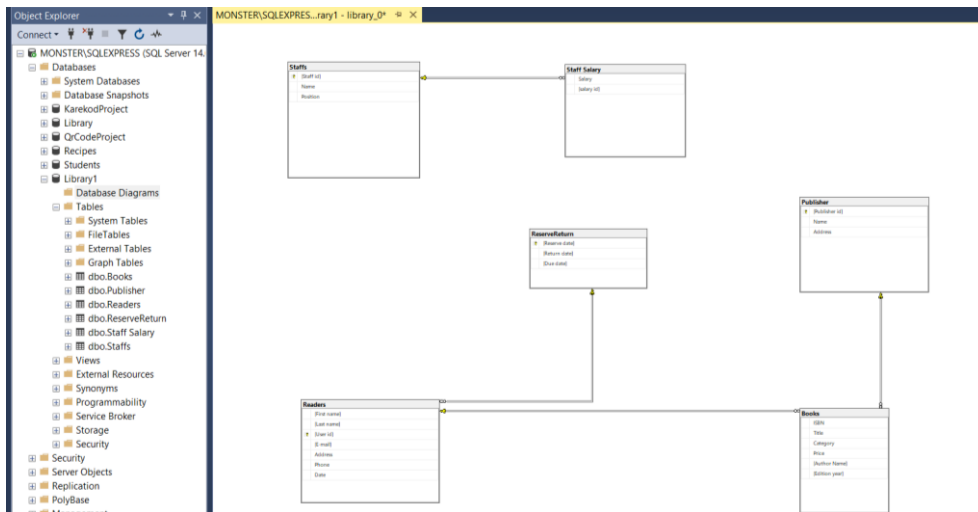




CSE202 Database Management Systems 2021/2022 Spring – Project -Part 2

Members	Student Numbers
Kübra Uçar	191805067
Kevser Öztürk	191805054
Rabia Yıldırım	191805043

- We made a database Diagram and we created primary keys and foreign keys in our tables.



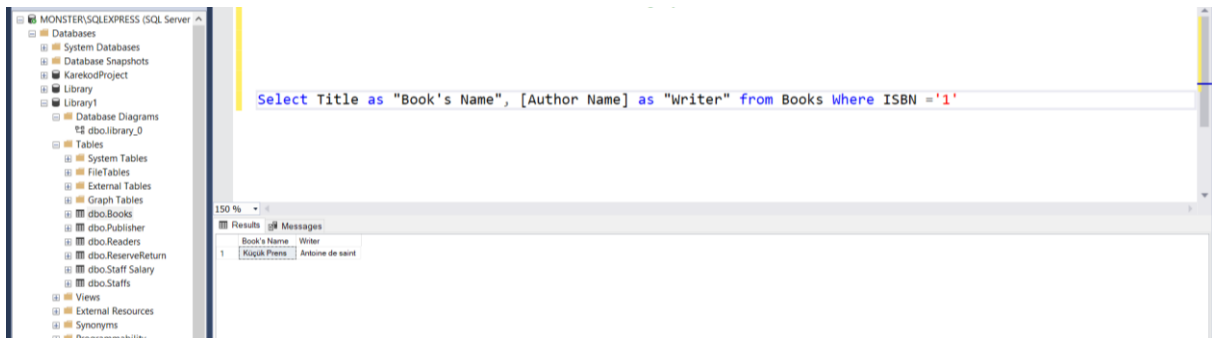
- We created the tables and we defined 10 values for each one.

The screenshot displays six SQL query results windows, each showing 10 rows of data from the database tables:

- SQLQuery8.sql - M_ONSTER\kubra (59):** Results for Staff table.
- SQLQuery7.sql - M_ONSTER\kubra (54):** Results for StaffSalary table.
- SQLQuery4.sql - M_ONSTER\kubra (55):** Results for Reservereturn table.
- SQLQuery6.sql - M_ONSTER\kubra (58):** Results for Publisher table.
- SQLQuery9.sql - M_ONSTER\kubra (60):** Results for Books table.
- SQLQuery5.sql - M_ONSTER\kubra (56):** Results for Readers table.

