID	ShipmentDate	InventoryCode	ProducerTaxNumber	LogisticsTaxNumber	TotalPrice
1	9000	2021-03-03 00:00:00	1	1000	1010	585
2	9002	2021-06-20 00:00:00	3	1005	1010	570
3	9004	2021-09-11 00:00:00	3	1000	1010	570
4	9005	2021-12-23 14:48:00	2	1000	1015	300

- 5) **Managers:** Table which contains ssn, full name, department number and department name of managers of each department.

```
Create VIEW [dbo].[Managers]
AS
SELECT e.Ssn,e.FirstName + ' ' + e.LastName as FullName, e.Dno,d.Name
FROM EMPLOYEE e inner join DEPARTMENT d on e.Ssn=d.ManagerSsn
```

	Ssn	FullName	Dno	Name
1	103	ALİ YETİM	1	IT
2	201	MELİSA DURMUŞ	2	MARKETING
3	301	EZGİ MOLA	3	SALES
4	401	RECEP TORLU	4	OPERATION
5	501	MEHMET DEMİR	5	QUALITY CONTROL

- 6) **NameOfContractedLogistics:** Table which contains names of producers, shops and logistic companies which work with them.

```
Create VIEW [dbo].[NameOfContractedLogistics]
AS
SELECT c.CompanyName, log.CompanyName as LogisticsName
FROM COMPANY c inner join CONTRACTED_LOGISTICS_COMPANY clc on c.TaxNumber=clc.CompanyTaxNumber
inner join
(select c2.TaxNumber, c2.CompanyName from COMPANY c2 inner join LOGISTICS l2 on c2.TaxNumber=l2.TaxNumber) log
on log.TaxNumber=clc.LogisticsTaxNumber
```

	CompanyName	LogisticsName
1	Kardeşler Tekstil	Aras Kargo
2	Kaplan Tuhafiye	MNG Kargo
3	Zara Bağdat Caddesi	MNG Kargo
4	H&M Maltepe	MNG Kargo

- 7) **OutgoingShipments:** Table which contains outgoing shipment information (our inventory to shops). Shipment Id, Shipment date, inventory code, shop's tax number, logistics' tax number and total price of shipment are shown in the table.

```
Create VIEW [dbo].[OutgoingShipments]
AS
SELECT s.ShipmentID,s.ShipmentDate,s.InventoryCode,s.ShopTaxNumber,s.LogisticsTaxNumber,s.TotalPrice
FROM SHIPMENT s
Where s.ProducerTaxNumber is NULL
```

	ShipmentID	ShipmentDate	InventoryCode	ShopTaxNumber	LogisticsTaxNumber	TotalPrice
1	9001	2021-03-04 00:00:00	2	1020	1015	1280
2	9003	2021-12-09 00:00:00	1	1030	1015	670
3	9006	2021-12-23 14:50:00	1	1020	1010	NULL

- 8) **ProfitOfClothes:** Table which contains information of clothes and their profits (sold price – bought price).

```
Create VIEW [dbo].[ProfitOfClothes]
AS
SELECT c.ClothingID,c.Name,c.Type,c.Color,c.Size,c.SoldPrice - c.BoughtPrice as Profit
FROM CLOTHING c
```

	ClothingID	Name	Type	Color	Size	Profit
1	1	Oduncu Gomlegi	GÖMLEK	TURUNCU	S	30
2	2	Boxer	İÇ GİYİM	MAVİ	XXL	15
3	3	Palazzo	PANTOLON	SİYAH	S	40
4	4	Kaz Tüyü Kaban	KABAN	KAHVERENGİ	XL	50
5	5	Deri Ceket	CEKET	SİYAH	M	70
6	6	Genç Çorap	İÇ GİYİM	GİRİ	L	7
7	7	Örme Eşofman Altı	EŞOFMAN	SİYAH	S	30
8	8	Şişme Mont	MONT	YEŞİL	XL	40
9	9	Kareli Gömlek	GÖMLEK	BEYAZ	L	25
10	10	Dik Yaka Şişme Mont	MONT	SİYAH	M	40
11	11	Mini Etek	ETEK	KIRMIZI	XS	40
12	12	Polo Yaka T-shirt	T-SHIRT	SARI	L	40
13	13	Zümrüt Yeşili Elbise	ELBİSE	YEŞİL	M	50
14	14	Pijama Altı	PİJAMA	SARI	M	30
15	15	Yüksek Bel Pantolon	PANTOLON	BEYAZ	L	30
16	16	Oversize Ceket	CEKET	SİYAH	L	50
17	17	Kapüşonlu Hırka	HIRKA	GİRİ	S	40
18	18	Örme Bluz	BLUZ	KIRMIZI	M	30
19	19	Basic T-shirt	T-SHIRT	SİYAH	L	30
20	20	Bisiklet Yaka T-shirt	T-SHIRT	MAVİ	XL	40
21	21	Kumaş Pantolon	PANTOLON	SİYAH	L	35
22	22	Sade Gömlek	GÖMLEK	BEYAZ	XS	30
23	23	Spor Çorap	İÇ GİYİM	MAVİ	S	10
24	24	V Yaka T-shirt	T-SHIRT	MOR	XXL	40
25	25	Pijama Üstü	PİJAMA	YEŞİL	XXS	20
26	26	Siyah Mont Çocuk	MONT	SİYAH	S	50
27	27	Beyaz Mont Çocuk	MONT	BEYAZ	S	50

- 9) **TopFiveEarnerEmployee:** Table which contains ssn, full name and salary of employees whose salary is in top five in all company.

```
CREATE VIEW [dbo].[TopFiveEarnerEmployee]
AS
SELECT Top 5 e.Ssn,e.FirstName + ' ' + e.LastName as FullName,e.Salary
FROM EMPLOYEE e
Order By e.Salary desc
```

