



MARMARA UNIVERSITY COMPUTER ENGINEERING DEPARTMENT

Step 3: Logical database design and mapping & Physical design and database implementation + (Req. ana. & Concep. db. des.)

Due To: 30/12/2022

Lecturer: Associate Professor Mustafa AĞAOĞLU

Course: CSE 3055 Database Systems

Name Surname	Student ID
Berkkan Rençber	150119011-GrRep
Murat Özcan	150119008
Faruk Akdemir	150119012

PROJECT DESCRIPTION

In this project we implemented a database for a data center company and present it to the user with a web interface. IstanbulDC is a big data center company which is provide a huge capacity servers to customers whose needs are to collect and store large amount of data. We created some tables which that'll keep the information of employees, customers, departments, servers, cabins, orders, network switchs and bills. After that, the user can use the web page to display this information. For instance, user can retrieve salary of a desired employee,add a new employee/customer, display the rented servers,etc. We tried to provide many features to the user in this database design.

SCOPE

What is included ?

- Personal and business information about employees in the company is stored.
- Personal and user information of customers ordering from the company is stored.
- It keeps the features of the servers owned by the company and whether these servers are rented. If it is rented, it adds customer and order information to the server.
- The cabin information where the servers in the company are located is stored.
- The network switch information in each cabin is stored.
- After each customer orders, invoice information is added and the information of the ordered server is updated.
- The information about departments in the company and employees working there are stored.
- Inserting new customers to the company and assigning a customer representative to that customer, updating customer information.
- Inserting a new server to the company and deleting it if the server is not owned by anyone.

What is excluded ?

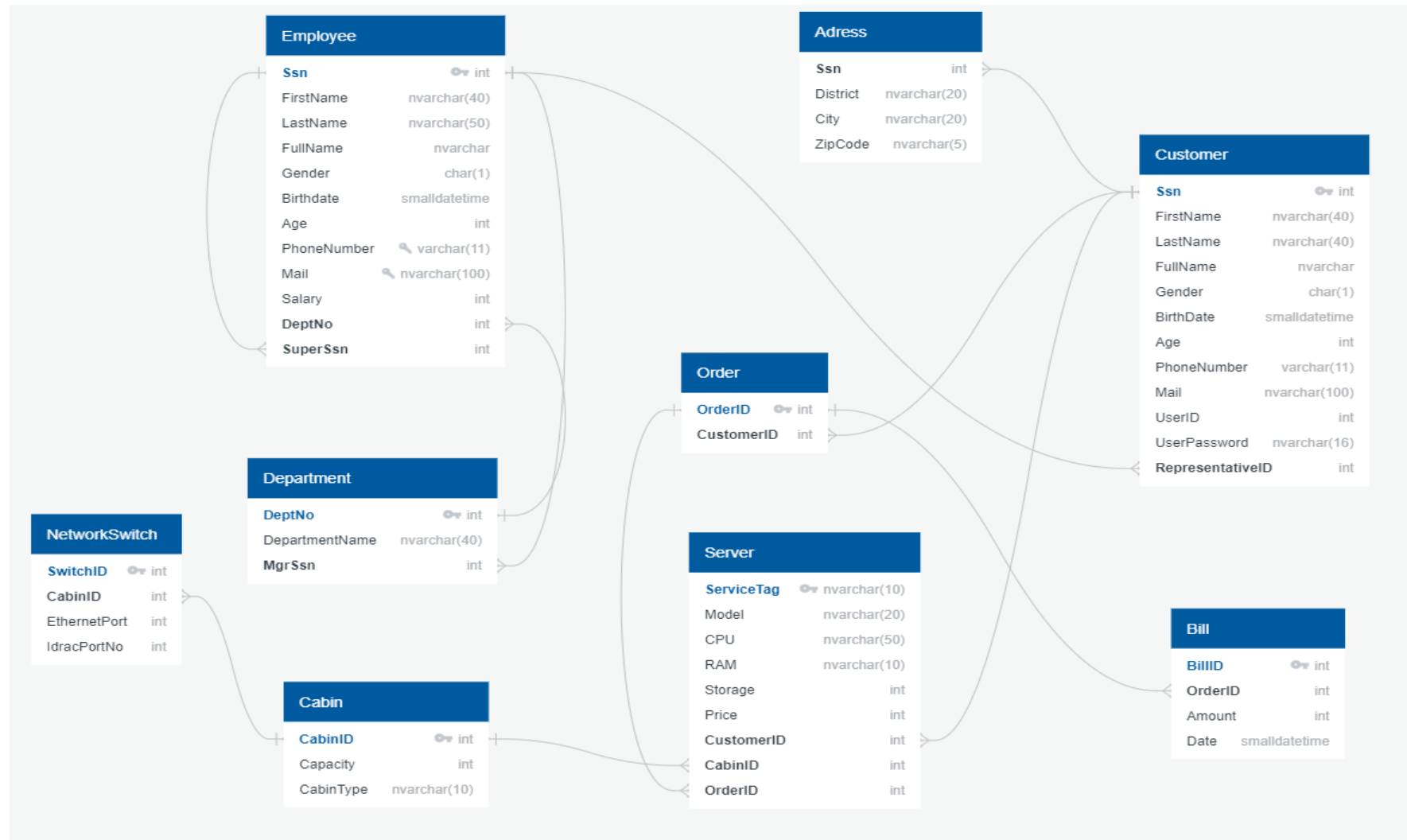
- Inserting new employees, departments to the company or deleting an existing one.
- Creating monthly bill at the orders is excluded. All bills are one-time only.
- An order does not contain more than one server.

DATA AND REQUIREMENTS ANALYSIS

We determined the most appropriate data types used in the database. We created the default values for Salary in the Employee table and CabinType in the Cabin table. We set the unique values for PhoneNumber and Mail in the Customer and Employee tables. We used Identity for Ssn in the Employee table, UserID in the Customer table, DeptNo in the Department table, and EthernetPort in the NetworksSwitch table. We used the indexes on all foreign keys and the UserID in the Customer table. We also applied the computed variables for Employee and Age in the Customer table and Amount in the Bill table.

- An employee working in the company can represent a maximum of 5 customers. We did this in the trigger that runs when a new client is added.
- Only one priority mail address of an employee and a customer is kept in the system.
- Each worker must have exactly 1 supervisor.
- Every department should have a manager.
- The company defines an identity for the customer and the customer uses it to enter the system, and also sets a minimum 8-character password himself/herself.
- When a new customer is defined in the system, a representative is assigned to the customer.
- When a customer comes to the system, the cheapest hardware for the customer's needs is recommended. When the customer selects the hardware, namely the server, it goes to the order section and then an bill is created. After this step, the server is taken into the cabin, the ethernet port, idrac port and software of the server are set and presented to the customer.
- There are 3 types of cabinets, quarter, half and full, and each cabinet has a network switch. Quarter rack 10, half rack 20, and full rack 40 servers.
- The company uses a minimum of 128GB of storage on a server and does not make the customer choose less.
- Bills are sent to the customer monthly via mail. The invoice amount is kept in the database as Amount.
- More than one employee cannot serve the same client's server at the same time, and vice versa.
- In the company, server enclosures are provided by the supplier. The servers, whose hardware is determined by the company, are kept ready in the cabinets. If the server is to be adjusted on the order of the customer, the server hardware outside the cabinet is adjusted and integrated into the chassis and taken into the cabinet in this way.
- If there is a hardware inadequacy regarding the server in the use of the customer, a hardware update is made.