Queries

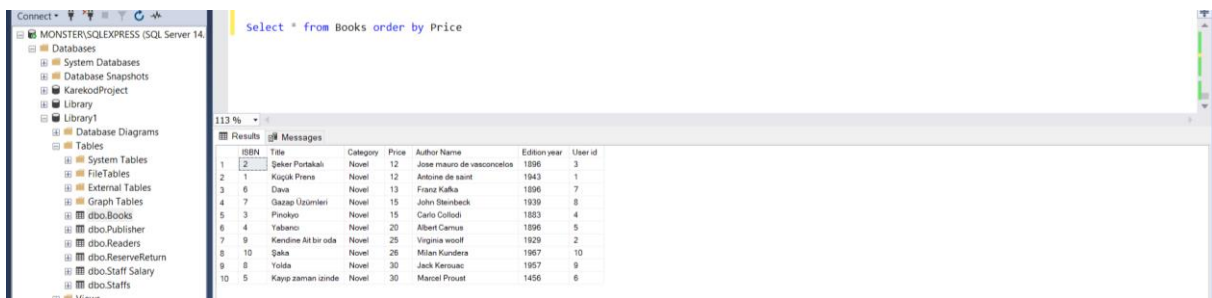
Query 1 : We used Where clause with two condition which names Book's Name and Author Name.



```
Select Title as "Book's Name", [Author Name] as "Writer" from Books where ISBN ='1'
```

Book's Name	Writer
Küçük Prens	Antoine de saint

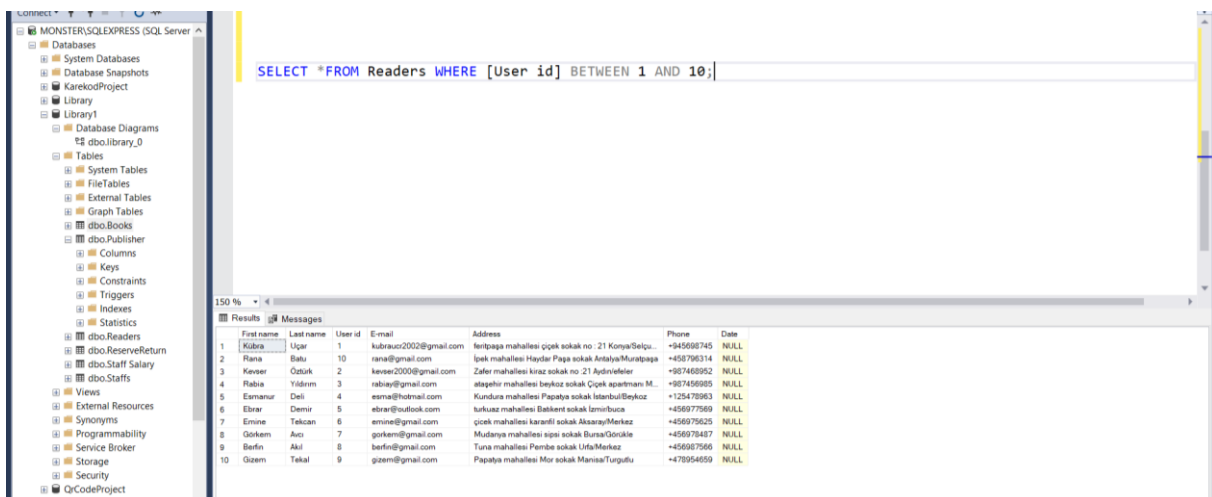
Query2 : We used order by to sort Book's Prices.



```
Select * from Books order by Price
```

ISBN	Title	Category	Price	Author Name	Edition year	User id
2	Şeker Portakalı	Novel	12	Jose mauro de vasconcelos	1896	3
1	Küçük Prens	Novel	12	Antoine de saint	1943	1
6	Dava	Novel	13	Franz Kafka	1896	7
7	Gastrap Uzunirisi	Novel	15	Johv Steinbeck	1939	8
3	Pinoquio	Novel	15	Carlo Collodi	1883	4
4	Yabancı	Novel	20	Albert Camus	1896	5
9	Kendine Ait bir oda	Novel	25	Virginia woolf	1829	2
10	Şaka	Novel	26	Milan Kundera	1987	10
8	Yolda	Novel	30	Jack Kerouac	1957	9
5	Kayıp zaman içinde	Novel	30	Marcel Proust	1456	6

Query 3: We used Between - and to find User id between at 1 and 10.



```
SELECT *FROM Readers WHERE [User id] BETWEEN 1 AND 10;
```