	Ssn	FullName	Salary
1	401	RECEP TORLU	7000
2	501	MEHMET DEMİR	7000
3	405	SİMGE ÖZER	6000
4	502	FATİH ŞİMŞEK	6000
5	503	MELEK TAŞKIN	6000

e) Triggers

- 1) **tg_IncrementNoOfWorkers:** Increases number of employees by one in the department that added employee works when new employee is added.

```
ALTER TRIGGER [dbo].[tg_IncrementNoOfWorkers]
on [dbo].[EMPLOYEE]
AFTER INSERT
As
Begin
    Update DEPARTMENT
    Set NoOfWorkers=NoOfWorkers + 1
    Where Dno=(Select Dno From Inserted)
End
```

Before the trigger:

	Dno	Name	ManagerSsn	NoOfWorkers
1	1	IT	103	5
2	2	MARKETING	201	5
3	3	SALES	301	5
4	4	OPERATION	401	5
5	5	QUALITY CONTROL	501	5

Inserting new employee to dno=4:

```
Insert Into EMPLOYEE(Ssn,Dno) Values (406,4)
```

After the trigger:

	Dno	Name	ManagerSsn	NoOfWorkers
1	1	IT	103	5
2	2	MARKETING	201	5
3	3	SALES	301	5
4	4	OPERATION	401	6
5	5	QUALITY CONTROL	501	5

- 2) **tg_DecrementNoOfWorkers:** Decreases number of employees by one in the department that deleted employee works when employee is deleted.

```
ALTER TRIGGER [dbo].[tg_DecrementNoOfWorkers]
on [dbo].[EMPLOYEE]
AFTER DELETE
As
Begin

    Update DEPARTMENT
    Set NoOfWorkers=NoOfWorkers - 1
    Where Dno=(Select Dno From Deleted)
End
```

Before the trigger:

	Dno	Name	ManagerSsn	NoOfWorkers
1	1	IT	103	5
2	2	MARKETING	201	5
3	3	SALES	301	5
4	4	OPERATION	401	6
5	5	QUALITY CONTROL	501	5

Deleting employee whose ssn=406, dno=4:

```
Delete From EMPLOYEE Where Ssn=406 and Dno=4
```


After the trigger:

	Dno	Name	ManagerSsn	NoOfWorkers
1	1	IT	103	5
2	2	MARKETING	201	5
3	3	SALES	301	5
4	4	OPERATION	401	5
5	5	QUALITY CONTROL	501	5

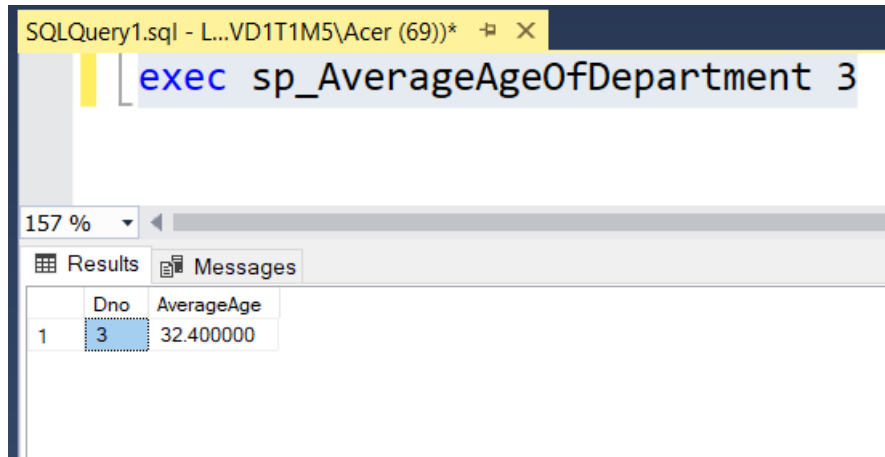
Stored Procedures

1 - sp_AverageAgeOfDepartment

Shows the average age of employees working in given department Dno

```
CREATE Procedure sp_AverageAgeOfDepartment
    @dno int
As
Begin
    Select d.Dno, avg(e.Age * 1.0) as AverageAge
    From EMPLOYEE e inner join DEPARTMENT d on e.Dno=d.Dno
    Group By d.Dno
    Having @dno = d.Dno

End
```



2 - sp_CreateClothing

Inserts clothing to CLOTHING table, calls sp_UpdateTotalPrice to update the total price of shipments after creation

```
Create Procedure sp_CreateClothing
    @Name nvarchar(50),
    @PTax int,
    @Type nvarchar(25),
    @Gender char(1),
    @Size nvarchar(5),
    @Color nvarchar(25),
    @BPrice int,
    @SPrice int,
    @ShipmentId int
As
Begin
    Insert Into CLOTHING (Name, ProducerTaxNumber, Type, Gender, Size, Color,
    BoughtPrice, SoldPrice, ShipmentId)
    Values (@Name, @PTax, @Type, @Gender, @Size, @Color, @BPrice, @SPrice,
    @ShipmentId)
    exec sp_UpdateTotalPrice

End
```

```
select * from CLOTHING
select * from SHIPMENT
```

157 %

Results Messages

	ClothingID	Name	ProducerTaxNumber	Type	Gender	Size	Color	BoughtPrice	SoldPrice	ShipmentId
20	20	Bisiklet Yaka T-shirt	1005	T-SHIRT	F	XL	MAVİ	80	120	9003
21	21	Kumaş Pantolon	1000	PANTOLON	M	L	SIYAH	100	135	9004
22	22	Sade Gömlek	1000	GÖMLEK	M	XS	BEYAZ	110	140	9004
23	23	Spor Çorap	1000	İÇ GIYIM	M	S	MAVİ	15	25	9004
24	24	V Yaka T-shirt	1000	T-SHIRT	F	XXL	MOR	150	190	9004
25	25	Pijama Üstü	1000	PIJAMA	M	XXS	YEŞİL	60	80	9004
26	26	Siyah Mont Çocuk	1000	MONT	M	S	SIYAH	100	150	9005
27	27	Beyaz Mont Çocuk	1000	MONT	M	S	BEYAZ	100	150	9005