DIAGRAM OF DATABASE



TABLES

EMPLOYEE	
Definition: This table stores the information of the workers in the company.	
Check Constraints: -	
Triggers: -	
Primary Keys: Ssn	
Foreign Keys: DeptNo,SuperSsn	
Identity: Ssn	
Defaults: Salary	
Uniques: PhoneNumber,Mail	
Indexes: DeptNo,SuperSsn	
Computed Columns: Age	
Column Name	Data Type
Ssn	int
FirstName	nvarchar(40)
LastName	nvarchar(50)
FullName	
Gender	char(1)
BirthDate	smalldatetime
Age	
PhoneNumber	varchar(11)
Mail	nvarchar(100)
Salary	int
DeptNo	int
SuperSsn	int

CUSTOMER	
Definition: This table stores personal information about customers.	
Check Constraints: UserPassword must be longer than 8 characters.	
Triggers: It updates representativeID when a customer is inserted.	
Primary Keys: Ssn	
Foreign Keys: RepresentativeID	
Identity: UserID	
Defaults: -	
Uniques: PhoneNumber,Mail	
Indexes: Ssn, RepresentativeID	
Computed Columns: Age	
Column Name	Data Type
Ssn	int
FirstName	nvarchar(40)
LastName	nvarchar(40)
FullName	
Gender	char(1)
BirthDate	smalldatetime
Age	
PhoneNumber	varchar(11)
Mail	nvarchar(100)
UserID	int
UserPassword	nvarchar(16)
RepresentativeID	int

ADDRESS	
Definition: This table stores the information of the workers and customers addresses in the company.	
Check Constraints: -	
Triggers: -	
Primary Keys: -	
Foreign Keys: Ssn	
Identity: -	
Defaults: -	
Uniques: -	
Indexes: Ssn	
Computed Columns: -	
Column Name	Data Type
Ssn	int
District	nvarchar(20)
City	nvarchar(20)
ZipCode	nvarchar(5)
Address	

ORDER	
Definition: This table stores the customer's orders.	
Check Constraints: -	
Triggers: It inserts to Bill Table value of inserted order.	
Primary Keys: OrderID	
Foreign Keys: CustomerID	
Identity: -	
Defaults: -	
Uniques: -	
Indexes: CustomerID	
Computed Columns: -	
Column Name	Data Type
OrderID	int
CustomerID	int

DEPARTMENT	
Definition: This table stores the information of the department in the company.	
Check Constraints: -	
Triggers: -	
Primary Keys: DeptNo	
Foreign Keys: MgrSsn	
Identity: DeptNo	
Defaults: -	
Uniques: -	
Indexes: MgrSsn	
Computed Columns: -	
Column Name	Data Type
DeptNo	int
DepartmentName	nvarchar(40)
MgrSsn	int

BILL	
Definition: This table stores detail information about orders of customer.	
Check Constraints: -	
Triggers: -	
Primary Keys: BillID	
Foreign Keys: OrderID	
Identity: -	
Defaults: -	
Uniques: -	
Indexes: OrderID	
Computed Columns: Amount	
Column Name	Data Type
BillID	int
OrderID	int
Amount	int
Date	smalldatetime

SERVER	
Definition: This table stores the information of Servers.	
Check Constraints: Storage of Server has to greater than or equal 128 GB	
Triggers: -	
Primary Keys: ServiceTag	
Foreign Keys: CustomerID,CabinID,OrderID	
Identity: -	
Defaults: -	
Uniques: -	
Indexes: CustomerID,CabinID,OrderID	
Computed Columns: -	
Column Name	Data Type
ServiceTag	nvarchar(10)
Model	nvarchar(20)
CPU	nvarchar(50)
RAM	nvarchar(10)
Storage	int
Price	int
CustomerID	int
CabinID	int
OrderID	int

CABIN	
Definition: This table stores the information of Cabins.	
Check Constraints: -	
Triggers: -	
Primary Keys: CabinID	
Foreign Keys: -	
Identity: -	
Defaults: CabinType is Quarter	
Uniques: -	
Indexes: -	
Computed Columns: -	
Column Name	Data Type
CabinID	int
Capacity	int
CabinType	nvarchar(10)

NETWORKSWITCH	
Definition: This table stores the information of Network Switches.	
Check Constraints: -	
Triggers: -	
Primary Keys: SwitchID	
Foreign Keys: CabinID	
Identity: EthernetPort	
Defaults: -	
Uniques: -	
Indexes: CabinID	
Computed Columns: -	
Column Name	Data Type
SwitchID	int
CabinID	int
EthernetPort	int
IdracPortNo	int

VIEWS

1) CabinNoOfEmptyRail

Definition: This view lists all available booths in the company.

```
CREATE VIEW CabinNoOfEmptyRail
AS
SELECT *
FROM
(
    SELECT c.CabinID, c.CabinType, c.Capacity - COUNT(*) AS EmptyRails
    FROM Cabin c
    LEFT OUTER JOIN Server s ON c.CabinID = s.CabinID
    GROUP BY c.CabinID, c.CabinType, c.Capacity
) nt
WHERE nt.EmptyRails > 0
ORDER BY nt.EmptyRails DESC, nt.CabinID DESC OFFSET 0 ROWS
GO
```

```
SELECT * FROM CabinNoOfEmptyRail
```

	CabinID	CabinType	EmptyRails
1	12	Full	22
2	11	Full	19
3	10	Full	15
4	6	Half	13
5	7	Half	10
6	4	Half	9
7	9	Full	8
8	5	Half	8
9	2	Quarter	7
10	1	Quarter	5
11	3	Quarter	4

2) NotRentedServer

Definition: This view lists all not rented servers in the company. This view does not store CustomerID and OrderID in Server Table because of not rented.

```
CREATE VIEW NotRentedServer
AS
SELECT s.ServiceTag, s.Model, s.CPU, s.RAM, s.Storage, s.Price, s.CabinID
FROM Server s
WHERE s.CustomerID IS NULL

SELECT * FROM NotRentedServer
```