First name	Last name	User id	E-mail	Address	Phone	Date
Kubra	Uşur	1	kubrucu2002@gmail.com	İzmirpapa mahallesi çiçek sokak no :21 Konya/Selçuk	+945060745	NULL
Rana	Batu	10	rana@gmail.com	İpek mahallesi Haydar Paşa sokak Antalya/Muratpaşa	+458796314	NULL
Kevser	Öztürk	2	kevser2000@gmail.com	Zafer mahallesi kız sokak no :21 Aydın/efeler	+987468952	NULL
Rabia	Yıldırım	3	rabray@gmail.com	ataşehir mahallesi beykoz sokak Çiçek apartmanı M...	+987456985	NULL
Ermanur	Deli	4	eemal@hotmail.com	Kundura mahallesi Papazpa sokak İstanbul/Beykoz	+125470963	NULL
Ebrar	Demir	5	ebrar@outlook.com	turkuz mahallesi Bakırcı sokak İzmir/Buza	+456977569	NULL
Emine	Telcan	6	emine@gmail.com	çiçek mahallesi karatlı sokak Akasay/Merkez	+456975625	NULL
Gökrem	Avcı	7	gorkem@gmail.com	Mudanya mahallesi ipci sokak Bursa/Gözüke	+456978487	NULL
Berfin	Akıl	8	berfin@gmail.com	Tuna mahallesi Pembe sokak Ufuk/Merkez	+456987596	NULL
Gözem	Tekel	9	gozem@gmail.com	Papazpa mahallesi Mor sokak Manisa/Turgutlu	+478954659	NULL

Query 4: We found values which do not have not null.

The screenshot shows the SQL Server Enterprise Manager interface. On the left, the Object Explorer displays the database structure for 'MONSTER\SQLEXPRESS (SQL Server)'. The central pane shows a SQL query: `Select * From Staffs Where Position is not null;`. The bottom pane displays the results of the query in a table with 10 rows and 3 columns: Staff id, Name, and Position.

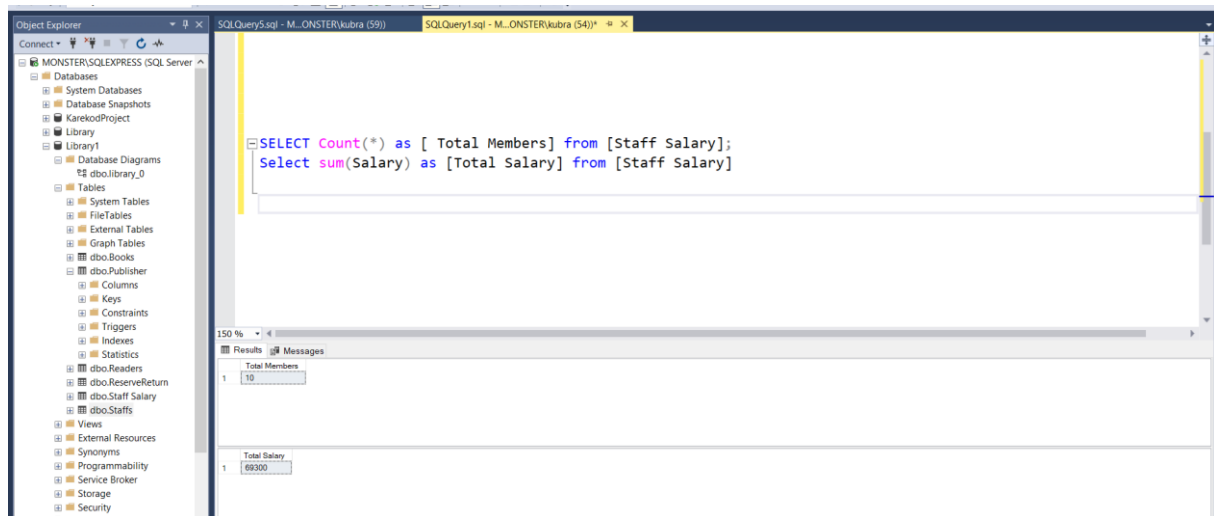
Staff id	Name	Position
1	Hasan Ciolek	manager
2	Hilal kar	Student
3	Ash Onsek	manager
4	Tudge Kaya	Student
5	Ermen Yildiz	servant
6	Kenan Aslan	Officer
7	Talat Orman	Officer
8	Marve Oz	Student
9	Tahsin erbag	Manager
10	Ayşe Uyar	Servant

Query5: We calculated the Max and min Salary. Then we Found their average .

The screenshot shows the SQL Server Enterprise Manager interface. On the left, the Object Explorer displays the database structure for 'MONSTER\SQLEXPRESS (SQL Server)'. The central pane shows a SQL query: `select max(Salary) as MaxSalary, avg(Salary) as AvgSalary, min(Salary) as MinSalary from [Staff Salary]`. The bottom pane displays the results of the query in a table with 1 row and 3 columns: MaxSalary, AvgSalary, and MinSalary.

MaxSalary	AvgSalary	MinSalary
12000	6930	800

Query 6: We calculated the count of total members and we calculated the total salary.