	ShipmentID	ShipmentDate	InventoryCode	ProducerTaxNumber	LogisticsTaxNumber	ShopTaxNumber	TotalPrice
1	9000	2021-03-03 00:00:00	1	1000	1010	NULL	585
2	9001	2021-03-04 00:00:00	2	NULL	1015	1020	1280
3	9002	2021-06-20 00:00:00	3	1005	1010	NULL	570
4	9003	2021-12-09 00:00:00	1	NULL	1015	1030	670
5	9004	2021-09-11 00:00:00	3	1000	1010	NULL	570
6	9005	2021-12-23 14:48:00	2	1000	1015	NULL	300
7	9006	2021-12-23 14:50:00	1	NULL	1010	1020	NULL

```
exec sp_CreateClothing 'Sarı Mont', 1000, 'MONT', 'M', 'S', 'SARI', 100, 150, 9005
select * from CLOTHING
select * from SHIPMENT
```

143 %

Results Messages

	ClothingID	Name	ProducerTaxNumber	Type	Gender	Size	Color	BoughtPrice	SoldPrice	ShipmentId
21	21	Kumaş Pantolon	1000	PANTOLON	M	L	SIYAH	100	135	9004
22	22	Sade Gömlek	1000	GÖMLEK	M	XS	BEYAZ	110	140	9004
23	23	Spor Çorap	1000	İÇ GIYIM	M	S	MAVİ	15	25	9004
24	24	V Yaka T-shirt	1000	T-SHIRT	F	XXL	MOR	150	190	9004
25	25	Pijama Üstü	1000	PIJAMA	M	XXS	YEŞİL	60	80	9004
26	26	Siyah Mont Çocuk	1000	MONT	M	S	SIYAH	100	150	9005
27	27	Beyaz Mont Çocuk	1000	MONT	M	S	BEYAZ	100	150	9005
28	28	Sarı Mont	1000	MONT	M	S	SARI	100	150	9005

	ShipmentID	ShipmentDate	InventoryCode	ProducerTaxNumber	LogisticsTaxNumber	ShopTaxNumber	TotalPrice
1	9000	2021-03-03 00:00:00	1	1000	1010	NULL	585
2	9001	2021-03-04 00:00:00	2	NULL	1015	1020	1280
3	9002	2021-06-20 00:00:00	3	1005	1010	NULL	570
4	9003	2021-12-09 00:00:00	1	NULL	1015	1030	670
5	9004	2021-09-11 00:00:00	3	1000	1010	NULL	570
6	9005	2021-12-23 14:48:00	2	1000	1015	NULL	450
7	9006	2021-12-23 14:50:00	1	NULL	1010	1020	NULL

! TotalPrice is updated

3 - sp_CreateDepartment

Creates a new department entry to DEPARTMENT. Initially, ManagerSsn is null and NoOfWorkers is 0.

```
CREATE Procedure sp_CreateDepartment(
    @Dno int,
    @Name nvarchar(50)
)
As
Begin
    Insert Into DEPARTMENT
    Values (@Dno, @Name, NULL, 0)
End
```

```
select * from DEPARTMENT
```

143 %

Results Messages

	Dno	Name	ManagerSsn	NoOfWorkers
1	1	IT	103	5
2	2	MARKETING	201	5
3	3	SALES	301	5
4	4	OPERATION	401	5
5	5	QUALITY CONTROL	501	5

```
exec sp_CreateDepartment 99, 'Boş işler müdürlüğü'
select * from DEPARTMENT
```

143 %

Results Messages

	Dno	Name	ManagerSsn	NoOfWorkers
1	1	IT	103	5
2	2	MARKETING	201	5
3	3	SALES	301	5
4	4	OPERATION	401	5
5	5	QUALITY CONTROL	501	5
6	99	Boş işler müdürlüğü	NULL	0

4 – sp_CreateEmployee

Creates a new employee record to EMPLOYEE table

```
CREATE Procedure sp_CreateEmployee
```

```
@Ssn int,
@FirstName nvarchar(50),
@LastName nvarchar(50),
@BirthDate smalldatetime,
@Dno int,
@Gender char(1),
@Country nvarchar(25),
@City nvarchar(25),
@PostalCode nchar(5),
@Salary int
```

```
As
Begin
```

```
Insert Into EMPLOYEE Values
(@Ssn,
@FirstName,
@LastName,
@BirthDate,
@Dno,
@Gender,
@Country,
@City,
@PostalCode,
@Salary)
```

```
End
```

select * from EMPLOYEE

107 %

Results Messages

	Ssn	FirstName	LastName	BirthDate	Age	Dno	Gender	Country	City	PostalCode	Salary
1	101	CEM	ANARAL	2001-07-18 00:00:00	20	1	M	TURKEY	İSTANBUL	34123	5000
2	102	HASAN	ŞENYURT	2000-02-15 00:00:00	21	1	M	TURKEY	İSTANBUL	34222	2000
3	103	ALİ	YETİM	2001-09-29 00:00:00	20	1	M	TURKEY	İSTANBUL	34142	2000
4	104	NİMET	ŞAHİN	1970-11-08 00:00:00	51	1	F	TURKEY	İSTANBUL	34414	4000
5	105	ABDULLAH	PEKMEZ	1993-01-03 00:00:00	28	1	M	TURKEY	İSTANBUL	34122	3500
6	201	MELİSA	DURMUŞ	2000-08-24 00:00:00	21	2	F	TURKEY	İSTANBUL	34678	4000
7	202	AYŞE	KESKİN	2000-07-21 00:00:00	21	2	F	TURKEY	İSTANBUL	34567	3000
8	203	ZEYNEP	TÜRKOĞLU	2000-03-15 00:00:00	21	2	F	TURKEY	İSTANBUL	34789	3000
9	204	BİLGE	DELEN	1989-05-13 00:00:00	32	2	M	TURKEY	İSTANBUL	34321	3700
10	205	KEMALETİN	KURUR	1967-09-21 00:00:00	54	2	M	TURKEY	İSTANBUL	34131	3000
11	301	EZGİ	MOLA	1998-02-23 00:00:00	23	3	F	TURKEY	İSTANBUL	34125	5000
12	302	ASLAN	ASLANKAYA	1995-05-19 00:00:00	26	3	M	TURKEY	İSTANBUL	34789	4000
13	303	MURAT	BELEK	1978-12-12 00:00:00	43	3	M	TURKEY	İSTANBUL	34567	4000
14	304	BAHAR	YILAR	1999-02-13 00:00:00	22	3	F	TURKEY	İSTANBUL	34133	4000
15	305	SEHEL	CAN	1973-04-28 00:00:00	48	3	F	TURKEY	İSTANBUL	34743	4500
16	401	RECEP	TORLU	1989-03-26 00:00:00	32	4	M	TURKEY	İSTANBUL	34123	7000
17	402	MUHAMMED	OLCAN	1992-06-16 00:00:00	29	4	M	TURKEY	ANKARA	06413	5000
18	403	MUSTAFA	ŞENTÜRK	1996-09-16 00:00:00	25	4	M	TURKEY	İZMİR	35412	5000
19	404	EGE	OĞLAK	1981-02-15 00:00:00	40	4	M	TURKEY	İSTANBUL	34832	5000
20	405	SİMGE	ÖZER	2000-05-12 00:00:00	21	4	F	TURKEY	İSTANBUL	34412	6000
21	501	MEHMET	DEMİR	1972-08-30 00:00:00	49	5	M	TURKEY	İSTANBUL	34987	7000
22	502	FATİH	ŞİMŞEK	1988-03-19 00:00:00	33	5	M	TURKEY	İSTANBUL	34344	6000
23	503	MELEK	TAŞKIN	1972-12-12 00:00:00	49	5	F	TURKEY	İSTANBUL	34423	6000
24	504	ALİHAN	KONUK	2000-03-19 00:00:00	21	5	M	TURKEY	İSTANBUL	34413	4000
25	505	İLAYDA	GÖREN	1997-08-06 00:00:00	24	5	F	TURKEY	İSTANBUL	34739	4000