	ServiceTag	Model	CPU	RAM	Storage	Price	CabinID
1	MP2560Y	R750	INTEL I3	256	256	3902	9
2	MS9K350	R240	AMD EPYC	64	1024	9497	9
3	MUBV68N	R550	INTEL XEON E-2300	128	5120	6824	9
4	MV4H3D	T640	INTEL CELERON	32	20480	1004	9
5	MY1217W	T640	INTEL I3	64	512	5368	9
6	MY4T61S	R750	INTEL I3	256	14336	5873	9
7	MY4T61B	XR12	AMD EPYC	128	1024	4521	9
8	MY9V11T	R750	INTEL PENTIUM	128	2048	2664	9
9	NB8L71E	R6515	AMD EPYC	32	4096	3141	10
10	NB9C23N	T150	INTEL XEON	64	15360	9304	10
11	ND7T68Q	XR11	INTEL CELERON	64	512	6260	10
12	NE5E77U	R340	INTEL PENTIUM	64	8192	1435	10
13	NK3Q13L	T440	INTEL I3	256	14336	7825	10
14	NO7H63N	R240	INTEL I3	256	14336	8693	10
15	NP1L64Y	R440	INTEL XEON	256	12288	8198	10
16	NP293L	R340	INTEL CELERON	256	8192	2856	10
17	NO7P42N	R240	INTEL XEON E-2300	32	20480	6162	10
18	NR3R36D	R210H	INTEL XEON E-2300	128	512	5105	10
19	NT2D41P	R250	AMD EPYC	32	12288	8647	10
20	NT9L32W	XR12	INTEL PENTIUM	32	2048	1730	10
21	NZ5T56K	R440	INTEL I3	256	256	4650	10
22	OA7N33N	R550	INTEL XEON	128	14336	7665	10
23	OF8Y97Z	R450	INTEL XEON	32	20480	3588	10
24	OH8K31S	T440	INTEL I3	256	8192	7012	10
25	OMSJ41C	R640	AMD EPYC	256	5120	5291	10

Query executed successfully.

MURAT/MRTZCN (14.0 RTM) | MURAT/mrtzc (54) | IstanbulDC | 00:00:00 | 72 rows

3) NumberOfEmployeeInDepartments

Definition: This view lists the number of employees in each departments.

```
Create View NumberOfEmployeeInDepartments
As
Select d.DeptNo,d.DepartmentName,d.MgrSsn, count(*) NoOfEmployee
From Department d
    inner join Employee e on d.DeptNo=e.DeptNo
Group By d.DeptNo,d.DepartmentName, d.MgrSsn

Select * From NumberOfEmployeeInDepartments
```

123 %

Results Messages

	DeptNo	DepartmentName	MgrSsn	NoOfEmployee
1	101	Central Office	2	2
2	201	Marketing	4	5
3	301	Human Resources	17	3
4	401	Sales	20	30

4) ExpensiveServer

Definition: This view brings the servers whose price greater than average price of whole servers.

```
Create View [ExpensiveServer]
As
Select *
From Server s
Where s.Price > (Select avg(s.Price) From Server s );
Go

Select* From [ExpensiveServer]
```

100 %

Results Messages

	ServiceTag	Model	CPU	RAM	Storage	Price	CustomerID	CabinID	OrderID
1	AQ4A69R	R7515	INTEL PENTIUM	32	4096	8568	705	0	4
2	AZ9Q12J	R6515	INTEL CELERON	32	20480	6077	705	0	6
3	BH5I79O	R540	INTEL CELERON	64	1024	8360	640	0	10
4	BI3U69Y	R450	INTEL i3	256	14336	9988	640	1	11
5	BJ6N35J	R7515	INTEL PENTIUM	256	20480	7438	780	1	12
6	BK9A88E	R7515	AMD EPYC	256	4096	9318	774	1	14
7	BM5H42R	R240	INTEL PENTIUM	32	20480	5803	874	2	17
8	BS2Z37V	R550	INTEL CELERON	128	8192	5944	983	2	18
9	CB8I65G	T150	AMD EPYC	64	2048	7171	843	3	24
10	CC8Y45D	R540	INTEL i3	128	12288	8179	758	4	25
11	CD4A28R	T150	INTEL XEON E-2300	256	256	5891	758	4	26
12	CG4B53F	R450	AMD EPYC	32	12288	7707	842	4	27
13	CG7N52I	R540	AMD EPYC	128	1024	8964	805	4	28
14	CL5F11M	T550	AMD EPYC	64	256	6540	618	4	29
15	CM3T46Q	R250	INTEL PENTIUM	64	1024	7740	851	4	30
16	CN7C27H	R440	INTEL CELERON	32	512	7212	846	4	31
17	DF6A29V	XR11	INTEL CELERON	128	14336	9606	722	4	35
18	EA8M69L	R350	INTEL XEON	64	512	9142	741	5	38
19	EK5B53H	T150	INTEL i3	64	1024	7447	766	5	42
20	EK7N41H	R210	INTEL CELERON	32	1024	9263	676	5	43
21	EN7S65G	R940	INTEL PENTIUM	128	14336	8041	708	5	46
22	EY4G79D	T550	INTEL XEON E-2300	64	5120	6167	585	6	50
23	FG1G63D	R6515	INTEL i3	64	15360	8868	700	6	52

5) TotalAmountCustomer

Definition: This view retrieves the total expense of each customer.

```
Create View [TotalAmountCustomer]
As
Select c.Ssn,c.FullName,sum(s.Price) as TotalAmountOfOrders
From Customer c inner join Orderr o on c.Ssn=o.CustomerID
    inner join Server s on o.CustomerID=s.CustomerID and o.OrderID=s.OrderID
Group by c.Ssn,c.FullName
Go
```

100 %

Results Messages

	Ssn	FullName	TotalAmountOfOrders
1	504	Tahir Eracar	2984
2	511	Somer Sevinçhan	5085
3	513	Kadir Özpek	4946
4	519	Hüma Mosso	6526
5	525	Ertuğ Balıkcı	11849
6	530	Mahir Kapudere	1486
7	534	Abdülkadir Çirkin	6086
8	537	Cemre Özşimşek	2752
9	541	Burhan Bağcı	5094
10	546	Özge Evsen	8444
11	548	Tayfun Akbay	2065
12	555	Vural Çintay	9151
13	564	Burak Erçin	8731
14	567	Sadık Kartal	2111
15	583	Faruk Söylem	7285
16	584	Tolgay Tutak	5422
17	585	Can İboğlu	7682
18	589	Ayberk Subaşı	1143
19	605	Veli Kenesari	5422

Query executed successfully.

FFARUKK\MSSQLSERVER04 (15.0... FFARUKK\

TRIGGERS

1) insert_Representative

Definition: This trigger assigns a customer representative to that customer when a new customer is inserted into the company.

```

Create Trigger insert_Representative on Customer
After Insert As
Begin
    declare
    @empID int

    Select @empID=e.Ssn
    From Employee e
    Where e.Ssn in (Select e.Ssn
                    From Employee e
                    inner join Customer c on e.Ssn=c.RepresentativeID
                    Group By e.Ssn
                    Having count(*) < 5)

    Update Customer Set RepresentativeID = @empID Where RepresentativeID is null
End

```

Before Executing: Employees with id 119 have been assigned to 4 customers as representatives.