The screenshot shows the SQL Server Enterprise Manager interface. The Object Explorer on the left displays the database structure for 'MONSTER\SQLEXPRESS (SQL Server)'. The main window shows a SQL query in the 'SQLQuery1.sql' file:

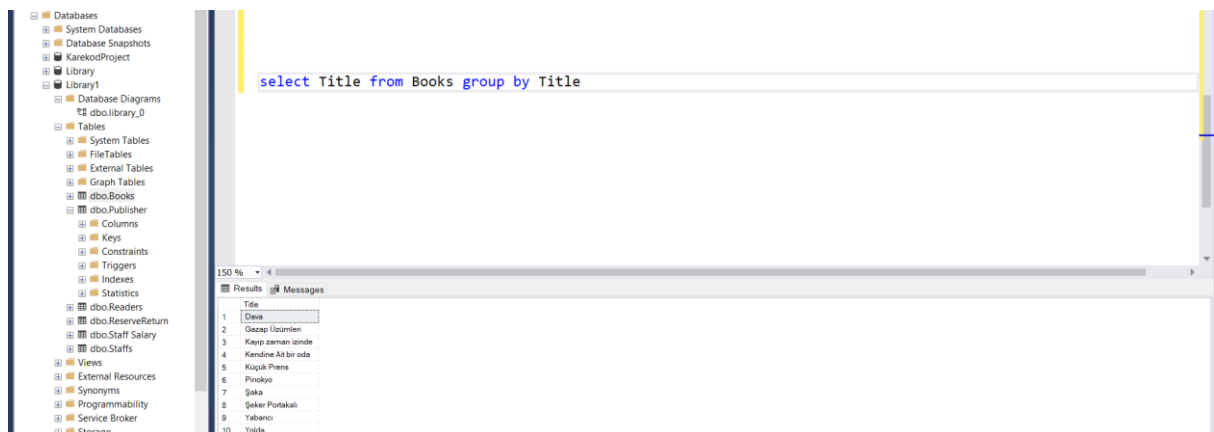
```
SELECT Count(*) as [Total Members] from [Staff Salary];  
Select sum(Salary) as [Total Salary] from [Staff Salary]
```

The Results pane at the bottom displays the output of the query:

	Total Members
1	10

	Total Salary
1	69300

Query 7 : We used 'Group by ' for allowing Grouping of rows of a query.



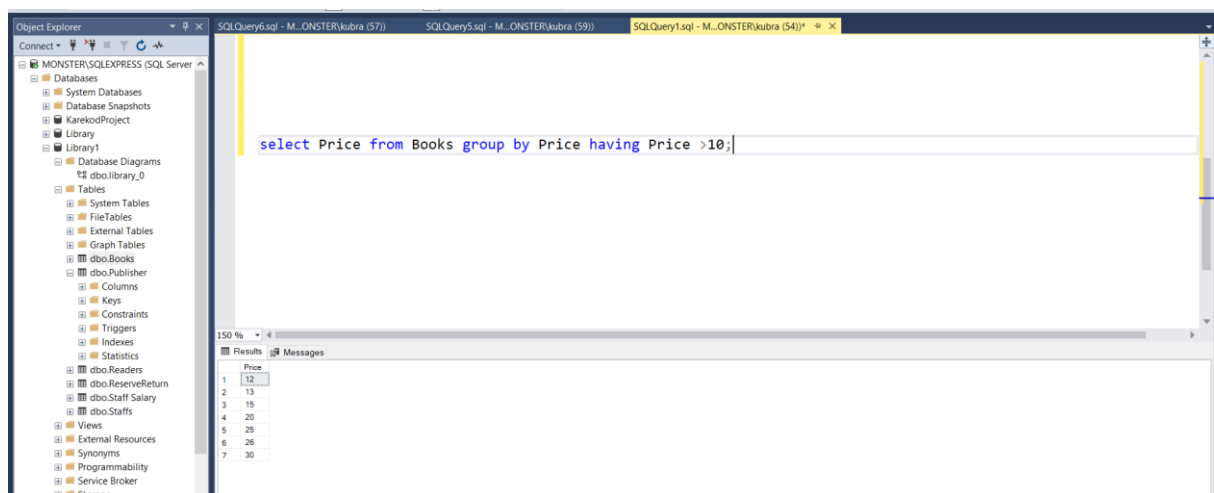
The screenshot shows the SQL Server Enterprise Manager interface. The Object Explorer on the left displays the database structure for 'MONSTER\SQLEXPRESS (SQL Server)'. The main window shows a SQL query in the 'SQLQuery1.sql' file:

```
select Title from Books group by Title
```

The Results pane at the bottom displays the output of the query:

	Title
1	Özlem
2	Özlem Üzümleri
3	Kayıp zaman içinde
4	Kendine ait bir oda
5	Küçük Prenses
6	Pirinç
7	Şaka
8	Şeker Portakalı
9	Yabancı
10	Yolda

Query 8: We having clause is used to filter record from the groups based on the specified condition.