exec sp_CreateEmployee 9999, 'Mustafa', 'Agaoglu', '1980-01-01 00:00:00', 1, 'M', 'TURKEY', 'İSTANBUL', '34000', 10999

select * from EMPLOYEE

107 %

Results Messages

	Ssn	FirstName	LastName	BirthDate	Age	Dno	Gender	Country	City	PostalCode	Salary
10	205	KEMALETİN	KURUR	1967-09-21 00:00:00	54	2	M	TURKEY	İSTANBUL	34131	3000
11	301	EZGİ	MOLA	1998-02-23 00:00:00	23	3	F	TURKEY	İSTANBUL	34125	5000
12	302	ASLAN	ASLANKAYA	1995-05-19 00:00:00	26	3	M	TURKEY	İSTANBUL	34789	4000
13	303	MURAT	BELEK	1978-12-12 00:00:00	43	3	M	TURKEY	İSTANBUL	34567	4000
14	304	BAHAR	YILAR	1999-02-13 00:00:00	22	3	F	TURKEY	İSTANBUL	34133	4000
15	305	SEHEL	CAN	1973-04-28 00:00:00	48	3	F	TURKEY	İSTANBUL	34743	4500
16	401	RECEP	TORLU	1989-03-26 00:00:00	32	4	M	TURKEY	İSTANBUL	34123	7000
17	402	MUHAMMED	OLCAN	1992-06-16 00:00:00	29	4	M	TURKEY	ANKARA	06413	5000
18	403	MUSTAFA	ŞENTÜRK	1996-09-16 00:00:00	25	4	M	TURKEY	İZMİR	35412	5000
19	404	EGE	OĞLAK	1981-02-15 00:00:00	40	4	M	TURKEY	İSTANBUL	34832	5000
20	405	SİMGE	ÖZER	2000-05-12 00:00:00	21	4	F	TURKEY	İSTANBUL	34412	6000
21	501	MEHMET	DEMİR	1972-08-30 00:00:00	49	5	M	TURKEY	İSTANBUL	34987	7000
22	502	FATİH	ŞİMŞEK	1988-03-19 00:00:00	33	5	M	TURKEY	İSTANBUL	34344	6000
23	503	MELEK	TAŞKIN	1972-12-12 00:00:00	49	5	F	TURKEY	İSTANBUL	34423	6000
24	504	ALİHAN	KONUK	2000-03-19 00:00:00	21	5	M	TURKEY	İSTANBUL	34413	4000
25	505	İLAYDA	GÖREN	1997-08-06 00:00:00	24	5	F	TURKEY	İSTANBUL	34739	4000
26	9999	Mustafa	Agaoglu	1980-01-01 00:00:00	41	1	M	TURKEY	İSTANBUL	34000	10999

Query executed successfully.

LAPTOP-JVD1T1M5 (15.0 RTM) | LAPTOP-JVD1T1M5\Acer (70) | WHOLESALE CLOTHIN

5 – sp_CreateIncomingShipment

Adds incoming shipment to SHIPMENT table. Since it came from a producer, ShopTaxNumber column for the new entry will be null.

Create Procedure sp_CreateIncomingShipment

```
@InventoryCode int, @ProducerTaxNumber int, @LogisticsTaxNumber int
As
Begin
```

```
Insert Into SHIPMENT (ShipmentDate, InventoryCode, ProducerTaxNumber,
LogisticsTaxNumber)
```

```
Values (GETDATE(), @InventoryCode,@ProducerTaxNumber,@LogisticsTaxNumber)
```

```
End
```

```
select * from SHIPMENT
```

107 %

Results Messages

	ShipmentID	ShipmentDate	InventoryCode	ProducerTaxNumber	LogisticsTaxNumber	ShopTaxNumber	TotalPrice
1	9000	2021-03-03 00:00:00	1	1000	1010	NULL	585
2	9001	2021-03-04 00:00:00	2	NULL	1015	1020	1280
3	9002	2021-06-20 00:00:00	3	1005	1010	NULL	570
4	9003	2021-12-09 00:00:00	1	NULL	1015	1030	670
5	9004	2021-09-11 00:00:00	3	1000	1010	NULL	570
6	9005	2021-12-23 14:48:00	2	1000	1015	NULL	450
7	9006	2021-12-23 14:50:00	1	NULL	1010	1020	NULL

```
exec sp_CreateIncomingShipment 3, 1000, 1010
```

```
select * from SHIPMENT
```