Ssn	FirstName	LastName	FullName	Gender	BirthDate	Age	PhoneNum...	Mail	UserID	UserPasswo...	RepresentativeID
884	Selen	Keklikiçi	Selen Keklikiçi	K	1956-11-04 ...	66	05931328263	at@ac.wf	1020	YE4N44G1J4	115
885	Enver	Ariyan	Enver Ariyan	E	1976-05-09 ...	46	05859476560	Aliquam@h...	1030	JR5I88B2Z5	115
889	Berkan	Özgüven	Berkan Özgüven	E	1989-01-02 ...	33	05555328038	gravid@nu...	1082	JD6D57Y4L2	115
890	Mücahit	Yapar	Mücahit Yap...	E	1965-09-11 ...	57	05852921081	sem@Viva...	1008	GR7B58T4G7	115
895	Gonca	Ekim	Gonca Ekim	K	1987-03-28 ...	35	05973805560	dapibus@n...	1074	QW7E3BQ1...	116
899	Sami	Çetin	Sami Çetin	E	1968-08-19 ...	54	05831027689	adipidisscin...	1118	CKS578A3G4	116
908	Yağız	Baytak	Yağız Baytak	E	1975-07-17 ...	47	05268873163	odio@ac.pn	1044	LN2A44W4...	116
911	Engin	Parçalı	Engin Parçalı	E	1991-05-17 ...	31	05997593267	at@leo.ht	1094	SJ9W79L7K5	116
915	Özbay	Kaçar	Özbay Kaçar	E	1957-10-09 ...	65	05632923193	faucibus@ia...	1046	ST6C98D7A6	117
918	Şadi	Şanal	Şadi Şanal	E	1973-09-07 ...	49	05029097851	Nullam@sc...	1011	IV7D57P1M1	117
920	Süeda	Apohan	Süeda Apoh...	K	1959-06-16 ...	63	05942735378	velit@molli...	1062	CF2O63D6O8	117
927	Cemil	Çilğın	Cemil Çilğın	E	1963-02-13 ...	59	05661033348	mollis@Do...	1100	YI9G44C1Q9	118
928	Rıza	Doğan	Rıza Doğan	E	1973-05-30 ...	49	05858138281	diam@port...	1070	UL3J95O1G8	118
932	Arda	Berктаş	Arda Berктаş	E	1967-03-15 ...	55	05743245187	eger@laore...	1009	VU4P46O1K5	119
941	Bilal	Delen	Bilal Delen	E	1977-09-05 ...	45	05952904550	nunc@port...	1102	BW6W81K1...	119
946	Selim	Andaç	Selim Andaç	E	2000-07-15 ...	22	05464066806	consectetur...	1042	PM2E32B4U5	119
948	Halime	Altınkran	Halime Altı...	K	1995-07-02 ...	27	05165772987	Pellentesque...	1096	VY2Y42R1G5	119
952	Özqe	Kartal	Özqe Kartal	K	1952-10-10 ...	70	05499746384	at@arcu.pr	1024	FP4M25I5J7	37

Before Inserting First Customer:

```
INSERT INTO Customer (Ssn, FirstName, LastName, Gender, BirthDate, Mail, PhoneNumber, UserPassword)
VALUES ('3000', 'Murat', 'Özcan', 'E', '1968-12-19', 'Phdsaddudsas@lobortis.cc', '05300068850', 'ZHAdS6DSA51');
```

After Inserting First Customer:

[illegible]

Before Inserting Second Customer:

```
INSERT INTO Customer (Ssn, FirstName, LastName, Gender, BirthDate, Mail, PhoneNumber, UserPassword)
VALUES ('3001','Berkkan','Rençber','E','1968-12-19','Phudasddsas@lobortis.cc','05311168850','AHAdS6DSA51');
```

After Inserting First Customer: Since 119 workers with id were appointed as representatives to 5 customers, 118 workers with id were assigned to the newly added customer.

997	Oğuzhan	Buluş	Oğuzhan Bu...	E	1966-12-04 ...	56	05447872076	amet@eget...	1035	KU4C82A9A9	110
3000	Murat	Özcan	Murat Özcan	E	1968-12-19 ...	54	05300068850	Phdsadduds...	1132	ZHAdS6DS...	119
3001	Berkkan	Rençber	Berkkan Ren...	E	1968-12-19 ...	54	05311168850	Phudasddsas...	1133	AHAdS6DS...	118

2) insert_Bill

Definition: This trigger creates an invoice when a new order arrives.

```
CREATE TRIGGER insert_Bill
ON Orderr
AFTER INSERT
AS
BEGIN
    DECLARE @lastBillID int;
    DECLARE @totalAmount int;
    DECLARE @lastOrderID int;

    SELECT @lastBillID=MAX(BillID)
    FROM Bill

    SELECT @lastOrderID=i.OrderID,@totalAmount=sum(s.Price)
    FROM inserted i
    INNER JOIN Server s ON i.CustomerID=s.CustomerID
    GROUP BY i.OrderID

    INSERT INTO Bill(BillID,OrderID,Amount,Date) VALUES (@lastBillID+10,@lastOrderID,@totalAmount,GETDATE())
END
```

Before Executing Of Bill:

	BillID	OrderID	Amount	Date
	48512	37	1117	2015-03-14 00:0...
	48932	40	4687	2010-05-10 00:0...
	49158	85	1232	2015-06-23 00:0...
	49920	113	2976	2012-02-20 00:0...
*	NULL	NULL	NULL	NULL

After Executing:

```
Insert Into Orderr(CustomerID) Values(504)
```

0 %

Messages

(1 row affected)

(1 row affected)

Completion time: 2022-12-28T22:54:11.5049038+03:00

BillID	OrderID	Amount	Date
48932	40	4687	2010-05-10 00:0...
49158	85	1232	2015-06-23 00:0...
49920	113	2976	2012-02-20 00:0...
49930	534	6966	2022-12-28 22:5...
NULL	NULL	NULL	NULL

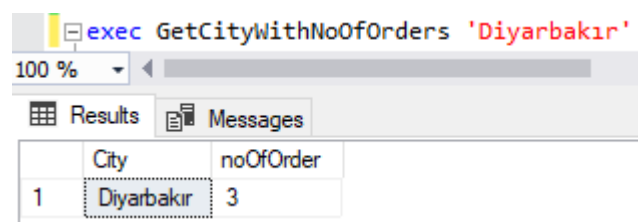
STORE PROCEDURES

1) GetCityWithNoOfOrders

Definition: This stored procedure lists the number of orders from that city by taking the entered city parameter as a condition.

```
Create Procedure GetCityWithNoOfOrders(@City as nvarchar(20))
As
Begin
    Select a.City, Count(*) as noOfOrder
    From Customer c
        inner join Orderr o on c.Ssn=o.CustomerID
        right outer join Address a on a.Ssn=c.Ssn
    Where a.City=@City
    Group By a.City
End
Go
```

After Executing:



```
exec GetCityWithNoOfOrders 'Diyarbakır'
```

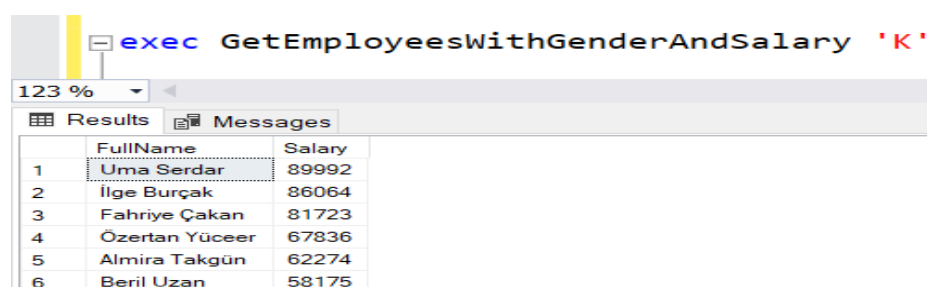
	City	noOfOrder
1	Diyarbakır	3

2) GetEmployeesWithGenderAndSalary

Definition: This stored procedure lists fullname and salary of employees by gender.

```
Create Procedure GetEmployeesWithGenderAndSalary
@Gender char(1)
as
    Select e.FullName, e.Salary
    From Employee e
    Where e.Gender=@Gender and e.Salary >= (Select avg(emp.Salary)
                                                From Employee emp)
    Order By e.Salary desc
```