The screenshot shows the SQL Server Enterprise Manager interface. The Object Explorer on the left displays the database structure for 'MONSTER\SQLEXPRESS (SQL Server)'. The main window shows a SQL query in the 'SQLQuery1.sql' file:

```
select Price from Books group by Price having Price >10;
```

The Results pane at the bottom displays the output of the query:

	Price
1	12
2	13
3	15
4	20
5	25
6	26
7	30

Query 9 :We used 'like' to find to contains 'a' substring.

The screenshot shows the SQL Server Enterprise Manager interface. The left pane displays the database structure for 'MONSTER\SQLEXPRESS (SQL Server 14)'. The right pane shows a query window with the following SQL code:

```
Select * From Books Where Title like '%a%'
```

Below the query window, the 'Results' pane displays the following data:

ISBN	Title	Category	Price	Author Name	Edition year
1	Şaka	Novel	26	Milan Kundera	1967
2	Şeker Portakalı	Novel	12	Jose mauro de vasconcelos	1896
3	Yabanı	Novel	20	Albert Camus	1896
4	Kayıp zaman içinde	Novel	30	Marcel Proust	1456
5	Dava	Novel	13	Franz Kafka	1896
6	Gazap Üzümleri	Novel	15	John Steinbeck	1939
7	Yolda	Novel	30	Jack Kerouac	1957
8	Kendine Ait bir oda	Novel	25	Virginia woolf	1929

Query 10: Using 'Update ', We added more 400 dollars to salaries for each Staff.

The screenshot shows the SQL Server Enterprise Manager interface. The left pane displays the database structure for 'MONSTER\SQLEXPRESS (SQL Server 14)'. The right pane shows a query window with the following SQL code:

```
UPDATE [Staff Salary] SET Salary=Salary+400

select * from [Staff Salary]

/*we Updated the staff's salaries*/
```

Below the query window, the 'Results' pane displays the following data:

Salary	salary id
13800	1
11200	2
10200	3
2000	4
9200	5
8200	6
10700	7
2200	8
5700	9
8700	10

Query 11: We combine records from two tables whenever there are matching values in a field common to both tables.(Books and Publisher)

The screenshot shows the SQL Server Enterprise Manager interface. The left pane displays the database structure for 'MONSTER\SQLEXPRESS (SQL Server 14)'. The right pane shows a query window with the following SQL code:

```
select * from Books inner join Publisher on Books.ISBN = Publisher.ISBN
```

Below the query window, the 'Results' pane displays the following data:

ISBN	Title	Category	Price	Author Name	Edition year	Publisher id	Name	Address	ISBN
1	Küçük Prenis	Novel	12	Antoine de saint	1943	1	Can	zzzzzzzzzzzz	1
2	Şeker Portakalı	Novel	12	Jose mauro de vasconcelos	1896	10	Kanyer	hhhhhhhhhh	2
3	Prodıyo	Novel	15	Carlo Collodi	1853	2	Yapı Kredi	yyyyyyyyyy	3
4	Yabanı	Novel	20	Albert Camus	1896	3	İş bankası	xxxxxxxxxx	4
5	Kayıp zaman içinde	Novel	30	Marcel Proust	1456	4	Doğan	bbbbbbbbbb	5
6	Dava	Novel	13	Franz Kafka	1896	5	Say	xxxxxxxxxx	6
7	Gazap Üzümleri	Novel	15	John Steinbeck	1939	6	Metis	pppppppppp	7
8	Yolda	Novel	30	Jack Kerouac	1957	7	Aynır	kkkkkkkkkk	8
9	Kendine Ait bir oda	Novel	25	Virginia woolf	1929	8	Baki	oooooooooo	9
10	Şaka	Novel	26	Milan Kundera	1967	9	Kırmızı Kedi	ssssssssssss	10

Query 12: We combine records from two tables whenever there are matching values in a field common to both tables.(Staff salary and Staff)

The screenshot shows the SQL Server Enterprise Manager interface. On the left, the 'Database Diagrams' folder is expanded, showing the 'dbo.Staff Salary' and 'dbo.Staffs' tables. The main window displays the following SQL query:

```
select * from [Staff Salary] r inner join Staffs s on r.Position = s.Position;
```

The results pane shows 25 rows of data. The columns are: Salary, salary id, Position, Staff id, First name, and Position. The data is as follows:

Salary	salary id	Position	Staff id	First name	Position
14200	1	Manager	1	Hasan Çiçek	Manager
11100	7	Manager	1	Hasan Çiçek	Manager
9100	10	Manager	1	Hasan Çiçek	Manager
10600	3	Student	10	Hilal kar	Student
2900	8	Student	10	Hilal kar	Student
14200	1	Manager	2	Ash Örnek	Manager
11100	7	Manager	2	Ash Örnek	Manager
9100	10	Manager	2	Ash Örnek	Manager
10600	3	Student	3	Tuğçe Kaya	Student
2900	8	Student	3	Tuğçe Kaya	Student
11600	2	Servant	4	Erme Yıldız	Servant
8800	6	Servant	4	Erme Yıldız	Servant
6100	9	Servant	4	Erme Yıldız	Servant
2400	4	Officer	5	Kerim Ailan	Officer
9600	5	Officer	5	Kerim Ailan	Officer
2400	4	Officer	6	Talat Osman	Officer
9600	5	Officer	6	Talat Osman	Officer
10600	3	Student	7	Manve Öz	Student
2900	8	Student	7	Manve Öz	Student
14200	1	Manager	8	Tahsin erb...	Manager
11100	7	Manager	8	Tahsin erb...	Manager
9100	10	Manager	8	Tahsin erb...	Manager
11600	2	Servant	9	Ape Uyar	Servant
8800	6	Servant	9	Ape Uyar	Servant
6100	9	Servant	9	Ape Uyar	Servant

Query 13: We combine records from two tables whenever there are matching values in a field common to both tables.(Readers and staff) We could not find any vallid value.

The screenshot shows the SQL Server Enterprise Manager interface. On the left, the 'Database Diagrams' folder is expanded, showing the 'dbo.Readers' and 'dbo.Staffs' tables. The main window displays the following SQL query:

```
select * from Readers inner join Staffs on Readers.[First name] = Staffs.[First name];
```

The results pane shows 0 rows of data, indicating that no records were found for the inner join.

Query 14: We used left outer join to extract the left table data only (Readers and Staffs)

The screenshot shows the SQL Server Enterprise Manager interface. On the left, the 'Database Diagrams' folder is expanded, showing the 'dbo.Readers' and 'dbo.Staffs' tables. The main window displays the following SQL query:

```
select * from Readers r left outer join Staffs s on r.[First name] = s.[First name];
```

The results pane shows 10 rows of data. The columns are: First name, Last name, User id, E-mail, Address, Phone, Date, Staff id, First name, and Position. The data is as follows:

First name	Last name	User id	E-mail	Address	Phone	Date	Staff id	First name	Position
Kilise	Uyar	1	kubrasur2002@gmail.com	tertipaga mahallesini cicek sokak no: 21 Konya/İzmir	+945698745	NULL	NULL	NULL	NULL
Rana	Batu	10	rana@gmail.com	İpek mahallesini Haydar Paşa sokak Antalya/Muratspa	+95796314	NULL	NULL	NULL	NULL
Keiser	Östürk	2	keiser2009@gmail.com	Zafer mahallesini kiraz sokak no: 21 Aydın/İzmir	+987488952	NULL	NULL	NULL	NULL
Rabia	Yıldırım	3	rabia@gmail.com	Ataşehir mahallesini beykoz sokak Çiçek apartmanı M...	+987456905	NULL	NULL	NULL	NULL
Ermener	Del	4	ermene@hotmail.com	Kundura mahallesini Paşapaya sokak İstanbul/Beykoz	+125478983	NULL	NULL	NULL	NULL
Ebrar	Demir	5	ebrai@outlook.com	Turkuz mahallesini Bakent sokak İzmir/buca	+456977569	NULL	NULL	NULL	NULL
Ermine	Tekcan	6	ermine@gmail.com	Çiçek mahallesini kararlı sokak Akaray/Merkez	+456975625	NULL	NULL	NULL	NULL
Gökrem	Avcı	7	gokrem@gmail.com	Mudanya mahallesini sipi sokak Bursa/Gözükle	+456978487	NULL	NULL	NULL	NULL
Berfin	Akı	8	berfin@gmail.com	Tuna mahallesini Pembe sokak Urfa/Merkez	+456987566	NULL	NULL	NULL	NULL
Özlem	Tekel	9	gozlem@gmail.com	Papaya mahallesini Mor sokak Manisa/Turgutlu	+478954859	NULL	NULL	NULL	NULL

Query 15: We used right outer join to extract the right table data only (Readers and Staffs)

The screenshot shows the SQL Server Enterprise Manager interface. The left pane displays the database structure, including the 'Library' database and its tables. The right pane shows the SQL query editor with the following query:

```
select * from Readers r right outer join Staffs s on r.[First name] = s.[First name];
```