107 %

Results Messages

	ShipmentID	ShipmentDate	InventoryCode	ProducerTaxNumber	LogisticsTaxNumber	ShopTaxNumber	TotalPrice
1	9000	2021-03-03 00:00:00	1	1000	1010	NULL	585
2	9001	2021-03-04 00:00:00	2	NULL	1015	1020	1280
3	9002	2021-06-20 00:00:00	3	1005	1010	NULL	570
4	9003	2021-12-09 00:00:00	1	NULL	1015	1030	670
5	9004	2021-09-11 00:00:00	3	1000	1010	NULL	570
6	9005	2021-12-23 14:48:00	2	1000	1015	NULL	450
7	9006	2021-12-23 14:50:00	1	NULL	1010	1020	NULL
8	9007	2021-12-23 15:56:00	3	1000	1010	NULL	NULL

6 – sp_CreateOutgoingShipment

Adds new shipment entry to SHIPMENT table. ProducerTaxNumber will be NULL since we are shipping it to shops.

```
CREATE Procedure sp_CreateOutgoingShipment
```

```
    @InventoryCode int, @ShopTaxNumber int, @LogisticsTaxNumber int
```

```
As
```

```
Begin
```

```
    Insert Into SHIPMENT (ShipmentDate, InventoryCode, ShopTaxNumber,
LogisticsTaxNumber)
```

```
    Values (GETDATE(), @InventoryCode,@ShopTaxNumber,@LogisticsTaxNumber)
```

```
End
```

```
select * from SHIPMENT
```

107 %

Results Messages

	ShipmentID	ShipmentDate	InventoryCode	ProducerTaxNumber	LogisticsTaxNumber	ShopTaxNumber	TotalPrice
1	9000	2021-03-03 00:00:00	1	1000	1010	NULL	585
2	9001	2021-03-04 00:00:00	2	NULL	1015	1020	1280
3	9002	2021-06-20 00:00:00	3	1005	1010	NULL	570
4	9003	2021-12-09 00:00:00	1	NULL	1015	1030	670
5	9004	2021-09-11 00:00:00	3	1000	1010	NULL	570
6	9005	2021-12-23 14:48:00	2	1000	1015	NULL	450
7	9006	2021-12-23 14:50:00	1	NULL	1010	1020	NULL
8	9007	2021-12-23 15:56:00	3	1000	1010	NULL	NULL

```
exec sp_CreateOutgoingShipment 3, 1020, 1015
select * from SHIPMENT
```

107 %

Results Messages

	ShipmentID	ShipmentDate	InventoryCode	ProducerTaxNumber	LogisticsTaxNumber	ShopTaxNumber	TotalPrice
1	9000	2021-03-03 00:00:00	1	1000	1010	NULL	585
2	9001	2021-03-04 00:00:00	2	NULL	1015	1020	1280
3	9002	2021-06-20 00:00:00	3	1005	1010	NULL	570
4	9003	2021-12-09 00:00:00	1	NULL	1015	1030	670
5	9004	2021-09-11 00:00:00	3	1000	1010	NULL	570
6	9005	2021-12-23 14:48:00	2	1000	1015	NULL	450
7	9006	2021-12-23 14:50:00	1	NULL	1010	1020	NULL
8	9007	2021-12-23 15:50:00	3	1000	1015	NULL	NULL
9	9008	2021-12-23 16:04:00	3	NULL	1015	1020	NULL

7 – sp_CreateProducer

Creates a new entry to COMPANY table, adds TaxNumber FK to PRODUCER table since there is an inheritance between them.

```
CREATE Procedure sp_CreateProducer
    @TaxNumber int,
    @CompanyName nvarchar(50),
    @Country nvarchar(25),
    @City nvarchar(25),
    @PostalCode nchar(5),
    @BankAccountNumber nvarchar(50)
As
Begin
    Insert Into COMPANY Values (
        @TaxNumber,
        @CompanyName,
        @Country,
        @City,
        @PostalCode,
        @BankAccountNumber
    )

    Insert Into PRODUCER Values (@TaxNumber)
End
```

```
select * from COMPANY
select * from PRODUCER
```

107 %

Results Messages

	TaxNumber	CompanyName	Country	City	PostalCode	BankAccountNumber
1	1000	Kardeşler Tekstil	TURKEY	ISPARTA	12313	5000
2	1005	Kaplan Tuhafiye	TURKEY	KOCAELİ	54213	5001
3	1010	Aras Kargo	TURKEY	İSTANBUL	12312	5002
4	1015	MNG Kargo	TURKEY	İSTANBUL	34521	5003
5	1020	Zara Bağdat Caddesi	TURKEY	İSTANBUL	13321	5004
6	1030	H&M Maltepe	TURKEY	İSTANBUL	33131	5005

	TaxNumber
1	1000
2	1005

```
exec sp_CreateProducer 9999, 'Cem Tekstil Ürünleri', 'TURKEY', 'KOCAELI', '12314', 1234215
select * from COMPANY
select * from PRODUCER
```

107 %

Results Messages

	TaxNumber	CompanyName	Country	City	PostalCode	BankAccountNumber
1	1000	Kardeşler Tekstil	TURKEY	ISPARTA	12313	5000
2	1005	Kaplan Tuhafiye	TURKEY	KOCAELİ	54213	5001
3	1010	Aras Kargo	TURKEY	İSTANBUL	12312	5002
4	1015	MNG Kargo	TURKEY	İSTANBUL	34521	5003
5	1020	Zara Bağdat Caddesi	TURKEY	İSTANBUL	13321	5004
6	1030	H&M Maltepe	TURKEY	İSTANBUL	33131	5005
7	9999	Cem Tekstil Ürünleri	TURKEY	KOCAELI	12314	1234215

	TaxNumber
1	1000
2	1005
3	9999

8 – sp_CreateShop

Adds shop information to COMPANY table, adds its tax number to SHOP table as FK due to inheritance.


```

CREATE Procedure sp_CreateShop
    @TaxNumber int,
    @CompanyName nvarchar(50),
    @Country nvarchar(25),
    @City nvarchar(25),
    @PostalCode nchar(5),
    @BankAccountNumber nvarchar(50)
As
Begin
    Insert Into COMPANY Values (
        @TaxNumber,
        @CompanyName,
        @Country,
        @City,
        @PostalCode,
        @BankAccountNumber
    )

    Insert Into SHOP Values (@TaxNumber)
End