After Executing:



```
exec GetEmployeesWithGenderAndSalary 'K'
```

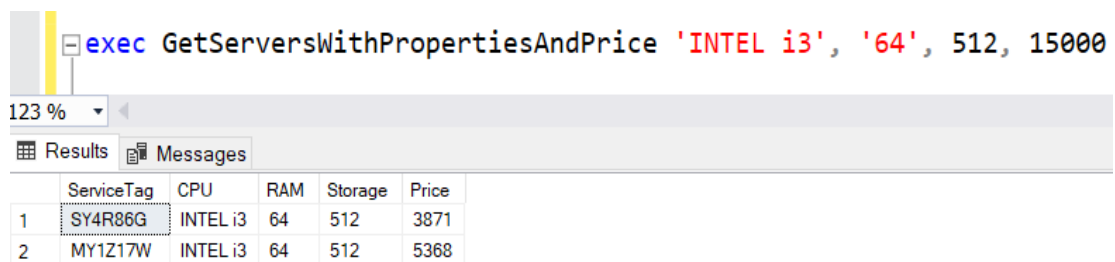
	FullName	Salary
1	Uma Serdar	89992
2	Ilge Burçak	86064
3	Fahriye Çakan	81723
4	Özertan Yüceer	67836
5	Almira Takgün	62274
6	Beril Uzan	58175

3) GetServersWithPropertiesAndPrice

Definition: This stored procedure lists the servers with the properties selected by the user.

```
Create Procedure GetServersWithPropertiesAndPrice
@CPU nvarchar(50),
@RAM nvarchar(10),
@Storage int,
@Price int
as
Select s.ServiceTag, s.CPU, s.RAM, s.Storage, s.Price
From Server s
Where s.ServiceTag in (Select nrs.ServiceTag
                        From NotRentedServer nrs)
and s.CPU = @CPU
and s.RAM = @RAM
and s.Storage = @Storage
and s.Price <= @Price
Order By s.Price asc
```

After Executing:



exec GetServersWithPropertiesAndPrice 'INTEL i3', '64', 512, 15000

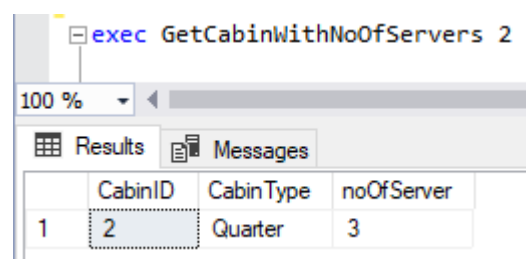
	ServiceTag	CPU	RAM	Storage	Price
1	SY4R86G	INTEL i3	64	512	3871
2	MY1Z17W	INTEL i3	64	512	5368

4) GetCabinWithNoOfServers

Definition: This stored procedure lists the number of Servers in that cabin by taking the CabinID parameter as a condition.

```
Create Procedure GetCabinWithNoOfServers(@CabinID as int)
As
Begin
Select c.CabinID,c.CabinType,COUNT(*) as noOfServer
From Cabin c
left outer join Server s on c.CabinID=s.CabinID
Where c.CabinID=@CabinID
Group By c.CabinID,c.CabinType
End
Go
```

After Executing:



exec GetCabinWithNoOfServers 2

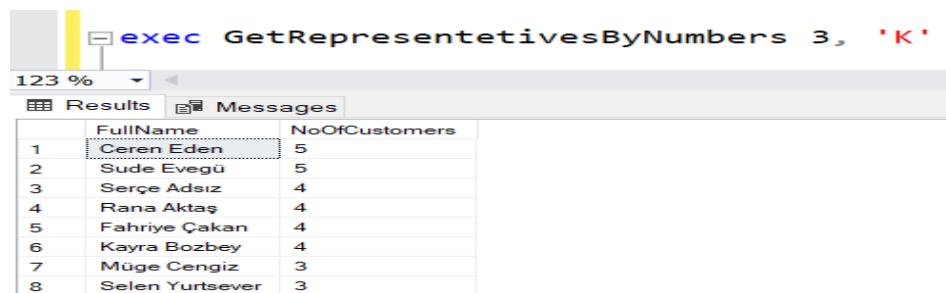
	CabinID	CabinType	noOfServer
1	2	Quarter	3

5) GetRepresentativesByNumbers

Definition: This stored procedure lists employees who represent more than a certain number of customers by gender.

```
Create Procedure GetRepresentativesByNumbers
@number int,
@Gender char(1)
as
    Select e.FullName, count(*) NoOfCustomers
    From Employee e
        inner join Customer c on e.Ssn=c.RepresentativeID
    Where e.Gender = @Gender
    Group By e.FullName
    Having count(*) >= @number
    Order By count(*) desc
```

After Executing:



exec GetRepresentativesByNumbers 3, 'K'

	FullName	NoOfCustomers
1	Ceren Eden	5
2	Sude Evegü	5
3	Serçe Adsız	4
4	Rana Aktaş	4
5	Fahriye Çakan	4
6	Kayra Bozbey	4
7	Müge Cengiz	3
8	Selen Yurtsever	3

6) GetRetireYear

Definition: This stored procedure calculates how many years are left for employees to retire.

```
Create Procedure GetRetireYear
@Ssn int
as
declare
@Age int,
@limityear int = 55,
@fullName nvarchar(40)

Select @Age=Age, @fullName=FullName
From Employee
Where Ssn=@Ssn

if(@limityear <= @Age)
    print @fullName + ' can retire'
else
    print @fullName + ' can not retire. Remaining years ' + convert(varchar, @limityear-@Age)
```

After Executing:

```
exec GetRetireYear 102
exec GetRetireYear 103
```

Messages

Ceren Eden can retire
Kayra Bozbey can not retire. Remaining years 9

Completion time: 2022-12-28T21:19:22.7115761+03:00

7) CheckCabinStatus

Definition: This stored procedure takes the CabinID parameter as condition and prints a message about whether a new Server can be placed in that cabin.

```
CREATE PROCEDURE CheckCabinStatus(@CabinID as int)
AS
BEGIN
    DECLARE @noOfEmptyRails int;
    DECLARE @tempCabinID int;
    Select @tempCabinID=c.CabinID,@noOfEmptyRails=c.EmptyRails
    From CabinNoOfEmptyRail c
    Where c.CabinID=@CabinID

    IF @tempCabinID is not null
    BEGIN
        PRINT 'There are ' + CAST(@noOfEmptyRails as nvarchar(10)) + ' empty rails in cabin with CabinID ' + CAST(@tempCabinID as nvarchar(10))+'. '
    END
    ELSE
    BEGIN
        PRINT 'There are no empty rail in cabin with CabinID ' + CAST(@CabinID as nvarchar(10))+'. '
    END
END
GO
```

After Executing:

```
exec CheckCabinStatus 12
```

100 %

Messages

There are 22 empty rails in cabin with CabinID 12.