Below the query editor, the 'Results' pane displays the output of the query. The results are as follows:

First name	Last name	User id	E-mail	Address	Phone	Date	Staff id	First name	Position
NULL	NULL	NULL	NULL	NULL	NULL	NULL	1	Hasan Çoek	Manager
NULL	NULL	NULL	NULL	NULL	NULL	NULL	10	Hilal kar	Student
NULL	NULL	NULL	NULL	NULL	NULL	NULL	2	Ahi Örnek	Manager
NULL	NULL	NULL	NULL	NULL	NULL	NULL	3	Tuğçe Kaya	Student
NULL	NULL	NULL	NULL	NULL	NULL	NULL	4	Emre Yıldız	Servant
NULL	NULL	NULL	NULL	NULL	NULL	NULL	5	Kerim Aslan	Officer
NULL	NULL	NULL	NULL	NULL	NULL	NULL	6	Talat Orman	Officer
NULL	NULL	NULL	NULL	NULL	NULL	NULL	7	Mane Öz	Student
NULL	NULL	NULL	NULL	NULL	NULL	NULL	8	Tahsin erbağ	Manager
NULL	NULL	NULL	NULL	NULL	NULL	NULL	9	Ayşe Uyar	Servant

Query 16: We combined the results of both left and right outer joins and returns all (matched or unmatched) rows from the tables on both sides of the join clause. (Readers and Staffs)

The screenshot shows the SQL Server Enterprise Manager interface. The left pane displays the database structure, including the 'Library' database and its tables. The right pane shows the SQL query editor with the following query:

```
select * from Readers r full outer join Staffs s on r.[First name] = s.[First name];
```

Below the query editor, the 'Results' pane displays the output of the query. The results are as follows:

First name	Last name	User id	E-mail	Address	Phone	Date	Staff id	First name	Position
Kubra	Uyar	1	kubrauc2002@gmail.com	feritpapa mahallesi: çiçek sokak no : 21 Konya/Seipu.	+945086745	NULL	NULL	NULL	NULL
Rana	Batu	10	rana@gmail.com	İpek mahallesi: Haydar Paşa sokak Antalya/Muratpaşa	+458796314	NULL	NULL	NULL	NULL
Kewer	Öztürk	2	kewer2000@gmail.com	Zafer mahallesi: kırsal sokak no : 21 Aydın/efeler	+867485952	NULL	NULL	NULL	NULL
Rabia	Yıldırım	3	rabia@gmail.com	atayeller mahallesi: İhsan sokak Çiçek apartmanı M.	+987456906	NULL	NULL	NULL	NULL
Ermannur	Delic	4	esma@hotmail.com	Kundura mahallesi: Papatyas sokak İstanbul/Beykoz	+125470963	NULL	NULL	NULL	NULL
Elbrar	Demir	5	elbrar@outlook.com	kurkuz mahallesi: Bakırcı sokak İzmir/buca	+456977569	NULL	NULL	NULL	NULL
Emine	Telcan	6	emine@gmail.com	çiçek mahallesi: karantlı sokak Aksaray/Merkez	+456975625	NULL	NULL	NULL	NULL
Ölkem	Akı	7	gozem@gmail.com	Mudanya mahallesi: ıyık sokak Bursa/Ortaköy	+456975457	NULL	NULL	NULL	NULL
Berfin	Akı	8	berfin@gmail.com	Tuna mahallesi: Pembe sokak Ufuk/Merkez	+456987566	NULL	NULL	NULL	NULL
Gözem	Tekel	9	gozem@gmail.com	Papatyas mahallesi: Mor sokak Manisa/Turgutlu	+478954659	NULL	NULL	NULL	NULL
NULL	NULL	NULL	NULL	NULL	NULL	NULL	1	Hasan Çoek	Manager
NULL	NULL	NULL	NULL	NULL	NULL	NULL	10	Hilal kar	Student
NULL	NULL	NULL	NULL	NULL	NULL	NULL	2	Ahi Örnek	Manager
NULL	NULL	NULL	NULL	NULL	NULL	NULL	3	Tuğçe Kaya	Student
NULL	NULL	NULL	NULL	NULL	NULL	NULL	4	Emre Yıldız	Servant
NULL	NULL	NULL	NULL	NULL	NULL	NULL	5	Kerim Aslan	Officer
NULL	NULL	NULL	NULL	NULL	NULL	NULL	6	Talat Orman	Officer
NULL	NULL	NULL	NULL	NULL	NULL	NULL	7	Mane Öz	Student
NULL	NULL	NULL	NULL	NULL	NULL	NULL	8	Tahsin erbağ	Manager
NULL	NULL	NULL	NULL	NULL	NULL	NULL	9	Ayşe Uyar	Servant

Query 17: We Created a Procedure for allow to create SQL queries to be stored and executed on the server,thus we can execute this query on server.

The screenshot shows the SQL Server Enterprise Manager interface. The left pane displays the database structure, including the 'Library' database and its tables. The right pane shows the SQL query editor with the following query:

```
CREATE PROCEDURE BookTest
as
Begin Select * from Readers
End
```

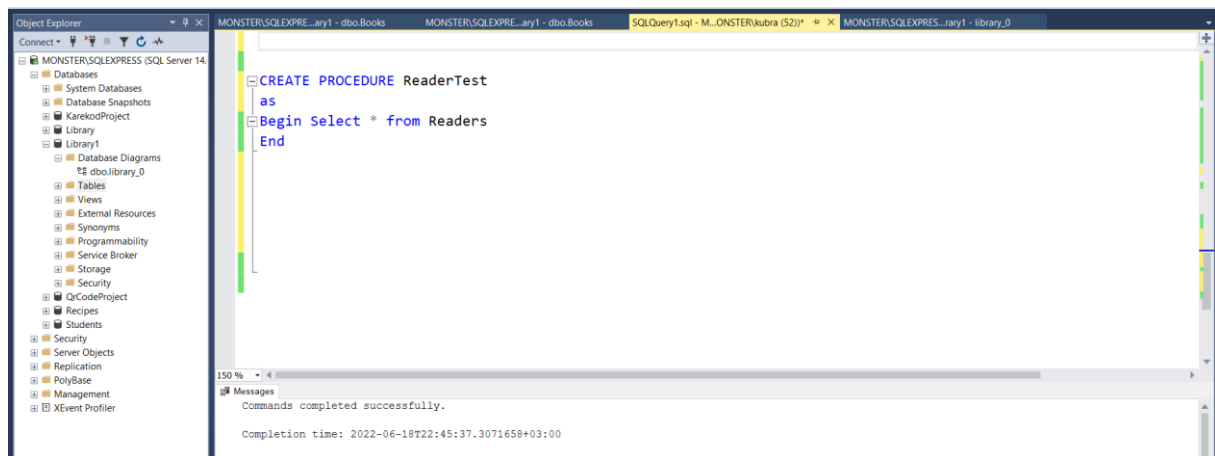
Below the query editor, the 'Messages' pane displays the output of the query. The results are as follows:

```
Commands completed successfully.

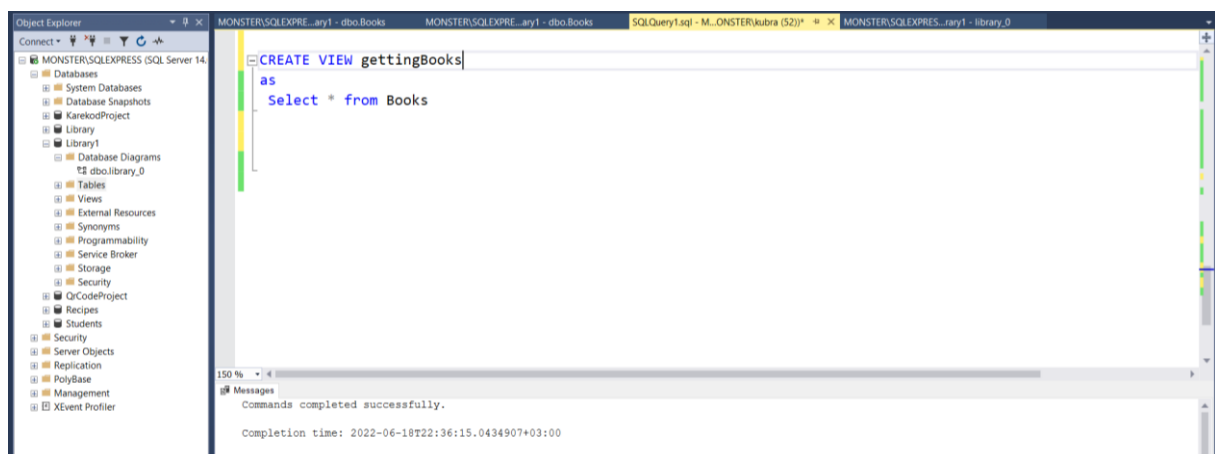
Completion time: 2022-06-18T22:47:21.1759553+03:00
```

Query 18:

We Created a Procedure for allow to create SQL queries to be stored and executed on the server,thus we can execute this query on server.



Query19 : Views are used for security purposes because they provide encapsulation of the name of the table



Query 20:

Views are used for security purposes because they provide encapsulation of the name of the table.