```

```

select * from COMPANY
select * from SHOP

```

107 %

Results Messages

	TaxNumber	CompanyName	Country	City	PostalCode	BankAccountNumber
1	1000	Kardeşler Tekstil	TURKEY	ISPARTA	12313	5000
2	1005	Kaplan Tuhafiye	TURKEY	KOCAELİ	54213	5001
3	1010	Aras Kargo	TURKEY	İSTANBUL	12312	5002
4	1015	MNG Kargo	TURKEY	İSTANBUL	34521	5003
5	1020	Zara Bağdat Caddesi	TURKEY	İSTANBUL	13321	5004
6	1030	H&M Maltepe	TURKEY	İSTANBUL	33131	5005
7	9999	Cem Tekstil Ürünleri	TURKEY	KOCAELİ	12314	1234215

	TaxNumber
1	1020
2	1030

```
exec sp_CreateShop 99999, 'LCW Hilltown', 'TURKEY', 'ISTANBUL', '34739', 49324
```

```
select * from COMPANY
```

```
select * from SHOP
```

107 %

Results Messages

	TaxNumber	CompanyName	Country	City	PostalCode	BankAccountNumber
1	1000	Kardeşler Tekstil	TURKEY	ISPARTA	12313	5000
2	1005	Kaplan Tuhafiye	TURKEY	KOCAELİ	54213	5001
3	1010	Aras Kargo	TURKEY	İSTANBUL	12312	5002
4	1015	MNG Kargo	TURKEY	İSTANBUL	34521	5003
5	1020	Zara Bağdat Caddesi	TURKEY	İSTANBUL	13321	5004
6	1030	H&M Maltepe	TURKEY	İSTANBUL	33131	5005
7	9999	Cem Tekstil Ürünleri	TURKEY	KOCAELİ	12314	1234215
8	99999	LCW Hilltown	TURKEY	ISTANBUL	34739	49324

	TaxNumber
1	1020
2	1030
3	99999

9 – sp_DeleteEmployee

Deletes an employee record from table EMPLOYEE

Create Procedure sp_DeleteEmployee

@Ssn int

As

Begin

Delete From EMPLOYEE

Where Ssn=@Ssn

End

```
select * from EMPLOYEE
```

107 %

Results Messages

	Ssn	FirstName	LastName	BirthDate	Age	Dno	Gender	Country	City	PostalCode	Salary
1	101	CEM	ANARAL	2001-07-18 00:00:00	20	1	M	TURKEY	İSTANBUL	34123	5000
2	102	HASAN	ŞENYURT	2000-02-15 00:00:00	21	1	M	TURKEY	İSTANBUL	34222	2000
3	103	ALİ	YETİM	2001-09-29 00:00:00	20	1	M	TURKEY	İSTANBUL	34142	2000
4	104	NİMET	ŞAHİN	1970-11-08 00:00:00	51	1	F	TURKEY	İSTANBUL	34414	4000
5	105	ABDULLAH	PEKMEZ	1993-01-03 00:00:00	28	1	M	TURKEY	İSTANBUL	34122	3500
6	201	MELİSA	DURMUŞ	2000-08-24 00:00:00	21	2	F	TURKEY	İSTANBUL	34678	4000
7	202	AYŞE	KESKİN	2000-07-21 00:00:00	21	2	F	TURKEY	İSTANBUL	34567	3000
8	203	ZEYNEP	TURKOĞLU	2000-03-15 00:00:00	21	2	F	TURKEY	İSTANBUL	34789	3000
9	204	BİLGE	DELEN	1989-05-13 00:00:00	32	2	M	TURKEY	İSTANBUL	34321	3700
10	205	KEMALETTİN	KURUR	1967-09-21 00:00:00	54	2	M	TURKEY	İSTANBUL	34131	3000
11	301	EZGİ	MOLA	1998-02-23 00:00:00	23	3	F	TURKEY	İSTANBUL	34125	5000
12	302	ASLAN	ASLANKAYA	1995-05-19 00:00:00	26	3	M	TURKEY	İSTANBUL	34789	4000
13	303	MURAT	BELEK	1978-12-12 00:00:00	43	3	M	TURKEY	İSTANBUL	34567	4000
14	304	BAHAR	YILAR	1999-02-13 00:00:00	22	3	F	TURKEY	İSTANBUL	34133	4000
15	305	SEHEL	CAN	1973-04-28 00:00:00	48	3	F	TURKEY	İSTANBUL	34743	4500
16	401	RECEP	TORLU	1989-03-26 00:00:00	32	4	M	TURKEY	İSTANBUL	34123	7000
17	402	MUHAMMED	OLCAN	1992-06-16 00:00:00	29	4	M	TURKEY	ANKARA	06413	5000
18	403	MUSTAFA	ŞENTÜRK	1996-09-16 00:00:00	25	4	M	TURKEY	İZMİR	35412	5000
19	404	EGE	OĞLAK	1981-02-15 00:00:00	40	4	M	TURKEY	İSTANBUL	34832	5000
20	405	SİMGE	ÖZER	2000-05-12 00:00:00	21	4	F	TURKEY	İSTANBUL	34412	6000
21	501	MEHMET	DEMİR	1972-08-30 00:00:00	49	5	M	TURKEY	İSTANBUL	34987	7000
22	502	FATİH	ŞİMŞEK	1988-03-19 00:00:00	33	5	M	TURKEY	İSTANBUL	34344	6000
23	503	MELEK	TAŞKIN	1972-12-12 00:00:00	49	5	F	TURKEY	İSTANBUL	34423	6000
24	504	ALİHAN	KONUK	2000-03-19 00:00:00	21	5	M	TURKEY	İSTANBUL	34413	4000
25	505	İLAYDA	GÖREN	1997-08-06 00:00:00	24	5	F	TURKEY	İSTANBUL	34739	4000
26	9999	Mustafa	Agaoglu	1980-01-01 00:00:00	41	1	M	TURKEY	İSTANBUL	34000	10999

```
exec sp_DeleteEmployee 9999
select * from EMPLOYEE
```