Completion time: 2022-12-28T21:16:11.2807563+03:00

```
exec CheckCabinStatus 8
```

100 %

Messages

There are no empty rail in cabin with CabinID 8.

Completion time: 2022-12-28T21:17:08.8473043+03:00

8) RentsServer

Definition: This stored procedure rents the server to the customer by taking the customer's id and the server's tag from the customer.

```
Create Procedure RentsServer
@Ssn int,
@ServiceTag nvarchar(10)
as
declare
@orderID int,
@price int;

Update Server Set CustomerID = @Ssn Where ServiceTag=@ServiceTag

Insert Into Orderr(CustomerID) values(@Ssn);

Select @orderID=OrderID From Orderr

Update Server Set OrderID = @orderID Where ServiceTag=@ServiceTag
```

Before Executing: (In Server Table)

ServiceTag	Model	CPU	RAM	Storage	Price	CustomerID	CabinID	OrderID
MP2S69Y	R750	INTEL i3	256	256	3982	NULL	9	NULL

After Executing:

```
exec RentsServer 519, 'MP2S69Y'
```

Messages

```
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
Completion time: 2022-12-28T21:33:37.4471036+03:00
```

In Server Table:

ServiceTag	Model	CPU	RAM	Storage	Price	CustomerID	CabinID	OrderID
MP2S69Y	R750	INTEL i3	256	256	3982	519	9	534

In Orderr Table:

(Order ID - CustomerID)

534	519
-----	-----

9) CreateServer

Definition: This stored procedure adds a new server to the Server table using the entered ServiceTag, Model, CPU, RAM, Storage, Price, CabinID parameters.

```
CREATE PROCEDURE CreateServer
(
    @ServiceTag as nvarchar(10),
    @Model as nvarchar(20),
    @CPU as nvarchar(50),
    @RAM as nvarchar(10),
    @Storage as int,
    @Price as int,
    @CabinID as int)
AS
BEGIN
    Insert Into Server(ServiceTag,Model,CPU,RAM,Storage,Price,CabinID)
    Values (@ServiceTag,@Model,@CPU,@RAM,@Storage,@Price,@CabinID)
END
```

Before Executing:

ServiceTag	Model	CPU	RAM	Storage	Price	CustomerID	CabinID	OrderID
KZ1B15F	R210II	INTEL PENTIUM	32	20480	4464	777	9	116
LD9U53N	R740	AMD EPYC	64	4096	9668	920	9	117
LG5Y43V	R350	INTEL XEON E-...	64	5120	7686	920	9	118
LJ4O82Y	R7515	INTEL CELERON	32	20480	4694	862	9	119
LK4K44J	R6515	INTEL XEON	32	5120	7780	941	9	120
LN5V63Z	R210	INTEL i3	128	12288	4420	800	9	121
LN5Z12R	R540	INTEL PENTIUM	128	1024	9589	895	9	122
LS3A23A	R640	INTEL CELERON	256	4096	7399	957	9	123
LT5G35W	R340	INTEL XEON	32	20480	1076	911	9	124
LV6J43N	R550	INTEL XEON E-...	64	512	6915	911	9	125
MB4D22U	T440	INTEL PENTIUM	32	8192	7285	583	9	126

After Executing:

```
exec CreateServer 'L43BYTK','R210','INTEL PENTIUM','8','1024',1230,0
```

100 %

Messages

(1 row affected)

Completion time: 2022-12-28T21:56:38.0491610+03:00

ServiceTag	Model	CPU	RAM	Storage	Price	CustomerID	CabinID	OrderID
KZ1B15F	R210II	INTEL PENTIUM	32	20480	4464	777	9	116
L43BYTK	R210	INTEL PENTIUM	8	1024	1230	NULL	0	NULL
LD9U53N	R740	AMD EPYC	64	4096	9668	920	9	117

10) DeleteServer

Definition: This stored procedure deletes the relevant server according to the entered ServiceTag parameter. Here, it also checks whether there is a customer belonging to this server and prints an error message accordingly so that the wrong server is not deleted.

```
Create Procedure DeleteServer (@ServiceTag as nvarchar(10))
As
Begin
declare
@tempServiceTag nvarchar(10);

Select @tempServiceTag=s.ServiceTag From Server s Where s.CustomerID is null and s.ServiceTag=@ServiceTag

if @tempServiceTag is not null
Begin
Delete From Server Where ServiceTag=@ServiceTag
End
else
Begin
print 'Server cannot be deleted because it has been hired.'
End
End
GO
```

Before Executing:

ServiceTag	Model	CPU	RAM	Storage	Price	CustomerID	CabinID	OrderID
AA4S82Y	XR11	INTEL XEON E-...	128	2048	1670	882	0	1
AJ3S82N	T550	AMD EPYC	32	12288	5090	703	0	2
AK6S46Z	R240	INTEL i3	64	15360	4319	918	0	3
AQ4A69R	R7515	INTEL PENTIUM	32	4096	8568	705	0	4
AS3C34H	R210	INTEL CELERON	128	20480	3538	705	0	5
AZ9Q12J	R6515	INTEL CELERON	32	20480	6077	705	0	6
BC7D31O	R6515	INTEL i3	256	4096	2880	918	0	7

ServiceTag	Model	CPU	RAM	Storage	Price	CustomerID	CabinID	OrderID
KZ1B15F	R210II	INTEL PENTIUM	32	20480	4464	777	9	116
L43BYTK	R210	INTEL PENTIUM	8	1024	1230	NULL	0	NULL
LD9U53N	R740	AMD EPYC	64	4096	9668	920	9	117
LG5Y43V	R350	INTEL XEON E-...	64	5120	7686	920	9	118
LJ4O82Y	R7515	INTEL CELERON	32	20480	4694	862	9	119
LK4K44J	R6515	INTEL XEON	32	5120	7780	941	9	120
LN5V63Z	R210	INTEL i3	128	12288	4420	800	9	121
LN5Z12R	R540	INTEL PENTIUM	128	1024	9589	895	9	122
LS3A23A	R640	INTEL CELERON	256	4096	7399	957	9	123
LT5G35W	R340	INTEL XEON	32	20480	1076	911	9	124
LV6J43N	R550	INTEL XEON E-...	64	512	6915	911	9	125
MB4D22U	T440	INTEL PENTIUM	32	8192	7285	583	9	126

After Executing:

```
exec DeleteServer 'AA4S82Y'
```

100 %

Messages

Server cannot be deleted because it has been hired.

Completion time: 2022-12-28T22:18:59.9289345+03:00

ServiceTag	Model	CPU	RAM	Storage	Price	CustomerID	CabinID	OrderID
AA4S82Y	XR11	INTEL XEON E-...	128	2048	1670	882	0	1
AJ3S82N	T550	AMD EPYC	32	12288	5090	703	0	2
AK6S46Z	R240	INTEL i3	64	15360	4319	918	0	3
AQ4A69R	R7515	INTEL PENTIUM	32	4096	8568	705	0	4
AS3C34H	R210	INTEL CELERON	128	20480	3538	705	0	5
AZ9Q12J	R6515	INTEL CELERON	32	20480	6077	705	0	6
BC7D31O	R6515	INTEL i3	256	4096	2880	918	0	7

```
exec DeleteServer 'L43BYTK'
```

100 %

Messages

(1 row affected)

Completion time: 2022-12-28T22:21:01.3730545+03:00

ServiceTag	Model	CPU	RAM	Storage	Price	CustomerID	CabinID	OrderID
KZ1B15F	R210II	INTEL PENTIUM	32	20480	4464	777	9	116
LD9U53N	R740	AMD EPYC	64	4096	9668	920	9	117
LG5Y43V	R350	INTEL XEON E-...	64	5120	7686	920	9	118
LJ4O82Y	R7515	INTEL CELERON	32	20480	4694	862	9	119
LK4K44J	R6515	INTEL XEON	32	5120	7780	941	9	120
LN5V63Z	R210	INTEL i3	128	12288	4420	800	9	121
LN5Z12R	R540	INTEL PENTIUM	128	1024	9589	895	9	122
LS3A23A	R640	INTEL CELERON	256	4096	7399	957	9	123
LT5G35W	R340	INTEL XEON	32	20480	1076	911	9	124
LV6J43N	R550	INTEL XEON E-...	64	512	6915	911	9	125
MB4D22U	T440	INTEL PENTIUM	32	8192	7285	583	9	126

11) totalAmountGivenDateRange

Definition: This procedure calculates the total amount of orders for given date range.

```

Create Procedure totalAmountGivenDateRange(
    @startDate date,
    @endDate date
)
as
Begin
    select sum(s.Price) as TotalAmountOfOrdersInGivenRange
    From Customer c inner join Orderr o on c.Ssn=o.CustomerID
        inner join Server s on o.CustomerID=s.CustomerID and o.OrderID=s.OrderID,Bill b
    Where @startDate<=b.Date and @endDate>=b.Date
    end;

exec totalAmountGivenDateRange '2017-03-05','2021-11-15'

```

100 %

Results Messages

	TotalAmountOfOrdersInGivenRange
1	25473132

12) totalAmount

Definition: This procedure calculates the total expense of a desired customer then shows the Ssn,FullName and calculated expense.

```

Create Procedure totalAmount(
    @FullName nvarchar(100),
    @Ssn int
)
as
Begin
    Select c.Ssn,c.FullName,sum(s.Price) as TotalAmountOfOrders
    From Customer c inner join Orderr o on c.Ssn=o.CustomerID
        inner join Server s on o.CustomerID=s.CustomerID and o.OrderID=s.OrderID
    Where @Ssn=o.CustomerID and @FullName=c.FullName
    Group by c.Ssn,c.FullName

    end;

exec totalAmount 'Tahir Eracar',504

```

100 %

Results Messages

	Ssn	FullName	TotalAmountOfOrders
1	504	Tahir Eracar	2984

13) customerInSameCabin

Definition: This procedure retrieves the Ssn,FullName and CabinID of customers whose share same cabin.

```
Create procedure customerInSameCabin(
    @CabinID int
)
as
Begin
    select c.Ssn,c.FullName,cbn.CabinID
    From Server s inner join Cabin cbn on s.CabinID=cbn.CabinID, Customer c
    Where @CabinID=cbn.CabinID
End

exec customerInSameCabin 1
```

100 %

Results Messages

	Ssn	FullName	CabinID
1	504	Tahir Eracar	1
2	511	Somer Sevinçhan	1
3	513	Kadir Özpek	1
4	519	Hüma Mosso	1
5	525	Ertuç Balıkcı	1
6	530	Mahir Kapudere	1
7	534	Abdülkadir Çirkin	1
8	537	Cemre Özşimşek	1
9	541	Burhan Bağcı	1
10	546	Özge Evsen	1
11	548	Tayfun Akbay	1
12	555	Vural Çintay	1
13	564	Burak Erçin	1
14	567	Sadık Kartal	1
15	583	Faruk Söylem	1
16	584	Tolgay Tutak	1

✓ Query executed successfully. FFARUKK\MSSQLSE

14) createCustomer

Definition: This stored procedure inserts a new customer to the company by personal informations.

```
Create Procedure createCustomer
@Ssn int,
@FirstName nvarchar(40),
@LastName nvarchar(50),
@Gender char(1),
@Birthdate smalldatetime,
@PhoneNumber nvarchar(11),
@Mail nvarchar(100),
@UserPassword nvarchar(16)
As
Insert Into Customer(Ssn, FirstName, LastName, Gender, BirthDate, PhoneNumber, Mail, UserPassword)
values(@Ssn, @FirstName, @LastName, @Gender, @Birthdate, @PhoneNumber, @Mail, @UserPassword);
```

Before Executing:

	Ssn	FirstName	LastName	FullName	Gender	BirthDate	Age	PhoneNum...	Mail	UserID	UserPasswo...	Representa...
	884	Selen	Keklikiçi	Selen Keklikçi	K	1956-11-04 ...	66	05931328263	at@ac.wf	1020	YE4N44G1J4	115
	885	Enver	Arıyan	Enver Arıyan	E	1976-05-09 ...	46	05859476560	Aliquam@h...	1030	JR5I88B2Z5	115
	889	Berkkan	Özgüven	Berkkan Öz...	E	1989-01-02 ...	33	05555328038	gravida@nu...	1082	DJ6D57Y4L2	115
	890	Mücahit	Yapar	Mücahit Yap...	E	1965-09-11 ...	57	05852921081	sem@Viva...	1008	GR7B58T4G7	115
	895	Gonca	Ekim	Gonca Ekim	K	1987-03-28 ...	35	05973805560	dapibus@n...	1074	QW7E38Q1...	116
	899	Sami	Çetin	Sami Çetin	E	1968-08-19 ...	54	05831027689	adipidsscin...	1118	CKS578A3G4	116
	908	Yağız	Baytak	Yağız Baytak	E	1975-07-17 ...	47	05268873163	odio@ac.pn	1044	LN2A44W4...	116
	911	Engin	Parçalı	Engin Parçalı	E	1991-05-17 ...	31	05997593267	at@leo.ht	1094	SJ9W79L7K5	116
	915	Özbay	Kaçar	Özbay Kaçar	E	1957-10-09 ...	65	05632923193	faucibus@ia...	1046	ST6C98D7A6	117
	918	Şadi	Şanal	Şadi Şanal	E	1973-09-07 ...	49	05029097851	Nullam@sc...	1011	IY7D57P1M1	117
	920	Süeda	Apohan	Süeda Apoh...	K	1959-06-16 ...	63	05942735378	velit@moli...	1062	CF2O63D6O8	117
	927	Cemil	Çilgın	Cemil Çilgın	E	1963-02-13 ...	59	05661033348	mollis@Do...	1100	YI9G44C1Q9	118
	928	Rıza	Doğan	Rıza Doğan	E	1973-05-30 ...	49	05858138281	diam@port...	1070	UL3J95O1G8	118
	932	Arda	Berktaş	Arda Berktaş	E	1967-03-15 ...	55	05743245187	eget@laore...	1009	VU4P46O1K5	119
	941	Bilal	Delen	Bilal Delen	E	1977-09-05 ...	45	05952904550	nunc@portt...	1102	BW6W81K1...	119
	946	Selim	Andaç	Selim Andaç	E	2000-07-15 ...	22	05464066806	consectetur...	1042	PM2E32B4U5	119
	948	Halime	Altınkırın	Halime Altı...	K	1995-07-02 ...	27	05165772987	Pellentesqu...	1096	YV2Y42R1G5	119
	952	Özge	Kartal	Özge Kartal	K	1952-10-30 ...	70	05499746384	at@arcu.pr	1024	FP4M25J5J7	37
	957	Nursel	Karakelle	Nursel Kara...	K	1992-08-26 ...	30	05986282524	imperdiet@...	1097	GA5Q24E8J7	37
	963	Birsen	Demez	Birsen Dem...	K	1958-01-31 ...	64	05338924822	pharetra@e...	1101	OM9D67G8...	47
	964	Gülnur	Akın	Gülnur Akın	K	1971-05-23 ...	51	05853073127	ante@male...	1037	BR3A22A6P7	106
	972	Öznur	Özüylmaz	Öznur Özüyl...	K	1952-11-24 ...	70	05486048008	massa@a.tn	1056	HQ8Y26R9...	26
	975	Öztürk	Köklü	Öztürk Köklü	E	1970-01-29 ...	52	05911873550	odio@mole...	1052	JQ9Y22T8C4	31
	983	Bekir	Bozmaz	Bekir Bozmaz	E	1987-12-14 ...	35	05127328028	convallis@i...	1098	CC6L61N5V4	31
	984	Fatih	Karakoyunlu	Fatih Karako...	E	2001-08-06 ...	21	05326600897	ornare@sus...	1036	UY3P71B6K4	109
	986	Memati	Baş	Memati Baş	E	1976-12-10 ...	46	05765008961	velit@dolor...	1025	SS9Q64G5B4	108
	987	Tuğçe	Ulukaya	Tuğçe Uluka...	K	1956-05-04 ...	66	05785579062	vitae@blan...	1013	CA4Q55D2...	110
	997	Oğuzhan	Buluş	Oğuzhan Bu...	E	1966-12-04 ...	56	05447872076	amet@eget...	1035	KU4C82A9A9	110
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