107 %

Results Messages

	Ssn	FirstName	LastName	BirthDate	Age	Dno	Gender	Country	City	PostalCode	Salary
1	101	CEM	ANARAL	2001-07-18 00:00:00	20	1	M	TURKEY	İSTANBUL	34123	5000
2	102	HASAN	ŞENYURT	2000-02-15 00:00:00	21	1	M	TURKEY	İSTANBUL	34222	2000
3	103	ALİ	YETİM	2001-09-29 00:00:00	20	1	M	TURKEY	İSTANBUL	34142	2000
4	104	NİMET	ŞAHİN	1970-11-08 00:00:00	51	1	F	TURKEY	İSTANBUL	34414	4000
5	105	ABDULLAH	PEKMEZ	1993-01-03 00:00:00	28	1	M	TURKEY	İSTANBUL	34122	3500
6	201	MELİSA	DURMUŞ	2000-08-24 00:00:00	21	2	F	TURKEY	İSTANBUL	34678	4000
7	202	AYŞE	KESKİN	2000-07-21 00:00:00	21	2	F	TURKEY	İSTANBUL	34567	3000
8	203	ZEYNEP	TÜRKOĞLU	2000-03-15 00:00:00	21	2	F	TURKEY	İSTANBUL	34789	3000
9	204	BİLGE	DELEN	1989-05-13 00:00:00	32	2	M	TURKEY	İSTANBUL	34321	3700
10	205	KEMALETTİN	KURUR	1967-09-21 00:00:00	54	2	M	TURKEY	İSTANBUL	34131	3000
11	301	EZGİ	MOLA	1998-02-23 00:00:00	23	3	F	TURKEY	İSTANBUL	34125	5000
12	302	ASLAN	ASLANKAYA	1995-05-19 00:00:00	26	3	M	TURKEY	İSTANBUL	34789	4000
13	303	MURAT	BELEK	1978-12-12 00:00:00	43	3	M	TURKEY	İSTANBUL	34567	4000
14	304	BAHAR	YILAR	1999-02-13 00:00:00	22	3	F	TURKEY	İSTANBUL	34133	4000
15	305	SEHEL	CAN	1973-04-28 00:00:00	48	3	F	TURKEY	İSTANBUL	34743	4500
16	401	RECEP	TORLU	1989-03-26 00:00:00	32	4	M	TURKEY	İSTANBUL	34123	7000
17	402	MUHAMMED	OLCAN	1992-06-16 00:00:00	29	4	M	TURKEY	ANKARA	06413	5000
18	403	MUSTAFA	ŞENTÜRK	1996-09-16 00:00:00	25	4	M	TURKEY	İZMİR	35412	5000
19	404	EGE	OĞLAK	1981-02-15 00:00:00	40	4	M	TURKEY	İSTANBUL	34832	5000
20	405	SİMGE	ÖZER	2000-05-12 00:00:00	21	4	F	TURKEY	İSTANBUL	34412	6000
21	501	MEHMET	DEMİR	1972-08-30 00:00:00	49	5	M	TURKEY	İSTANBUL	34987	7000
22	502	FATİH	ŞİMŞEK	1988-03-19 00:00:00	33	5	M	TURKEY	İSTANBUL	34344	6000
23	503	MELEK	TAŞKIN	1972-12-12 00:00:00	49	5	F	TURKEY	İSTANBUL	34423	6000
24	504	ALİHAN	KONUĞ	2000-03-19 00:00:00	21	5	M	TURKEY	İSTANBUL	34413	4000
25	505	İLAYDA	GÖREN	1997-08-06 00:00:00	24	5	F	TURKEY	İSTANBUL	34739	4000

10 – sp_DeleteProducer

Deletes a producer from both PRODUCER and COMPANY tables

Create Procedure sp_DeleteProducer

@TaxNumber int

As

Begin

Delete From PRODUCER Where TaxNumber=@TaxNumber

Delete From COMPANY Where TaxNumber=@TaxNumber

End

```
select * from COMPANY
select * from PRODUCER
```

107 %

Results Messages

	TaxNumber	CompanyName	Country	City	PostalCode	BankAccountNumber
1	1000	Kardeşler Tekstil	TURKEY	ISPARTA	12313	5000
2	1005	Kaplan Tuhafiye	TURKEY	KOCAELİ	54213	5001
3	1010	Aras Kargo	TURKEY	İSTANBUL	12312	5002
4	1015	MNG Kargo	TURKEY	İSTANBUL	34521	5003
5	1020	Zara Bağdat Caddesi	TURKEY	İSTANBUL	13321	5004
6	1030	H&M Maltepe	TURKEY	İSTANBUL	33131	5005
7	9999	Cem Tekstil Ürünleri	TURKEY	KOCAELİ	12314	1234215
8	99999	LCW Hilltown	TURKEY	İSTANBUL	34739	49324

	TaxNumber
1	1000
2	1005
3	9999

```

exec sp_DeleteProducer 9999
select * from COMPANY
select * from PRODUCER

```

107 %

Results Messages

	TaxNumber	CompanyName	Country	City	PostalCode	BankAccountNumber
1	1000	Kardeşler Tekstil	TURKEY	ISPARTA	12313	5000
2	1005	Kaplan Tuhafiye	TURKEY	KOCAELİ	54213	5001
3	1010	Aras Kargo	TURKEY	İSTANBUL	12312	5002
4	1015	MNG Kargo	TURKEY	İSTANBUL	34521	5003
5	1020	Zara Bağdat Caddesi	TURKEY	İSTANBUL	13321	5004
6	1030	H&M Maltepe	TURKEY	İSTANBUL	33131	5005
7	99999	LCW Hilltown	TURKEY	ISTANBUL	34739	49324

	TaxNumber
1	1000
2	1005

11 – sp_DeleteShop

Deletes a shop from both SHOP and COMPANY tables.

```

Create Procedure sp_DeleteShop
    @TaxNumber int

```

As

Begin

```

Delete From SHOP Where TaxNumber=@TaxNumber
Delete From COMPANY Where TaxNumber=@TaxNumber

```

End

```

select * from COMPANY
select * from SHOP

```

107 %

Results Messages

	TaxNumber	CompanyName	Country	City	PostalCode	BankAccountNumber
1	1000	Kardeşler Tekstil	TURKEY	ISPARTA	12313	5000
2	1005	Kaplan Tuhafiye	TURKEY	KOCAELİ	54213	5001
3	1010	Aras Kargo	TURKEY	İSTANBUL	12312	5002
4	1015	MNG Kargo	TURKEY	İSTANBUL	34521	5003
5	1020	Zara Bağdat Caddesi	TURKEY	İSTANBUL	13321	5004
6	1030	H&M Maltepe	TURKEY	İSTANBUL	33131	5005
7	99999	LCW Hilltown	TURKEY	ISTANBUL	34739	49324

	TaxNumber
1	1020
2	1030
3	99999

```
exec sp_DeleteShop 99999
select * from COMPANY
select * from SHOP
```

107 %

Results Messages

	TaxNumber	CompanyName	Country	City	PostalCode	BankAccountNumber
1	1000	Kardeşler Tekstil	TURKEY	ISPARTA	12313	5000
2	1005	Kaplan Tuhafiye	TURKEY	KOCAELİ	54213	5001
3	1010	Aras Kargo	TURKEY	İSTANBUL	12312	5002
4	1015	MNG Kargo	TURKEY	İSTANBUL	34521	5003
5	1020	Zara Bağdat Caddesi	TURKEY	İSTANBUL	13321	5004
6	1030	H&M Maltepe	TURKEY	İSTANBUL	33131	5005

	TaxNumber
1	1020
2	1030

12 – sp_EmptyManager

Sets given departments manager to null

```
Create Procedure sp_EmptyManager(
    @Dno int
)
As
Begin
    Update DEPARTMENT
    Set ManagerSsn=NULL
    Where Dno=@Dno
End
```

```
select * from DEPARTMENT
```

107 %

Results Messages

	Dno	Name	ManagerSsn	NoOfWorkers
1	1	IT	103	5
2	2	MARKETING	201	5
3	3	SALES	301	5
4	4	OPERATION	401	5
5	5	QUALITY CONTROL	501	5

```
exec sp_EmptyManager 5
select * from DEPARTMENT
```

107 %				
Results		Messages		
	Dno	Name	ManagerSsn	NoOfWorkers
1	1	IT	103	5
2	2	MARKETING	201	5
3	3	SALES	301	5
4	4	OPERATION	401	5
5	5	QUALITY CONTROL	NULL	5

13 – sp_SetManager

Assigns manager to DEPARTMENT table.

```
Create Procedure sp_SetManager(
    @Ssn int,
    @Dno int
)
As
Begin
    Update DEPARTMENT
    Set ManagerSsn=@Ssn
    Where Dno=@Dno
End
```

```
select * from DEPARTMENT
```

107 %				
Results		Messages		
	Dno	Name	ManagerSsn	NoOfWorkers
1	1	IT	103	5
2	2	MARKETING	201	5
3	3	SALES	301	5
4	4	OPERATION	401	5
5	5	QUALITY CONTROL	NULL	5

```
exec sp_SetManager 101, 5
select * from DEPARTMENT
```

107 %				
Results		Messages		
	Dno	Name	ManagerSsn	NoOfWorkers
1	1	IT	103	5
2	2	MARKETING	201	5
3	3	SALES	301	5
4	4	OPERATION	401	5
5	5	QUALITY CONTROL	101	5

14 – sp_UpdateTotalPrice

Updates total price of every entry in SHIPMENT. Total price is calculated using the clothes that shipment contains.

```
Create Procedure sp_UpdateTotalPrice
```

```
As
```

```
Begin
```

```
Update s
```

```
Set s.TotalPrice= new.TotalPrice
```

```
From SHIPMENT s inner join (Select s.ShipmentID,sum(c.SoldPrice) as TotalPrice
```

```
From CLOTHING c inner join SHIPMENT s on c.ShipmentID=s.ShipmentID
```

```
Group By s.ShipmentID) new on s.ShipmentID=new.ShipmentID
```

```
End
```

```
select * from SHIPMENT
```

107 %

	ShipmentID	ShipmentDate	InventoryCode	ProducerTaxNumber	LogisticsTaxNumber	ShopTaxNumber	TotalPrice
1	9000	2021-03-03 00:00:00	1	1000	1010	NULL	585
2	9001	2021-03-04 00:00:00	2	NULL	1015	1020	1280
3	9002	2021-06-20 00:00:00	3	1005	1010	NULL	570
4	9003	2021-12-09 00:00:00	1	NULL	1015	1030	670
5	9004	2021-09-11 00:00:00	3	1000	1010	NULL	570
6	9005	2021-12-23 14:48:00	2	1000	1015	NULL	450
7	9006	2021-12-23 14:50:00	1	NULL	1010	1020	NULL
8	9007	2021-12-23 15:56:00	3	1000	1010	NULL	NULL
9	9008	2021-12-23 16:04:00	3	NULL	1015	1020	NULL

```
sp_UpdateTotalPrice
```

```
select * from SHIPMENT
```

ustel

107 %

	ShipmentID	ShipmentDate	InventoryCode	ProducerTaxNumber	LogisticsTaxNumber	ShopTaxNumber	TotalPrice
1	9000	2021-03-03 00:00:00	1	1000	1010	NULL	585
2	9001	2021-03-04 00:00:00	2	NULL	1015	1020	1280
3	9002	2021-06-20 00:00:00	3	1005	1010	NULL	570
4	9003	2021-12-09 00:00:00	1	NULL	1015	1030	670
5	9004	2021-09-11 00:00:00	3	1000	1010	NULL	570
6	9005	2021-12-23 14:48:00	2	1000	1015	NULL	450
7	9006	2021-12-23 14:50:00	1	NULL	1010	1020	NULL
8	9007	2021-12-23 15:56:00	3	1000	1010	NULL	NULL
9	9008	2021-12-23 16:04:00	3	NULL	1015	1020	150