After Executing:

```
exec createCustomer '3000', 'Murat', 'Özcan', 'E', '1968-12-19', '05300068850', 'Phdsaddudsas@lobortis.cc', 'ZHA56DSA51'
```

Messages

(1 row affected)

(1 row affected)

Completion time: 2022-12-30T17:22:57.6564890+03:00

Ssn	FirstName	LastName	FullName	Gender	BirthDate	Age	PhoneNum...	Mail	UserID	UserPasswo...	Representa...
885	Enver	Arıyan	Enver Arıyan	E	1976-05-09 ...	46	05859476560	Aliquam@h...	1030	JR5I88B2Z5	115
889	Berkkan	Özgüven	Berkkan Öz...	E	1989-01-02 ...	33	05555328038	gravida@nu...	1082	DJ6D57Y4L2	115
890	Mücahit	Yapar	Mücahit Yap...	E	1965-09-11 ...	57	05852921081	sem@Viva...	1008	GR7B58T4G7	115
895	Gonca	Ekim	Gonca Ekim	K	1987-03-28 ...	35	05973805560	dapibus@n...	1074	QW7E38Q1...	116
899	Sami	Çetin	Sami Çetin	E	1968-08-19 ...	54	05831027689	adipidsscin...	1118	CKS578A3G4	116
908	Yağiz	Baytak	Yağiz Baytak	E	1975-07-17 ...	47	05268873163	odio@ac.pn	1044	LN2A44W4...	116
911	Engin	Parçalı	Engin Parçalı	E	1991-05-17 ...	31	05997593267	at@leo.ht	1094	SJ9W79L7K5	116
915	Özbay	Kaçar	Özbay Kaçar	E	1957-10-09 ...	65	05632923193	faucibus@ia...	1046	ST6C98D7A6	117
918	Şadi	Şanal	Şadi Şanal	E	1973-09-07 ...	49	05029097851	Nullam@sc...	1011	IY7D57P1M1	117
920	Süeda	Apohan	Süeda Apoh...	K	1959-06-16 ...	63	05942735378	velit@moli...	1062	CF2063D608	117
927	Cemil	Çilgin	Cemil Çilgin	E	1963-02-13 ...	59	05661033348	mollis@Do...	1100	YI9G44C1Q9	118
928	Rıza	Doğan	Rıza Doğan	E	1973-05-30 ...	49	05858138281	diam@port...	1070	UL3J95O1G8	118
932	Arda	Berktaş	Arda Berktaş	E	1967-03-15 ...	55	05743245187	eget@laore...	1009	VU4P46O1K5	119
941	Bilal	Delen	Bilal Delen	E	1977-09-05 ...	45	05952904550	nunc@port...	1102	BW6W81K1...	119
946	Selim	Andaç	Selim Andaç	E	2000-07-15 ...	22	05464066806	consectetur...	1042	PM2E32B4U5	119
948	Halime	Altınkiran	Halime Altı...	K	1995-07-02 ...	27	05165772987	Pellentesqu...	1096	VY2Y42R1G5	119
952	Özge	Kartal	Özge Kartal	K	1952-10-30 ...	70	05499746384	at@arcu.pr	1024	FP4M25J5J7	37
957	Nursel	Karakelle	Nursel Kara...	K	1992-08-26 ...	30	05986282524	imperdiet@...	1097	GA5Q24E8J7	37
963	Birsen	Demez	Birsen Dem...	K	1958-01-31 ...	64	05338924822	pharetra@e...	1101	OM9D67G8...	47
964	Gülnur	Akın	Gülnur Akın	K	1971-05-23 ...	51	05853073127	ante@male...	1037	BR3A22A6P7	106
972	Öznur	Özüylmaz	Öznur Özüyl...	K	1952-11-24 ...	70	05486048008	massa@a.tn	1056	HQ8Y26R9...	26
975	Öztürk	Koklu	Öztürk Koklu	E	1970-01-29 ...	52	05911873550	odio@mole...	1052	JQ9Y22T8C4	31
983	Bekir	Bozmaz	Bekir Bozmaz	E	1987-12-14 ...	35	05127328028	convallis@i...	1098	CC6L61N5V4	31
984	Fatih	Karakoyunlu	Fatih Karako...	E	2001-08-06 ...	21	05326600897	ornare@sus...	1036	UY3P71B6K4	109
986	Memati	Baş	Memati Baş	E	1976-12-10 ...	46	05765008961	velit@dolor...	1025	SS9Q64G5B4	108
987	Tuğçe	Ulukaya	Tuğçe Uluka...	K	1956-05-04 ...	66	05785579062	vitae@blan...	1013	CA4Q55D2...	110
997	Oğuzhan	Buluş	Oğuzhan Bu...	E	1966-12-04 ...	56	05447872076	amet@eget...	1035	KU4C82A9A9	110
3000	Murat	Özcan	Murat Özcan	E	1968-12-19 ...	54	05300068850	Phdsadduds...	1135	ZHAd56DS...	119
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL