

## **EINLEITUNG**

In diesem Projekt wurde Database der Supermarktes entwickelt. Auf der ersten Seite sehen wir die Anwendungen mit dem Namen Use Case Diagram oder auf Deutsch Anwendungsfalldiagramm.

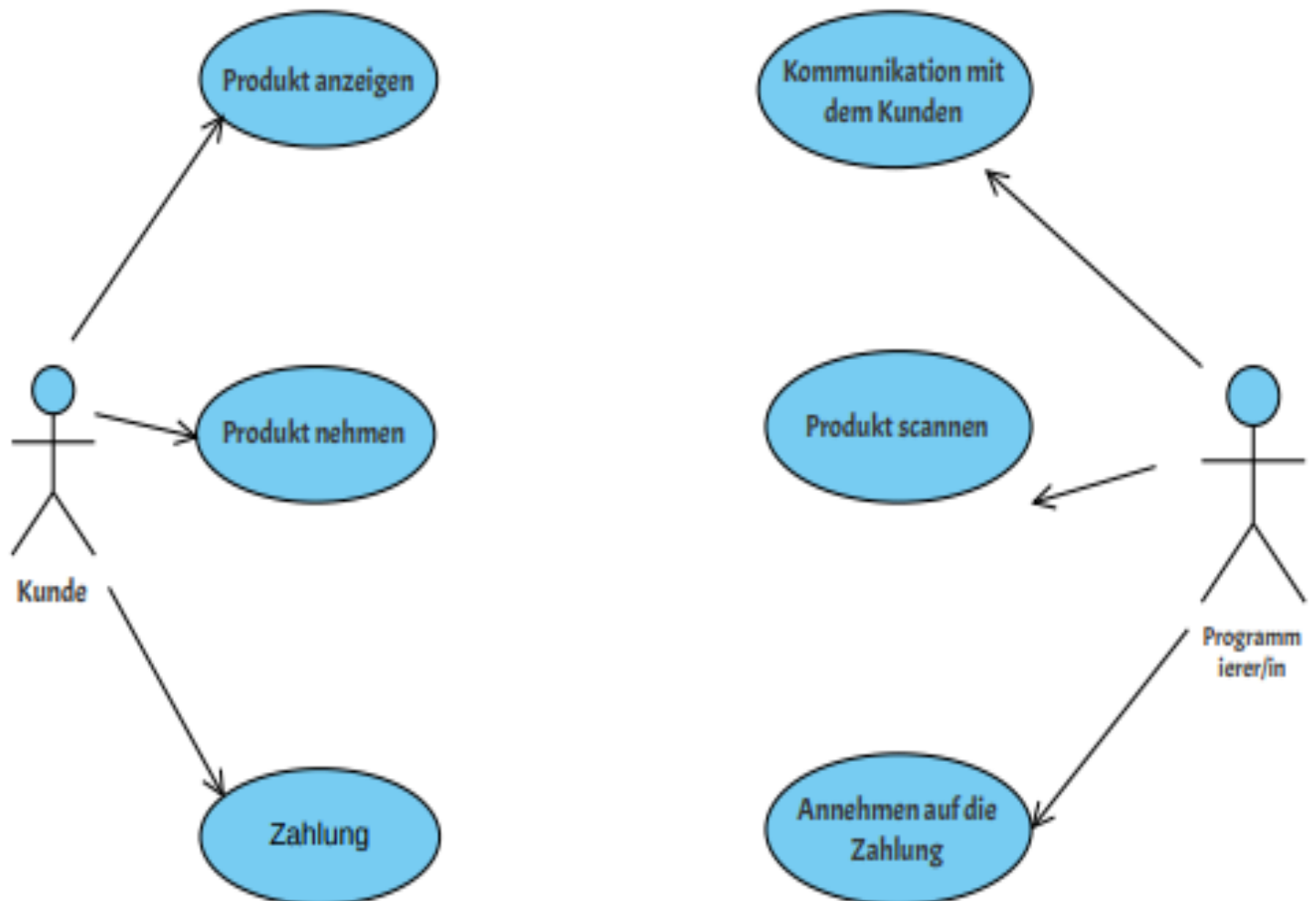
Auf der nächste Seite sehen wir E-R-Diagramm werden alle Tabellen, die in der Datenbank des Systems verwendet werden, mit den Verbindungen zwischen ihnen aufgelistet. Und auf der letzten Seiten sehen wir wieder gleichen Tabellen aber dieses Mal Haben wir sehen dass, die Relation zwischen Tabellen.

## **GESCHÄFTSREGEL**

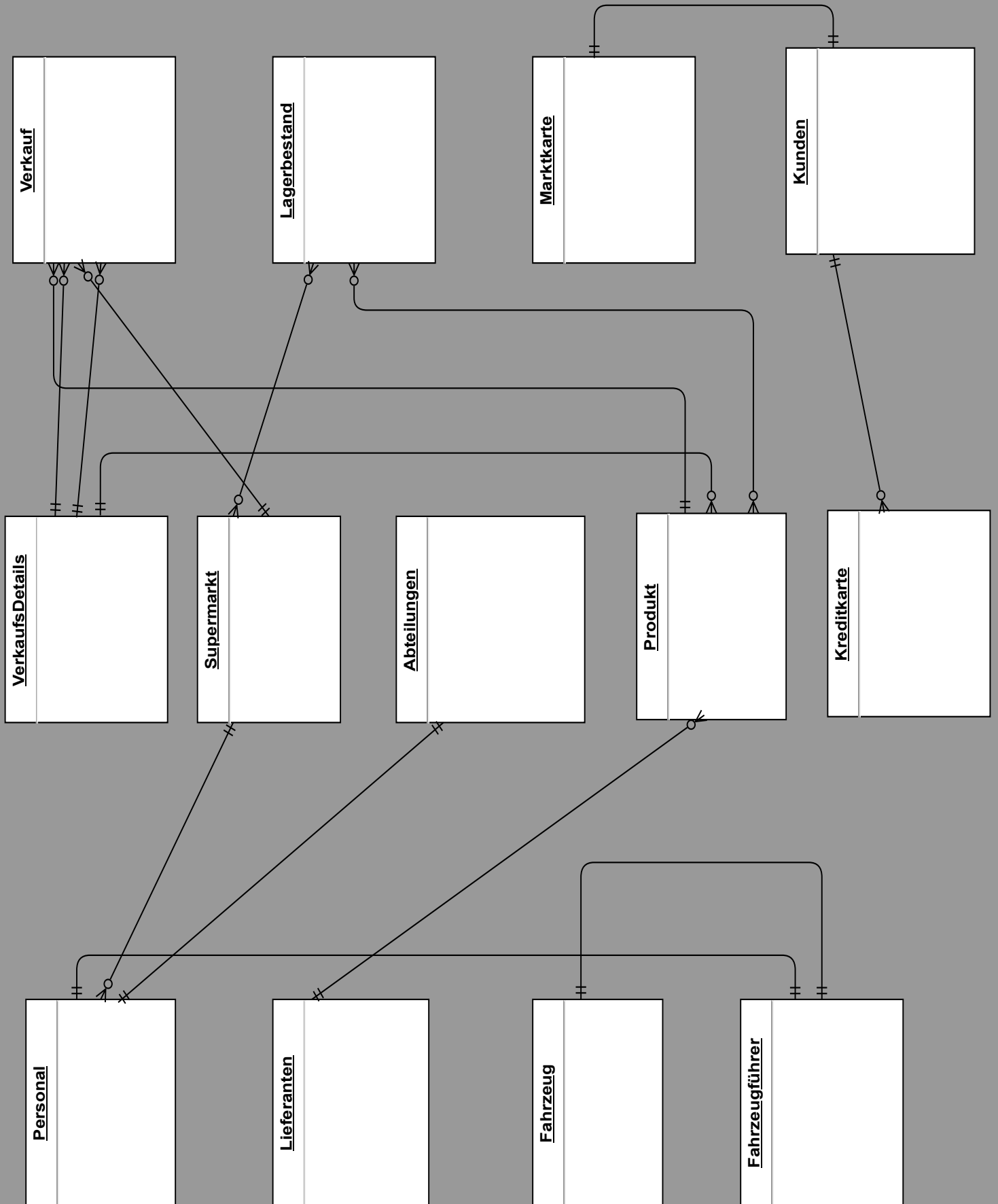
In diesem Supermarktkette für jede Supermarkt haben wir eigenen Personal außer Fahrer. Und in diesem Supermarktkette jede Produkt haben nur eine Liferanten haben Zusätzlich ein Liferanten kann mehrere Produkte liefern. Jedes Personal haben nur ein Abteilungen und zusätzlich eine Abteilungen kann mehrere Personal haben. Jedes Fahrzeugführer haben eigener Fahrzeug. In jedes verkauf nutzen nur ein Kreditkarte Zusätzlich ein Kreditkarte kann mehrere Verkauf nutzen. Eine Kunde kann mehrere Kreditkarte nutzen aber eine Kreditkarte kann nur ein Kunde nutzen wird. Jede Marktkarte haben nur ein Kunde und zusätzlich jede Kunde haben nur eine Marktkarte oder nicht.

# Use Case Diagramm

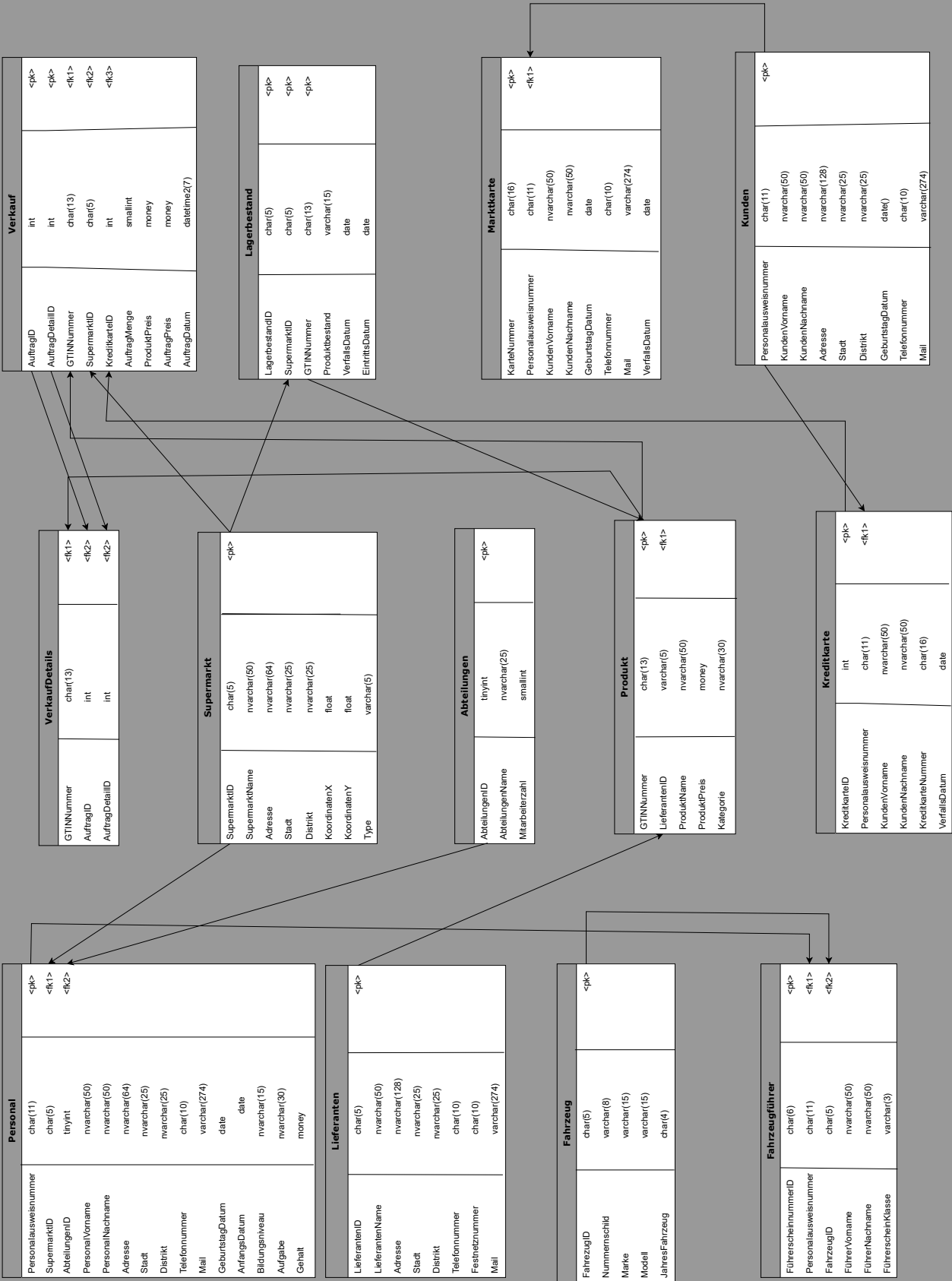
Verkauf Use Case Diagramm



# E-R Diagramm



# Relationales Modell



# SELECT

1)

```
1 SELECT Kunden.KundenVorname,KundenNachname FROM Kunden
2 EXCEPT
3 SELECT Personal.PersonalVorname,PersonalNachname FROM Personal
```

Results	Messages
KundenVorname	KundenNachname
1	Aleyna Tilki
2	Ali Koçak
3	Aylin Güner
4	Beren Saat
5	Canan Özgül
6	Cem Yılmaz
7	Ceylan Erten
8	Dila Armutlu
9	Doğa Denizkaya
10	Ezgi Mola
11	Fatih Terin
12	Gökhan Türkmen
13	Ilkem Erdoğan
14	Kenan Doğulu
15	Mabel Matiz
16	Müge Özgül
17	Nûkhet Duru
18	Selçuk Bayraktar
19	Sıla Gençoğlu
20	Sinige Sağın
21	Sueda Güney
22	Uras Özgül
23	Yasin Çolpan

Diese folgende Abfrage bezieht sich auf unsere Mitarbeiter, die nicht unsere Kunden sind.

```
SELECT Kunden.KundenVorname, KundenNachname FROM Kunden
EXCEPT
SELECT Personal.PersonalVorname,PersonalNachname FROM Personal
```

2)

```

1 SELECT Personalausweisnummer,PersonalVorname,PersonalNachname,Bildungsniveau FROM Personal
2 WHERE Bildungsniveau='Hochschule'
3 ORDER BY PersonalVorname

```

Results

Messages

	Personalausweisnummer	PersonalVorname	PersonalNachname	Bildungsniveau
1	14569703567	Acun	Ilıcalı	Hochschule
2	14777381996	Aslı	Ünlü	Hochschule
3	23435465346	Barış	Manço	Hochschule
4	38129323545	Ebru	Gündeş	Hochschule
5	12345678954	Efe	Uygaç	Hochschule
6	29346765421	Enis	Arikan	Hochschule
7	82345654323	Kenan	İmirzalıoğlu	Hochschule
8	12345646534	Mustafa Emre	Yıldız	Hochschule
9	34646432345	Orkun	İşitmak	Hochschule
10	9246465345	Serenay	Sarıkaya	Hochschule
11	12934203432	Steve	Jobs	Hochschule
12	44695853138	Yiğit	Ergün	Hochschule

Die folgende Abfrage zeigt dass, die Personen wer von Hochschule graduieren hat.

```
SELECT Personalausweisnummer, PersonalVorname,
PersonalNachname, Bildungsniveau FROM Personal
WHERE Bildungsniveau= 'Hochschule'
ORDER BY PersonalVorname
```

3) Die folgende Anfrage zeigt Mitarbeiter, die im Jahr 2001 geboren wurden und sortieren mit GeburtstagDatum.

```
SELECT Personalausweisnummer,PersonalVorname,PersonalNachname,GeburtstagDatum FROM Personal
WHERE GeburtstagDatum like '2001%'
ORDER BY GeburtstagDatum
```

```

1 SELECT Personalausweisnummer,PersonalVorname,PersonalNachname,GeburtstagDatum FROM Personal
2 WHERE GeburtstagDatum like '2001%'
3 ORDER BY GeburtstagDatum

```

Results
Messages

	Personalausweisnummer	PersonalVorname	PersonalNachname	GeburtstagDatum
1	13444555326	Melis	Çiçekli	2001-02-01
2	44695853138	Yiğit	Ergün	2001-02-21
3	23456789654	Onur	Güler	2001-09-22
4	19353657432	Onurcan	Ceyhan	2001-10-10
5	14777381996	Aslı	Ünlü	2001-10-22
6	13256789054	Süleyman	Karagöz	2001-11-10

- 4) Die folgende Abfrage zeigt dass, Summe die Verkäufe per Tag und sie haben mit AuftragDatum gruppieren.

```
SELECT SUM(AuftragPreis) AS Verkauf_per_Tag , AuftragDatum FROM Verkauf
GROUP BY Verkauf.AuftragDatum
```

```
1 SELECT SUM(AuftragPreis) AS Verkauf_per_Tag,AuftragDatum FROM Verkauf
2 GROUP BY Verkauf.AuftragDatum
```

Results Messages

	Verkauf_per_Tag	AuftragDatum
1	520,84	2022-09-23 00:00:00.00000...
2	556,86	2022-09-24 00:00:00.00000...
3	407,89	2022-09-25 00:00:00.00000...
4	377,89	2022-09-26 00:00:00.00000...
5	332,92	2022-09-27 00:00:00.00000...
6	215,94	2022-09-28 00:00:00.00000...

5)

```
1 SELECT COUNT(*) AS Verkaufzahl_per_day FROM Verkauf V
2 GROUP BY AuftragDatum HAVING COUNT(*) >= 5
```

Results Messages

	Verkaufzahl_per_day
1	8
2	6
3	5
4	5

Die folgende Abfrage zeigt Tage mit mehr als 5 Verkäufen pro Tag.

```
SELECT COUNT(*) AS Verkaufzahl_per_day FROM Verkauf V
GROUP BY AuftragDatum HAVING COUNT(*) >= 5
```

- 6) Die folgende Abfrage zeigt, dass Verkauf wo AuftragID gleich 15 ist.

```
SELECT AuftragID, GTINNummer, SupermarktID, ProduktPreis, AuftragMenge, AuftragPreis FROM Verkauf
WHERE EXISTS (SELECT Produkt.ProduktName FROM Produkt WHERE Produkt.GTINNummer=Verkauf.GTINNummer
AND AuftragID=15)
```

```
1 SELECT AuftragID,GTINNummer,SupermarktID,ProduktPreis,AuftragMenge,AuftragPreis FROM Verkauf
2 WHERE EXISTS (SELECT Produkt.ProduktName FROM Produkt WHERE Produkt.GTINNummer=Verkauf.GTINNummer AND AuftragID=15)
```

Results Messages

	AuftragID	GTINNummer	SupermarktID	ProduktPreis	AuftragMenge	AuftragPreis
1	15	8690629440374	01001	55,99	2	111,98

7)

```
1 SELECT Stadt, Distrikt FROM Kunden
2 UNION
3 SELECT Stadt , Distrikt FROM Personal
```

Results Messages

	Stadt	Distrikt
1	Adana	Çukurova
2	Ankara	Bahçelievler
3	Antalya	Belek
4	Eskişehir	Odunpazarı
5	Eskişehir	Tepebaşı
6	İstanbul	Ataşehir
7	İstanbul	Bahçelievler
8	İstanbul	Bakırköy
9	İstanbul	Bebek
10	İstanbul	Beşiktaş

Die folgende Abfrage zeigt die verschiedenen Stadt und Distrikt der Kunden- und Personentabelle.

```
SELECT Stadt, Distrikt FROM Kunden
UNION
SELECT Stadt, Distrikt FROM Personal
```

8)

```
1 SELECT Stadt FROM Lieferanten
2 INTERSECT
3 SELECT Stadt FROM Supermarkt
```

Results Messages

	Stadt
1	Eskişehir
2	İstanbul
3	İzmir
4	Sinop

Die folgende Abfrage zeigt dieselben Städte und Distrikt wie in der Kunden- und Mitarbeitertabelle.

```
SELECT Stadt FROM Lieferanten
INTERSECT
SELECT Stadt FROM Supermarkt
```

9) Die folgende Abfrage zeigt dass, Summer der Mitarbeiterzahl im Unternehmen.

```
SELECT SUM(Mitarbeiterzahl) AS Summe_der_Personal FROM Abteilungen
```

```
1 SELECT SUM(Mitarbeiterzahl) AS Summe_der_Personal FROM Abteilungen
```

Results Messages

	Summe_der_Personal
1	66

10)

Die folgende Abfrage zeigt Fahrzeugführer mit einem Führerschein der Klasse B.

```
SELECT * FROM Fahrzeugführer
WHERE FührerscheinKlasse = ('B')
```

1

SELECT \* FROM Fahrzeugführer

2

WHERE FührerscheinKlasse = ('B')

Results

Messages

	FührerscheinnummerID	Personalausweisnummer	FahrzeugID	FührerVorname	FührerNachname	FührerscheinKlasse
1	123451	32345678765	5	Çağatay	Ulusoy	B
2	123453	12343564667	7	Murat	Boz	B
3	123456	82345654323	1	Kenan	İmirzalıoğlu	B
4	123457	19353657432	2	Onurcan	Ceyhan	B
5	346464	34646432345	11	Orkun	İşıtmak	B
6	432167	43216778543	12	Berkcan	Güven	B

11) Die folgende Abfrage zeigt, dass der Name des Kunden mit a beginnt und eine Kreditkarte hat.

```
SELECT * FROM Kreditkarte
WHERE KundenVorname LIKE 'a%'
```

1

SELECT \* FROM Kreditkarte

2

WHERE KundenVorname LIKE 'a%'

Results

Messages

	KreditKarteID	Personalausweisnummer	KundenVorname	KundenNachname	KreditkarteNummer	VerfallsDatum
1	17	14777381996	Aslı	Ünlü	1234567899876543	09-27
2	22	14569703567	Acun	İlıcılı	3234434543453432	08-26
3	23	14777381996	Aslı	Ünlü	5124400348591023	11-34
4	31	24325686543	Ali	Koçak	2191948483732626	04-25
5	32	24540545454	Aleyna	Tilki	2992129384736281	09-30
6	44	92044549344	Aylin	Güner	2020393948485753	01-29
7	47	14777381996	Aslı	Ünlü	4402039402249283	02-23

12)

Die folgende Abfrage zeigt dass, die Produkte wer Produkte Milchprodukte Kategorie.

```
SELECT * FROM Produkt
WHERE Kategorie = 'Milchprodukte'
```

1SELECT \* FROM Produkt

2WHERE Kategorie = 'Milchprodukte'

Results

Messages

	GTINNummer	LieferantenID	ProduktName	ProduktPreis	Kategorie
1	1234567891233	35001	PINAR ORG.LIGHT SUT 200ML	14,99	Milchprodukte
2	8690767674785	34005	SÜTAŞ AYRAN 150ML	8,99	Milchprodukte
3	8690767716102	34005	SÜTAŞ YOĞURT 2000 GR KAYM...	19,99	Milchprodukte



- 13) Die folgende Abfrage zeigt dass, die Produkte wer Verfallsdatum von Produkte fangen mit 2025 an.

```
SELECT * FROM Lagerbestand
WHERE Verfallsdatum like '2025%'
ORDER BY Verfallsdatum
```

```
1 SELECT * FROM Lagerbestand
2 WHERE Verfallsdatum like '2025%'
3 ORDER BY Verfallsdatum
```

	LagerbestandID	SupermarktID	GTINNummer	Produktbestand	Verfallsdatum	Eintrittsdatum
1	26001	26001	9000100097898	600	2025-01-20	2022-11-20
2	35001	35001	9000100097898	400	2025-01-20	2023-01-04
3	57001	57001	8690526096094	400	2025-04-12	2023-01-02
4	34001	34001	8690565000519	875	2025-05-01	2023-01-03
5	34001	34001	8690632052687	485	2025-05-02	2023-01-03
6	35001	35001	8690767674785	200	2025-05-10	2023-01-02
7	01001	01001	8690767674785	150	2025-06-05	2022-12-15
8	34002	34002	8690527110416	120	2025-06-13	2023-01-13
9	57001	57001	8690632013855	500	2025-08-12	2022-12-30
10	34002	34002	8690624100990	340	2025-08-31	2023-06-13
11	06002	06002	8690624100990	400	2025-09-09	2022-12-31
12	26001	26001	8699264516536	569	2025-09-12	2022-09-20
13	35001	35001	8690624100990	250	2025-09-12	2023-08-20
14	06001	06001	8690632073392	450	2025-10-10	2022-12-12
15	06001	06001	8690527110416	500	2025-12-10	2022-10-12

- 14) Die folgende Anfrage zeigt dass, die Produktliste einer Lieferant in Izmir.

```
SELECT GTINNummer, LieferantenID, ProduktName, ProduktPreis, Kategorie FROM Produkt
WHERE LieferantenID like '35001'
```

```
1 SELECT GTINNummer, LieferantenID, ProduktName, ProduktPreis, Kategorie FROM Produkt
2 WHERE LieferantenID like '35001'
```

	GTINNummer	LieferantenID	ProduktName	ProduktPreis	Kategorie
1	1234567891233	35001	PINAR ORG.LIGHT SUT 200ML	14,99	Milchprodukte
2	8690527110416	35001	PINAR BÖREK TEPŞİ PEYNİRL...	59,99	Fertiggerichte
3	8690527120170	35001	PINAR KÖFTE-ADANA KEBAP 3...	44,99	Fertiggerichte
4	8690565000519	35001	PINAR KETCAP 300 GR	39,99	Sauce

- 15) Die folgende Abfrage zeigt dass, die Produkte wer Produktpreis von Produkte kleiner als 10 und Kategorie von Produkte Junk-food ist.

```
SELECT * FROM Produkt
WHERE ProduktPreis < 10 AND Kategorie = 'Junk-Food'
```

```
1 SELECT * FROM Produkt
2 WHERE ProduktPreis < 10 AND Kategorie = 'Junk-food'
```

	GTINNummer	LieferantenID	ProduktName	ProduktPreis	Kategorie
1	8690504015307	34001	ÜLKER BAHARATLI CUBUK KRA...	7,99	Junk-Food
2	8690526094670	26001	ETİ GONG MISIR	8,99	Junk-Food
3	8690526096094	26001	ETİ TOPKEK PORTAKALLI	5,99	Junk-Food
4	8690632052687	34008	NESTLE CRUNCH CITIR CIKOL...	7,99	Junk-Food

- 16) Die folgende Abfrage zeigt dass, Durchschnitt der Verkauf pro Tag und sie gruppiert mit AuftragDatum.

```
SELECT AVG(AuftragPreis) AS Durchschnitt_der_Verkauf, AuftragDatum AS Datum
FROM Verkauf GROUP BY AuftragDatum
```

```
1 SELECT AVG(AuftragPreis) AS Durchschnitt_der_Verkauf, AuftragDatum AS Datum
2 FROM Verkauf GROUP BY AuftragDatum
```

Results		Messages
	Durchschnitt_der_Verkauf	Datum
1	65,105	2022-09-23 00:00:00.00000...
2	92,81	2022-09-24 00:00:00.00000...
3	81,578	2022-09-25 00:00:00.00000...
4	75,578	2022-09-26 00:00:00.00000...
5	83,23	2022-09-27 00:00:00.00000...
6	107,97	2022-09-28 00:00:00.00000...

- 17) Die folgende Abfrage zeigt dass, welche abteilungen haben größer als 10 mitarbeiter ist.

```
SELECT * FROM Abteilungen
WHERE Mitarbeiterzahl>10
```

```
1 SELECT * FROM Abteilungen
2 WHERE Mitarbeiterzahl>10
```

Results		Messages	
	AbteilungenID	AbteilungenName	Mitarbeiterzahl
1	1	HumanRessourcen	14
2	2	Vertrieb	12

- 18) Die folgende Abfrage zeigt Fahrzeuge, die vor 2016 hergestellt wurden.

```
SELECT * FROM Fahrzeug
```

```
1 SELECT * FROM Fahrzeug
2 WHERE JahresFahrzeug < 2016
```

Results

Messages

	FahrzeugID	Nummernschild	Marke	Modell	JahresFahrzeug
1	4	26ESK260	Renault	Symbol	2014
2	5	01MRK889	Ford	Transit	2005
3	7	06KBT977	Hyundai	Accent	2015
4	9	44BGH765	Renault	Megane	2014
5	10	11BLC443	Toyota	Auris	2012
6	11	39AZ3355	Fiat	Ducato	2013
7	13	28KE6543	Fiat	Panda	2009

19)

```
1 SELECT TOP 10 SUM(AuftragPreis) AS Zehn_teuerste_Auftrag FROM Verkauf
2 GROUP BY AuftragID ORDER BY SUM(AuftragPreis) DESC
```

Results Messages

	Zehn_teuerste_Auftrag
1	199,96
2	179,97
3	179,95
4	167,97
5	159,98
6	111,98
7	111,98
8	109,98
9	109,98
10	107,97

Die folgende Abfrage zeigt dass, Zehn teuerste Auftrag ,sie gruppiert mit AuftragID und sortieren nach AuftragPreis.

```
SELECT TOP 10 SUM(AuftragPreis) AS Zehn_teuerste_Auftrag
FROM Verkauf
GROUP BY AuftragID ORDER BY SUM(AuftragPreis) DESC
```

20)

```
1 SELECT Stadt AS Kunde_Stadt, COUNT(Stadt) AS KundenMenge
2 FROM Kunden
3 GROUP BY Stadt
4 ORDER BY KundenMenge DESC
```

Results Messages

	Kunde_Stadt	KundenMenge
1	İstanbul	15
2	Eskişehir	6
3	Adana	1
4	Ankara	1
5	Antalya	1
6	İzmir	1
7	Kocaeli	1
8	Kütahya	1
9	Muğla	1
10	Sinop	1
11	Sivas	1

Die folgende Anfrage zeigt dass, die Kundenzahl pro Stadt, sie wird gruppiert mit Stadt und sortieren nach Kundenmenge.

```
SELECT Stadt AS Kunde_Stadt, COUNT (Stadt) AS KundenMenge
FROM Kunden
GROUP BY Stadt
ORDER BY KundenMenge DESC
```

21)

Die folgende Abfrage zeigt dass, die Personal, wer hat größer als 10000 Gehalt und sie sortieren nach Gehalt.

```
1 SELECT SupermarktID, PersonalVorname, PersonalNachname, Aufgabe, Gehalt FROM Personal WHERE Gehalt >= 10000 ORDER BY Gehalt
```

Results Messages

	SupermarktID	PersonalVorname	PersonalNachname	Aufgabe	Gehalt
1	35001	Rümeysa	Atlıhan	Leiterin	10000,00
2	41001	Efe	Uygaç	Leiter	12000,00
3	06001	Acun	İlıcılı	Chef	13000,00
4	01001	Serenay	Sarıkaya	Chef	13000,00
5	34003	Sezen	Aksu	Leiterin	14000,00
6	26003	Barış	Manço	Leiter	15000,00
7	48001	Yıldız	Tilbe	Leiter	15000,00
8	58001	Mustafa Emre	Yıldız	IT-Spezialist	15000,00
9	06003	Steve	Jobs	IT-Spezialist	16000,00
10	34001	Aslı	Ünlü	IT-Spezialist	25000,00

```
SELECT SupermarktID,
PersonalVorname, PersonalNachname,
Aufgabe, Gehalt
FROM Personal
WHERE Gehalt >= 10000
ORDER BY Gehalt
```

22)

```
1 SELECT Stadt AS Supermarkt_Stadt, COUNT (Stadt) AS Stadt_Menge
2 FROM Supermarkt
3 GROUP BY Stadt
4 ORDER BY Stadt_Menge DESC
```

Results Messages		
	Supermarkt_Stadt	Stadt_Menge
1	İstanbul	4
2	Ankara	3
3	Eskişehir	3
4	İzmir	2
5	Kocaeli	1
6	Kütahya	1
7	Muğla	1
8	Sinop	1
9	Sivas	1
10	Adana	1
11	Antalya	1

Die folgende Abfrage zeigt dass, die Supermarktzahl pro Städte.

```
SELECT Stadt AS Supermarkt_Stadt,
COUNT (Stadt) AS Stadt_Menge
FROM Supermarkt
GROUP BY Stadt
ORDER BY Stadt_Menge DESC
```

23)

```
1 SELECT Stadt AS Alle_Stadt FROM Personal
2 UNION SELECT Stadt FROM Kunden
```

Results Messages	
	Alle_Stadt
1	Adana
2	Ankara
3	Antalya
4	Eskişehir
5	İstanbul
6	İzmir
7	Kocaeli
8	Kütahya
9	Muğla
10	Sinop
11	Sivas

Die folgende Abfrage zeigt dass, die alle verschiedene Städte in der Personaltable und Kundentable.

```
SELECT Stadt AS Alle_Stadt FROM Personal
UNION SELECT Stadt FROM Kunden
```

# INNER JOIN

1)

Die folgende Abfrage zeigt dass, die Personen, wer hat im Istanbul arbeiten.

```
SELECT Supermarkt.SupermarktID, Supermarkt.SupermarktName, Supermarkt.Adresse, Personalausweisnummer, PersonalVorname, PersonalNachname, TelefonNummer FROM Personal INNER JOIN Supermarkt ON Personal.SupermarktID=Supermarkt.SupermarktID WHERE Personal.SupermarktID like '34%'
```

```
1 SELECT Supermarkt.SupermarktID, Supermarkt.SupermarktName, Supermarkt.Adresse, Personalausweisnummer, PersonalVorname, PersonalNachname, TelefonNummer FROM Personal
2 INNER JOIN Supermarkt
3 ON Personal.SupermarktID=Supermarkt.SupermarktID WHERE Personal.SupermarktID like '34%'
```

Results Messages

	SupermarktID	SupermarktName	Adresse	Personalausweisnummer	PersonalVorname	PersonalNachname	TelefonNummer
1	34001	MOONGROSS Acıbadem	Acıbadem mh. Acıbadem cd...	13435466676	Zeynep	Bastık	5052934905
2	34003	MOONGROSS KemerCountry	Göktürk mh. Kemerburgaz y...	13444555326	Melis	Çiçekli	5436785643
3	34001	MOONGROSS Acıbadem	Acıbadem mh. Acıbadem cd...	14777381996	Aslı	Ünlü	5534705756
4	34004	MOONGROSS Kalamış	Fenerbahçe mh. Cemil Topu...	23456789654	Onur	Güler	5436785432
5	34003	MOONGROSS KemerCountry	Göktürk mh. Kemerburgaz y...	98765432123	Sezen	Aksu	5326781289

2)

```
1 SELECT TOP 10 Supermarkt.SupermarktName, Supermarkt.Adresse, AuftragPreis FROM Verkauf
2 INNER JOIN Supermarkt
3 ON Supermarkt.SupermarktID=Verkauf.SupermarktID WHERE Verkauf.AuftragPreis >=100
4 ORDER BY AuftragPreis DESC
```

Results Messages

	SupermarktName	Adresse	AuftragPreis
1	MOONGROSS Acıbadem	Acıbadem mh. Acıbadem cd...	199,96
2	MOONGROSS Kütahya	Servi mh Mithatpaşa Cd No...	179,97
3	MOONGROSS TepePrime	Mustafa Kemal mh. Dumlupı...	179,95
4	MOONGROSS Acıbadem	Acıbadem mh. Acıbadem cd...	167,97
5	MOONGROSS Altındağ	Birlik mh 4217. Sokak No:4	159,98
6	MOONGROSS Yenışehir	Yenışehir mh. Cumhuriyet ...	111,98
7	MOONGROSS Toros	Toros mh. Kenan Evren Bul...	111,98
8	MOONGROSS EnverBahadır	Ada mh. Enver Bahadır Cd...	109,98
9	MOONGROSS Acıbadem	Acıbadem mh. Acıbadem cd...	109,98
10	MOONGROSS Altındağ	Birlik mh 4217. Sokak No:4	107,97

Die folgende Abfrage zeigt dass, die Top 10 Supermarkt , welche Supermarkt hat größer als 100TL verkaufen und sie sortieren nach Auftragpreis als größer nach kleiner.

```
SELECT TOP 10 Supermarkt.SupermarktName, Supermarkt.Adresse, AuftragPreis FROM Verkauf INNER JOIN Supermarkt ON Supermarkt.SupermarktID=Verkauf.SupermarktID WHERE Verkauf.AuftragPreis >= 100 ORDER BY AuftragPreis DESC
```

3)

Die folgende Abfrage zeigt dass, Lieferanten und Supermarkt hat ähnliche Distrikt ist.

```
SELECT LieferantenID, LieferantenName, SupermarktID, SupermarktName, S.Stadt, S.Distrikt FROM Lieferanten L INNER JOIN Supermarkt S ON L.Distrikt = S.Distrikt WHERE L.Distrikt = S.Distrikt
```

```
1 SELECT LieferantenID, LieferantenName, SupermarktID, SupermarktName, S.Stadt, S.Distrikt FROM Lieferanten L
2 INNER JOIN Supermarkt S
3 ON L.Distrikt = S.Distrikt WHERE L.Distrikt = S.Distrikt
```

Results Messages

	LieferantenID	LieferantenName	SupermarktID	SupermarktName	Stadt	Distrikt
1	34006	PENTİ ÇORAP SAN. VE TİC.A...	06001	MOONGROSS 7.Cadde	Ankara	Bahçelievler
2	26002	GSM İLETİSİM	26002	MOONGROSS Ihlamurkent	Eskişehir	Odunpazarı
3	26001	ETİ Gıda Sanayi ve Ticare...	26003	MOONGROSS Sakarya-2	Eskişehir	Tepebaşı
4	35001	PINAR SÜT MAMULLERİ SANAY...	35002	MOONGROSS Altındağ	İzmir	Bornova

4)

```
1 SELECT PersonalVorname, PersonalNachname, SupermarktID FROM Personal
2 LEFT JOIN Kunden
3 ON Kunden.Personalausweisnummer=Personal.Personalausweisnummer
```

Results Messages

	PersonalVorname	PersonalNachname	SupermarktID
1	Hülya	Avşar	06002
2	Demet	Akalın	01001
3	Murat	Boz	07001
4	Mustafa Emre	Yıldız	58001
5	Efe	Uygaç	41001
6	Seda	Sayan	26002
7	Steve	Jobs	06003
8	Süleyman	Karagöz	43001
9	Zeynep	Bastık	34001
10	Melis	Çiçekli	34003
11	Rümeysa	Atlıhan	35001
12	Ferit	Karakaya	58001
13	Acun	Ilıcalı	06001
14	Aslı	Ünlü	34001
15	Yusuf Efe	Nural	35002
16	Onurcan	Ceyhan	06001
17	Barış	Manço	26003
18	Yıldız	Tilbe	48001
19	Onur	Güler	34004
20	Enis	Arıkan	48001
21	Çağatay	Ulusoy	06001
22	Orkun	Işıtmak	41001
23	Ebru	Gündeş	26001

Die folgende Abfrage zeigt dass, die Personal diesem Unternehmen und gleichzeitig Kunden und Personal von diesem Unternehmen aber nicht nur Kunden von diesem Unternehmen.

```
SELECT PersonalVorname, PersonalNachname, SupermarktID
FROM Personal
LEFT JOIN Kunden
ON Kunden.Personalausweisnummer=
Personal.Personalausweisnummer
```

5)

Die folgende Abfrage zeigt Personen, die sowohl Kunden als auch Mitarbeiter dieses Unternehmens sind.

```
SELECT PersonalVorname, PersonalNachname, SupermarktID FROM Personal
INNER JOIN Kunden
ON Kunden.Personalausweisnummer = Personal.Personalausweisnummer
```

```
1 SELECT PersonalVorname, PersonalNachname, SupermarktID FROM Personal
2 INNER JOIN Kunden
3 ON Kunden.Personalausweisnummer=Personal.Personalausweisnummer
4
```

Results Messages

	PersonalVorname	PersonalNachname	SupermarktID
1	Hülya	Avşar	06002
2	Melis	Çiçekli	34003
3	Acun	Ilıcalı	06001
4	Aslı	Ünlü	34001
5	Barış	Manço	26003
6	Serenay	Sarıkaya	01001



6) Die folgende Abfrage zeigt dass, die Fahrzeugführer, wer die Marke von Fahrzeug Ford ist.

```
SELECT Nummernschild, Marke, Modell, FührerscheinnummerID, FührerVorname, FührerNachname
FROM Fahrzeug F
INNER JOIN Fahrzeugführer FF
ON F.FahrzeugID=FF.FahrzeugID WHERE Marke='Ford'
```

1

2

3

4

```
SELECT Nummernschild, Marke, Modell, FührerscheinnummerID, FührerVorname, FührerNachname
FROM Fahrzeug F
INNER JOIN Fahrzeugführer FF
ON F.FahrzeugID=FF.FahrzeugID WHERE Marke='Ford'
```

Results

Messages

	Nummernschild	Marke	Modell	FührerscheinnummerID	FührerVorname	FührerNachname
1	01MRK889	Ford	Transit	123451	Çağatay	Ulusoy
2	06AB2899	Ford	Focus	123452	Onur	Güler
3	34AS2670	Ford	Fiesta	123457	Onurcan	Ceyhan
4	23ALT478	Ford	Transit	432167	Berkcan	Güven

7) Die folgende Anfrage zeigt dass, die Verkäufe, die am 2022-09-24 sein und sie verkaufte mit Kreditkarte.

```
SELECT Verkauf.KreditKarteID, KundenVorname, KundenNachname, SupermarktID, Verkauf.AuftragPreis, Verkauf.AuftragDatum
FROM Kreditkarte
INNER JOIN Verkauf ON Kreditkarte.KreditkarteID=Verkauf.KreditKarteID WHERE Verkauf.AuftragDatum = '2022-09-24'
```

1

2

```
SELECT Verkauf.KreditKarteID, KundenVorname, KundenNachname, SupermarktID, Verkauf.AuftragPreis, Verkauf.AuftragDatum FROM Kreditkarte
INNER JOIN Verkauf ON Kreditkarte.KreditkarteID=Verkauf.KreditKarteID WHERE Verkauf.AuftragDatum = '2022-09-24'
```

Results

Messages

	KreditKarteID	KundenVorname	KundenNachname	SupermarktID	AuftragPreis	AuftragDatum
1	33	Gökhan	Türkmen	41001	50,97	2022-09-24 00:00:00.00000...
2	33	Gökhan	Türkmen	41001	49,98	2022-09-24 00:00:00.00000...
3	45	Serenay	Sarıkaya	35002	107,97	2022-09-24 00:00:00.00000...
4	45	Serenay	Sarıkaya	35002	159,98	2022-09-24 00:00:00.00000...
5	45	Serenay	Sarıkaya	35002	7,99	2022-09-24 00:00:00.00000...
6	17	Aslı	Ünlü	43001	179,97	2022-09-24 00:00:00.00000...

8) Die folgende Abfrage zeigt dass, die Produkte, die bestand von Produkte gleich or kleiner als 200 ist.

```
SELECT LagerbestandID, SupermarktID, P.ProduktName, Produktbestand
FROM Lagerbestand L
INNER JOIN Produkt P
ON P.GTINNummer= L.GTINNummer
WHERE L.Produktbestand <= 200
```

1

2

3

4

5

```
SELECT LagerbestandID, SupermarktID, P.ProduktName, Produktbestand FROM Lagerbestand L
INNER JOIN Produkt P
ON P.GTINNummer=L.GTINNummer
WHERE L.Produktbestand <= 200
```

Results

Messages

	LagerbestandID	SupermarktID	ProduktName	Produktbestand
1	01001	01001	SÜTAŞ AYRAN 150ML	150
2	01001	01001	PERSİL MATİK 6KG DENİZ ES...	100
3	26001	26001	PENTİ YAZ SICAKLIK 57 - 3...	152
4	34002	34002	PINAR BÖREK TEPŞİ PEYNİRL...	120
5	35001	35001	ETİ TOPKEK PORTAKALLI	200
6	35001	35001	SÜTAŞ AYRAN 150ML	200
7	57001	57001	PINAR BÖREK TEPŞİ PEYNİRL...	100

9) Die folgende Abfrage zeigt dass, die Personal , wer nur Fahrzeugführer ist.

```
SELECT PersonalVorname, PersonalNachname, Personal.Personal FROM Personal  
RIGHT JOIN Fahrzeugführer  
ON Personal.Personalausweisnummer=Fahrzeugführer.Personalausweisnummer
```

1 SELECT PersonalVorname,PersonalNachname,Personal.Personalausweisnummer FROM Personal

2 RIGHT JOIN Fahrzeugführer

3 ON Personal.Personalausweisnummer=Fahrzeugführer.Personalausweisnummer

Results Messages

	PersonalVorname	PersonalNachname	Personalausweisnummer
1	Çağatay	Ulusoy	32345678765
2	Onur	Güler	23456789654
3	Murat	Boz	12343564667
4	Hasan Can	Kaya	83456774567
5	Kenan	İmirzalıoğlu	82345654323
6	Onurcan	Ceyhan	19353657432
7	Ali	Ağaoğlu	45456754321
8	Yiğit	Ergün	44695853138
9	Enis	Arıkan	29346765421
10	Orkun	İşıtmak	34646432345
11	Berkcan	Güven	43216778543

10) Die folgende Abfrage zeigt die 10 meistverkauften Produkte und sortiert sie nach Anzahl der Bestellungen.

```
SELECT TOP 10  ProduktName, AuftragMenge  
FROM Verkauf  
INNER JOIN Produkt  
ON Verkauf.GTINNummer=Produkt.GTINNummer  
WHERE Verkauf.GTINNummer = Produkt.GTINNummer  
ORDER BY AuftragMenge DESC
```

```

1 SELECT TOP 10 ProduktName,AuftragMenge
2 FROM Verkauf
3 INNER JOIN Produkt
4 ON Verkauf.GTINNummer=Produkt.GTINNummer WHERE Verkauf.GTINNummer = Produkt.GTINNummer ORDER BY AuftragMenge DESC

```

Results

Messages

	ProduktName	AuftragMenge
1	DR.OUTKER PİZZA GUSEPPE 4...	10
2	ÜLKER BİZİM AYÇİÇEK 18L*1	5
3	NESTLE 900 GR CHOKELLA	4
4	NESTLE CRUNCH CITIR CIKOL...	4
5	SÜTAŞ AYRAN 150ML	3
6	NESTLE KARE CIKOLATA BLAC...	3
7	ÜLKER BİZİM AYÇİÇEK 18L*1	3
8	PINAR BÖREK TEPŞİ PEYNİRL...	3
9	NESTLE COFFEE MATE 625GR...	3
10	LAYS BAHARAT ÇEŞNİLİ	3



# INSERT INTO

- 1) Die folgende Abfrage zeigt dass, von Personaltable auf Kundentable ein Data einfügen.

```
INSERT INTO Kunden (Kunden.Personalausweisnummer, KundenVorname, KundenNachname, TelefonNummer, Stadt, Distrikt)
SELECT Personal.Personalsuweisnummer, Personal.PersonalVorname, Personal.PersonalNachname, Personal.Telefonnummer,
Personal.Stadt, Personal.Distrikt
FROM Personal WHERE Personal.Personalausweisnummer= '44695853138'
```

```
1 INSERT INTO Kunden (Kunden.Personalausweisnummer,KundenVorname,KundenNachname,TelefonNummer,Stadt,Distrikt)
2 SELECT Personal.Personalausweisnummer,Personal.PersonalVorname,Personal.PersonalNachname,Personal.Telefonnummer,Personal.Stadt,Personal.Distrikt
3 FROM Personal WHERE Personal.Personalausweisnummer='44695853138'
```

## Messages

17:47:40      Started executing query at Line 1  
(1 row affected)  
Total execution time: 00:00:00.021

- 2) Die folgende Abfrage zeigt dass, von Kundentable auf Lieferantentable ein Data einfügen.

```
INSERT INTO Lieferanten (LieferantenID,Stadt,Distrikt,LieferantenName,Telefonnummer,Adresse,Mail)
SELECT '26002',Kunden.Stadt,Kunden.Distrikt,'GSM İLETİŞİM', Kunden.TelefonNummer, Kunden.Adresse, Kunden.Mail
FROM Kunden WHERE Kunden.KundenVorname='Müge'
```

```
1 INSERT INTO Lieferanten (LieferantenID,Stadt,Distrikt,LieferantenName,Telefonnummer,Adresse,Mail)
2 SELECT '26002',Kunden.Stadt,Kunden.Distrikt,'GSM İLETİŞİM',Kunden.TelefonNummer,Kunden.Adresse,Kunden.Mail FROM Kunden
3 WHERE Kunden.KundenVorname='Müge'
```

## Messages

19:23:26      Started executing query at Line 1  
(1 row affected)  
Total execution time: 00:00:00.017

- 3) Die folgende Abfrage zeigt, wie Daten zum Abteilungentable hinzugefügt werden.

```
INSERT INTO Abteilungen (AbteilungenName, Mitarbeiterzahl)
VALUES ('HumanRessourcen', 14)
```

```
1 INSERT INTO Abteilungen (AbteilungenName, Mitarbeiterzahl)
2 VALUES ( 'HumanRessourcen', 14)
```

## Messages

15:14:21      Started executing query at Line 1  
(1 row affected)  
Total execution time: 00:00:00.037

- 4) Die folgende Abfrage zeigt, wie Daten zum Lieferantentable hinzugefügt werden.

```
INSERT INTO Lieferanten (LieferantenID, LieferantenName, Adresse, Stadt, Distrikt, Mail)
VALUES ('34001', 'Ülker Çikolata San. ve Tic. A.Ş', 'Maltepe Mah. Davutpaşa Caddesi
Nummer:10', 'İstanbul', 'Fatih', 'ulkercikolata@gmail.com')
```

```
1  INSERT INTO Lieferanten (LieferantenID, LieferantenName, Adresse, Stadt, Distrikt, Mail)
2  VALUES ('34001', 'Ülker Çikolata San. ve Tic. A.Ş', 'Maltepe Mah. Davutpaşa Caddesi Nummer:10', 'İstanbul', 'Fatih', 'ulkercikolata@gmail.com')
```

#### Messages

17:47:40      Started executing query at Line 1  
(1 row affected)  
Total execution time: 00:00:00.021

- 5) Die folgende Abfrage zeigt dass, wie Daten zum Fahrzeugtable hinzugefügt werden.

```
INSERT INTO Fahrzeug (Nummernschild, Marke, Modell, JahresFahrzeug)
VALUES ('34AB2806', 'Fiat', 'Ducato', '2019')
```

```
1  INSERT INTO Fahrzeug (Nummernschild, Marke, Modell, JahresFahrzeug)
2  VALUES ('34AB2806', 'Fiat', 'Ducato', '2019')
```

#### Messages

00:36:41      Started executing query at Line 1  
(1 row affected)  
Total execution time: 00:00:00.040

## UPDATE

- 1) Die folgende Abfrage zeigt dass, 10.99 TL zu die alte Produktpreis von Produkte hinzugefügt.

UPDATE Produkt

SET ProduktPreis = ProduktPreis + 10.99

```
1  UPDATE Produkt
2  SET ProduktPreis = ProduktPreis + 10.99
```

### Messages

```
20:17:22      Started executing query at Line 1
              (40 rows affected)
              Total execution time: 00:00:00.018
```

- 2) Die folgende Abfrage zeigt dass, die Supermarkt, hat die Supermarktnamen MOONGROSS TepePrime ist. Werden die neue Type von Supermarkt MMM ist.

UPDATE Supermarkt

SET [Type] = 'MMM' WHERE SupermarktName= 'MOONGROSS TepePrime'

```
1  UPDATE Supermarkt
2  SET [Type]='MMM' WHERE SupermarktName='MOONGROSS TepePrime'
```

### Messages

```
20:35:34      Started executing query at Line 1
              (1 row affected)
              Total execution time: 00:00:00.032
```

3) Die folgende Abfrage zeigt dass, die Neue Telefonnummer von Personal.

UPDATE Personal

SET Telefonnummer = '5536789203' WHERE Telefonnummer= '5019343545'

```
1 UPDATE Personal
```

```
2 SET Telefonnummer='5536789203' WHERE Telefonnummer='5019343545'
```

#### Messages

```
20:30:54 Started executing query at Line 1
(1 row affected)
Total execution time: 00:00:00.016
```

4) Die folgende Abfrage macht dass, den Produktbestand 200 in allen Lagern für das Produkt.

UPDATE Lagerbestand

SET Produktbestand = 200 WHERE GTINNummer= '9000100097898'

```
1 UPDATE Lagerbestand
```

```
2 SET Produktbestand=200 WHERE GTINNummer='9000100097898'
```

#### Results Messages

```
20:27:52 Started executing query at Line 3
(3 rows affected)
Total execution time: 00:00:00.030
```

5) Die folgende Abfrage zeigt dass, die Lieferanten, hat Lieferanten ID 26002 ist die neue LieferantenID von diesem Unternehmen 57001 ist.

UPDATE Lieferanten

SET Stadt = 'Sinop', LieferantenID= '57001' WHERE LieferantenID= '26002'

```
1 UPDATE Lieferanten
```

```
2 SET Stadt='Sinop',LieferantenID='57001' WHERE LieferantenID = '26002'
```

#### Messages

```
20:22:55 Started executing query at Line 1
(1 row affected)
Total execution time: 00:00:00.025
```

## DELETE

1) 1 `DELETE FROM Produkt`

2 `WHERE GTINNummer='8695895101332'`

Die folgende Anfrage zeigt dass, die Produkt, das GTIN Nummer wird gesehen hat gelöscht werden.

### Messages

21:15:52

Started executing query at Line 1  
(1 row affected)

Total execution time: 00:00:00.036

`DELETE FROM Produkt`

`WHERE GTINNummer='8695895101332'`

2)

1 `DELETE Marktkarte`

2 `WHERE Personalausweisnummer='23435465346'`

Die folgende Abfrage zeigt dass, die Person , wer Marktkarte haben wird gelöscht.

### Messages

21:17:26

Started executing query at Line 1  
(1 row affected)

Total execution time: 00:00:00.037

`DELETE Marktkarte`

`WHERE`

`Personalausweisnummer='2345465346'`

3)

Die folgende Anfrage zeigt dass, die Person wird gelöscht.

`DELETE Personal`

`WHERE Personalausweisnummer='98765432123'`

1 `DELETE Personal`

2 `WHERE Personalausweisnummer='98765432123'`

### Messages

21:22:00

Started executing query at Line 1  
(1 row affected)

Total execution time: 00:00:00.015

- 4) Die folgende Abfrage zeigt dass, die Lieferant, diesem LieferantenID hat wird gelöscht werden.

```
DELETE Lieferanten  
WHERE LieferantenID='58002'
```

```
1  DELETE Lieferanten  
2  WHERE LieferantenID='58002'
```

#### Messages

```
21:24:11      Started executing query at Line 1  
              (1 row affected)  
              Total execution time: 00:00:00.021
```

- 5) Die folgende Abfrage zeigt dass, die Fahrzeug , wer diese FahrzeugID hat wird gelöscht.

```
DELETE Fahrzeug  
WHERE FahrzeugID='1002'
```

```
1  DELETE Fahrzeug  
2  WHERE FahrzeugID='1002'
```

#### Messages

```
21:25:31      Started executing query at Line 1  
              (1 row affected)  
              Total execution time: 00:00:00.019
```

## ALTER TABLE

- 1) Die folgende Abfrage zeigt dass, das Datatype von Spalte Kundenvorname hat geändert werden.

```
1 ALTER TABLE Kunden
2 ALTER COLUMN KundenVorname nvarchar(30)
```

### Messages

21:30:36      Started executing query at Line 1  
Commands completed successfully.  
Total execution time: 00:00:00.023

ALTER TABLE Kunden  
ALTER COLUMN  
KundenVorname nvarchar(30)

- 2) Die folgende Abfrage zeigt dass, das Datatype von Spalte Kundennachname hat geändert werden.

```
1 ALTER TABLE Kunden
2 ALTER COLUMN Kundennachname nvarchar(30)
```

### Messages

21:30:36      Started executing query at Line 1  
Commands completed successfully.  
Total execution time: 00:00:00.023

ALTER TABLE Kunden  
ALTER COLUMN  
Kundennachname nvarchar(30)

- 3) Die folgende Abfrage zeigt dass, das Spalte , Spaltenname Jahresfahrzeug von Spalte ist wird gelöscht.

```
1 ALTER TABLE Fahrzeug
2 DROP COLUMN JahresFahrzeug
```

### Messages

21:35:04      Started executing query at Line 1  
Commands completed successfully.  
Total execution time: 00:00:00.015

ALTER TABLE Fahrzeug  
DROP COLUMN JahresFahrzeug

- 4) Die folgende Abfrage zeigt dass, die Spalte , Spaltenname Jahresfahrzeug ist mit char datatype wird hinzugefügt.

```
ALTER TABLE Fahrzeug
ADD JahresFahrzeug char(4)
```

```
1  ALTER TABLE Fahrzeug
2  ADD JahresFahrzeug char(4)
```

#### Messages

21:37:56	<u>Started executing query at Line 1</u> Commands completed successfully. Total execution time: 00:00:00.018
----------	--

- 5) Die folgende Abfrage zeigt dass, die Spalte , das Spaltenname von Spalte Verfallsdatum ist wird gelöscht.

```
ALTER TABLE Produkt
DROP COLUMN VerfallsDatum
```

```
1  ALTER TABLE Produkt
2  DROP COLUMN VerfallsDatum
```

#### Messages

21:47:50	<u>Started executing query at Line 1</u> Commands completed successfully. Total execution time: 00:00:00.014
----------	--



## DROP TABLE

- 1) Die folgende Abfrage zeigt dass, die Tabelle gelöscht werden.

DROP TABLE Fahrzeug

---

```
1  DROP TABLE Fahrzeug
```

---

### Messages

```
03:31:24      Started executing query at Line 1
              Commands completed successfully.
              Total execution time: 00:00:00.018
```

- 2) Die folgende Abfrage zeigt dass, die Tabelle gelöscht werden.

DROP TABLE Supermarkt

---

```
1  DROP TABLE Supermarkt
```

---

### Messages

```
03:33:32      Started executing query at Line 1
              Commands completed successfully.
              Total execution time: 00:00:00.013
```

# CREATE VIEW

- 1) Die folgende Abfrage zeigt Produkte, die mit Eti beginnen. Aber in diesem Schritt werden wir nur erstellen dann wir werden verlaufen.

```
CREATE VIEW [Die_Produkte_anfangen-mit_ETI] AS SELECT GTINNummer, ProduktName, Kategorie
FROM Produkt WHERE ProduktName like N'Eti%';
```

```
1 CREATE VIEW [Die_Produkte_anfangen-mit_ETI] AS SELECT GTINNummer, ProduktName, Kategorie
2 FROM Produkt WHERE ProduktName like N'Eti%';
```

## Messages

20:07:14      Started executing query at Line 1  
Commands completed successfully.  
Total execution time: 00:00:00.020

- 2) Die folgende Abfrage zeigt Produkte, die Produkte in Junkfood Kategorie ist aber in diesem Schritt werden wir nur erstellen dann wir werden verlaufen.

```
CREATE VIEW [Junkfood_Produkte] AS SELECT GTINNummer, ProduktName, Kategorie FROM Produkt
WHERE Kategorie='Junk_food';
```

```
1 CREATE VIEW [Junkfood_Produkte] AS SELECT GTINNummer, ProduktName, Kategorie FROM Produkt WHERE Kategorie='Junk-food';
```

## Messages

02:36:05      Started executing query at Line 1  
Commands completed successfully.  
Total execution time: 00:00:00.016

- 3) Die folgende Abfrage zeigt Produkte, die Produktpreis von Produkt kleiner als 30TL aber in diesem Schritt werden wir nur erstellen dann wir werden verlaufen.

```
CREATE VIEW [ProduktePreis_als_unter_30tl] AS SELECT GTINNummer , LieferantenID , ProduktName, ProduktPreis
FROM Produkt WHERE ProduktPreis<30
```

```
1 CREATE VIEW [ProduktePreis_als_unter_30tl] AS SELECT GTINNummer , LieferantenID , ProduktName, ProduktPreis
2 FROM Produkt WHERE ProduktPreis<30
```

## Messages

20:02:56      Started executing query at Line 1  
Commands completed successfully.  
Total execution time: 00:00:00.019

- 4) Die folgende Abfrage zeigt dass, die alle Supermarkt, welche in Istanbul ist aber in diesem Schritt nur erstellen dann werden wir verlaufen.

```
CREATE VIEW [Supermarkt_in_Istanbul] AS SELECT SupermarktID , SupermarktName,Adresse,Stadt,Distrikt
FROM Supermarkt WHERE Stadt='istanbul';
```

```
1 CREATE VIEW [Supermarkt_in_Istanbul] AS SELECT SupermarktID,SupermarktName,Adresse,Stadt,Distrikt
2 FROM Supermarkt WHERE Stadt='istanbul';
```

#### Messages

20:01:52      Started executing query at Line 1  
Commands completed successfully.  
Total execution time: 00:00:00.063

- 5) Die folgende Abfrage zeigt dass, wer mit einen Kreditkarte bezahlen aber in diesem Schritt nur erstellen dann werden wir verlaufen.

```
CREATE VIEW [Mit_KreditKarte_Zahlung] AS SELECT AuftragID, AuftragDetailID, GTINNummer, KreditkarteID
FROM Verkauf WHERE Barzahlung='no';
```

```
1 CREATE VIEW [Mit_KreditKarte_Zahlung] AS SELECT AuftragID,AuftagDetailID,GTINNummer,KreditkarteID FROM Verkauf
2 WHERE Barzahlung ='no';
```

#### Messages

19:59:49      Started executing query at Line 1  
Commands completed successfully.  
Total execution time: 00:00:00.021

- 6) Die folgende Abfrage zeigt dass, welche Produktbestand von Produkt kleiner als 300 aber in diesem Schritt nur erstellen dann werden wir verlaufen.

```
CREATE VIEW [Die_Lagerbestand_der_Produkt] AS SELECT GTINNummer, Produktbestand
FROM Lagerbestand WHERE Produktbestand < 300
```

```
1 CREATE VIEW [Die_Lagerbestand_der_Produkt] AS SELECT GTINNummer, Produktbestand
2 FROM Lagerbestand WHERE Produktbestand < 300
```

#### Messages

20:08:01      Started executing query at Line 1  
Commands completed successfully.  
Total execution time: 00:00:00.021

- 7) Die folgende Abfrage zeigt dass, die Kunden, die in Eskişehir wohnen aber in diesem Schritt nur erstellen dann werden wir verlaufen.

```
CREATE VIEW [Kunden_die_in_Eskişehir_Wohnen] AS SELECT KundenVorname,KundenNachname,Adresse,Stadt,Distrikt
FROM Kunden WHERE Stadt = N'Eskişehir';
```

```
1 CREATE VIEW [Kunden_die_in_Eskişehir_Wohnen] AS SELECT KundenVorname,KundenNachname,Adresse,Stadt,Distrikt FROM Kunden WHERE Stadt = N'Eskişehir';
```

#### Messages

02:42:08 Started executing query at Line 1  
Commands completed successfully.  
Total execution time: 00:00:00.016

- 8) Die folgende Abfrage zeigt dass, die Produkte, die unterdurchschnittlichem Preis ist aber in diesem Schritt nur erstellen dann werden wir verlaufen.

```
CREATE VIEW [Produkte_mit_Unterdurchschnittlichem_Preis]
AS SELECT ProduktName, GTINNummer, Kategorie, ProduktPreis
FROM Produkt WHERE ProduktPreis < (SELECT AVG(ProduktPreis) FROM Produkt)
```

```
1 CREATE VIEW [Produkte_mit_Unterdurchschnittlichem_Preis]
2 AS SELECT ProduktName,GTINNummer,Kategorie,ProduktPreis
3 FROM Produkt WHERE ProduktPreis < (SELECT AVG(ProduktPreis) FROM Produkt)
```

#### Messages

02:52:32 Started executing query at Line 1  
Commands completed successfully.  
Total execution time: 00:00:00.015

- 9) Die folgende Abfrage zeigt dass, die Fahrzeug, die Marke von Fahrzeug Fiat ist aber in diesem Schritt nur erstellen dann werden wir verlaufen.

```
CREATE VIEW [Fahrzeug_die_Fiat_ist] AS SELECT * FROM Fahrzeug WHERE Marke='fiat';
```

```
1 CREATE VIEW [Fahrzeug_die_Fiat_ist] AS SELECT * FROM Fahrzeug WHERE Marke='fiat';
```

#### Messages

20:15:04 Started executing query at Line 1  
Commands completed successfully.  
Total execution time: 00:00:00.019

- 10) Die folgende Abfrage zeigt dass, die Fahrzeug, die stellen nach 2015 her aber in diesem Schritt nur erstellen dann werden wir verlaufen.

```
CREATE VIEW [Jahres_von_Fahrzeug_größer_als_2015] AS SELECT * FROM Fahrzeug WHERE Jahresfahrzeug >= 2015;
```

```
1 CREATE VIEW [Jahres_von_Fahrzeug_größer_als_2015] AS SELECT * FROM Fahrzeug WHERE Jahresfahrzeug >= 2015;
```

#### Messages

20:15:27 Started executing query at Line 1  
Commands completed successfully.  
Total execution time: 00:00:00.019

# SELECT FROM VIEW

- 1) Die folgende Abfrage zeigt dass, wer mit eine Kreditkarte bezahlen.

**SELECT \* FROM** Mit\_KreditKarte\_Zahlung

1 **SELECT \* FROM** Mit\_KreditKarte\_Zahlung

**Results** Messages

	AuftragID ▾	AuftragDetailID ▾	GTINNumber ▾	KreditkarteID ▾
1	15	1	8690629440374	22
2	16	2	8690632073392	22
3	17	1	9000100097898	23
4	18	2	8690632052687	23
5	19	3	8690624100990	23
6	20	1	8690767674785	26
7	21	2	9000100944632	26
8	22	3	9000100097898	26
9	23	1	8690632481166	33
10	24	2	8690698509149	33
11	27	1	8691375623103	45
12	28	2	8690637699443	45
13	29	3	8690632052687	45
14	30	1	8690527110416	17

- 2) Die folgende Abfrage zeigt dass, die Supermarkte, die in Istanbul ist.

**SELECT \* FROM** (Supermarkt\_in\_Istanbul)

1 **SELECT \* FROM** [Supermarkt\_in\_Istanbul]

**Results** Messages

	SupermarktID ▾	SupermarktName ▾	Adresse ▾	Stadt ▾	Distrikt ▾
1	34001	MOONGROSS Acıbadem	Acıbadem mh. Acıbadem cd...	İstanbul	Kadıköy
2	34002	MOONGROSS Barbaros	Barbaros mh. Azizboy sk. ...	İstanbul	Ataşehir
3	34003	MOONGROSS KemerCountry	Göktürk mh. Kemerburgaz y...	İstanbul	Eyüp
4	34004	MOONGROSS Kalamış	Fenerbahçe mh. Cemil Topu...	İstanbul	Kadıköy

- 3) Die folgende Abfrage zeigt dass, die Fahrzeug, die Marke von Fahrzeug Fiat ist.

**SELECT \* FROM** Fahrzeug\_die\_Fiat\_ist

1 **SELECT \* FROM** Fahrzeug\_die\_Fiat\_ist

**Results** Messages

	FahrzeugID ▾	Nummernschild ▾	Marke ▾	Modell ▾	JahresFahrzeug ▾
1	1	34AY2803	Fiat	Ducato	2018
2	3	34YGT446	Fiat	Punto	2020
3	11	39AZ3355	Fiat	Ducato	2013
4	13	28KE6543	Fiat	Panda	2009
5	1022	34AB2806	Fiat	Ducato	2019

4) Die folgende Abfrage zeigt dass, die Produkte, die Produktpreis unter 30TL ist.

```
SELECT * FROM ProduktePreis_als_unter_30tl
```

1 SELECT \* FROM ProduktePreis\_als\_unter\_30tl

Results		Messages		
	GTINNummer	LieferantenID	ProduktName	ProduktPreis
1	1234567891233	35001	PINAR ORG.LIGHT SUT 200ML	25,98
2	8690504015307	34001	ÜLKER BAHARATLI CUBUK KRA...	18,98
3	8690526085241	26001	ETİ HOŞBEŞ 44 GR VANILYA	27,98
4	8690526094670	26001	ETİ GONG MISIR	19,98
5	8690526096094	26001	ETİ TOPKEK PORTAKALLI	16,98
6	8690533131139	26001	ETİ CİCİ BEBE 1200 GR	25,98
7	8690574004317	53001	LİPTON İCETEA ŞEFTALİ	24,98
8	8690632052687	34008	NESTLE CRUNCH CITIR CIKOL...	18,98
9	8690632481166	34008	NESTLE KARE CIKOLATA BLAC...	27,98
10	8690767674785	34005	SÜTAŞ AYRAN 150ML	19,98

5) Die folgende Abfrage zeigt dass, die Produkte, die in Junkfood Kategorie ist.

```
SELECT * FROM [Junkfood_Produnkte]
```

1 SELECT \* FROM [Junkfood\_Produnkte]

Results		Messages	
	GTINNummer	ProduktName	Kategorie
1	8690504015307	ÜLKER BAHARATLI CUBUK KRA...	Junk-Food
2	8690526085241	ETİ HOŞBEŞ 44 GR VANILYA	Junk-Food
3	8690526094670	ETİ GONG MISIR	Junk-Food
4	8690526096094	ETİ TOPKEK PORTAKALLI	Junk-Food
5	8690533131139	ETİ CİCİ BEBE 1200 GR	Junk-Food
6	8690624100990	LAYS BAHARAT ÇEŞNİLİ	Junk-Food
7	8690632052687	NESTLE CRUNCH CITIR CIKOL...	Junk-Food
8	8690632073392	NESTLE 900 GR CHOKELLA	Junk-Food
9	8690632481166	NESTLE KARE CIKOLATA BLAC...	Junk-Food



6) Die folgende Abfrage zeigt dass, die Produkte, die Produktname fangen mit Eti an.

SELECT \* FROM Die\_Produnkte\_anfangen\_mit\_ETI

1 SELECT \* FROM Die\_Produnkte\_anfangen\_mit\_ETI

Results		Messages	
	GTINNummer	ProduktName	Kategorie
1	8690526085241	ETİ HOŞBEŞ 44 GR VANILYA	Junk-Food
2	8690526094670	ETİ GONG MISIR	Junk-Food
3	8690526096094	ETİ TOPKEK PORTAKALLI	Junk-Food
4	8690533131139	ETİ CİCİ BEBE 1200 GR	Junk-Food

7)

1 SELECT \* FROM Die\_Lagerbestand\_der\_Produkt

Results		Messages	
	GTINNummer	Produktbestand	
1	8690629440374	222	
2	8690767674785	150	
3	9000100097898	200	
4	8699264485733	152	
5	9000100097898	200	
6	8690527110416	120	
7	8690526096094	200	
8	8690624100990	250	
9	8690767674785	200	
10	9000100097898	200	
11	8690527110416	100	

Die folgende Abfrage zeigt dass, die Produktbestand von Produkt.

SELECT \* FROM Die\_Lagerbestand\_der\_Produkt

8) Die folgende Abfrage zeigt dass, die Kunden, die in Eskişehir wohnen.

SELECT \* FROM [Kunden\_die\_in\_Eskişehir\_Wohnen]

1 SELECT \* FROM [Kunden\_die\_in\_Eskişehir\_Wohnen]

Results		Messages			
	KundenVorname	KundenNachname	Adresse	Stadt	Distrikt
1	Müge	Özgül	Sümer mh. Sapmaz Sk. No:23	Eskişehir	Odunpazarı
2	Sueda	Güney	Arifiye mh. Hamamyolu cd...	Eskişehir	Odunpazarı
3	Canan	Özgül	Güllük mh. Kırımlılar sk...	Eskişehir	Tepebaşı
4	Uras	Özgül	Güllük mh. Kırımlılar sk...	Eskişehir	Tepebaşı
5	Yasin	Çolpan	Ömerağa mh. Cengiz Topel ...	Eskişehir	Tepebaşı
6	Dila	Armutlu	Vişnelik mh. Atatürk Bulv...	Eskişehir	Odunpazarı

- 9) Die folgende Abfrage zeigt dass, die Produkte, die Unterdurchschnittlichem preis ist.

**SELECT \* FROM [Produkte\_mit\_Unterdurchschnittlichem\_Preis]**

1 SELECT \* FROM [Produkte\_mit\_Unterdurchschnittlichem\_Preis]

Results Messages

	ProduktName	GTINNummer	Kategorie	ProduktPreis
1	PINAR ORG.LIGHT SUT 200ML	1234567891233	Milchprodukte	25,98
2	VİVİDENT XYLIT	4345676543456	Bonbon	30,98
3	VİVİDENT ÇİLEKLİ	8079188134565	Bonbon	30,98
4	ÜLKER BAHARATLI CUBUK KRA...	8690504015307	Junk-Food	18,98
5	ETİ HOŞBEŞ 44 GR VANILYA	8690526085241	Junk-Food	27,98
6	ETİ GONG MISIR	8690526094670	Junk-Food	19,98
7	ETİ TOPKEK PORTAKALLI	8690526096094	Junk-Food	16,98
8	ETİ CİCİ BEBE 1200 GR	8690533131139	Junk-Food	25,98
9	LİPTON İCETEA ŞEFTALİ	8690574004317	Getränke	24,98
10	LAYS BAHARAT ÇEŞNİLİ	8690624100990	Junk-Food	30,98
11	NESTLE COFFEE MATE 625GR...	8690632013855	Getränke	40,98
12	NESTLE CRUNCH CITIR CIKOL...	8690632052687	Junk-Food	18,98
13	NESTLE KARE CIKOLATA BLAC...	8690632481166	Junk-Food	27,98
14	LİPTON EARL GREY 2GR*100L...	8690637043093	Getränke	40,98
15	UNO FIRINDAN TAM BUGDAY E...	8690698509149	Grundnahrungsmittel	35,98
16	SÜTAŞ AYRAN 150ML	8690767674785	Milchprodukte	19,98
17	SÜTAŞ YOĞURT 2000 GR KAYM...	8690767716102	Milchprodukte	30,98
18	PENTİ FİT DİZ ALTI İNCE 5...	8699094081785	Kleidung	41,98
19	PENTİ YAZ SICAKLIK 57 – 3...	8699264485733	Kleidung	30,98
20	COLGATE DIS FIRCASI MAX W...	8714789417066	Körperpflege	45,98

- 10) Die folgende abfrage zeigt dass, die Fahrzeug, die gleich oder nach 2015 herstellen.

**SELECT \* FROM Jahres\_von\_Fahrzeug\_größer\_als\_2015**

1

SELECT \* FROM Jahres\_von\_Fahrzeug\_größer\_als\_2015

Results

Messages

	FahrzeugID	Nummernschild	Marke	Modell	JahresFahrzeug
1	1	34AY2803	Fiat	Ducato	2018
2	2	34AS2670	Ford	Fiesta	2016
3	3	34YGT446	Fiat	Punto	2020
4	6	06AB2899	Ford	Focus	2022
5	7	06KBT977	Hyundai	Accent	2015
6	8	35KSR556	Renault	Clio	2017
7	12	23ALT478	Ford	Transit	2018
8	1017	26YA2813	Renault	Menage	2022
9	1022	34AB2806	Fiat	Ducato	2019



# INDEX

- 1) Die folgende Abfrage zeigt dass, eine Liste wird erstellen und die Liste zeigen Kundenvorname und Kundennachname in einer Liste.

```
CREATE INDEX KundenVorname_KundenNachname_index  
ON Kunden(KundenVorname,KundenNachname)
```

```
1 CREATE INDEX KundenVorname_KundenNachname_index ON Kunden(KundenVorname,KundenNachname)
```

## Messages

```
20:16:17 Started executing query at Line 1  
Commands completed successfully.  
Total execution time: 00:00:00.035
```

- 2) Die folgende Abfrage zeigt dass, unique index wird erstellen.

```
CREATE UNIQUE INDEX LieferantenName_index ON Lieferanten(LieferantenName);
```

```
1 CREATE UNIQUE INDEX LieferantenName_index ON Lieferanten(LieferantenName);
```

## Messages

```
20:19:54 Started executing query at Line 1  
Commands completed successfully.  
Total execution time: 00:00:00.025
```

- 3) Die folgende Abfrage zeigt dass, unique index wird erstellen.

```
CREATE UNIQUE INDEX ProduktName_index ON Produkt(ProduktName);
```

```
1 CREATE UNIQUE INDEX ProduktName_index ON Produkt(ProduktName)
```

## Messages

```
20:19:30 Started executing query at Line 1  
Commands completed successfully.  
Total execution time: 00:00:00.024
```

- 4) Die folgende Abfrage zeigt dass, unique index wird erstellen.

```
CREATE UNIQUE INDEX Supermarkt_index ON Supermarkt(SupermarktName);
```

```
1 CREATE UNIQUE INDEX Supermarkt_index ON Supermarkt(SupermarktName);
```

#### Messages

20:20:16      Started executing query at Line 1  
Commands completed successfully.  
Total execution time: 00:00:00.021

- 5) Die folgende Abfrage zeigt dass, eine Liste wird erstellen und die Liste zeigen Personalvorname und Personalnachname in einer Liste.

```
CREATE UNIQUE INDEX PersonalVorname_index ON Personal(PersonalVorname, PersonalNachname);
```

```
1 CREATE INDEX PersonalVorname_index ON Personal(PersonalVorname, PersonalNachname)
```

#### Messages

20:17:32      Started executing query at Line 1  
Commands completed successfully.  
Total execution time: 00:00:00.050

# TRANSAKTIONEN

- 1) Die folgende Abfrage zeigt dass, die Produktpreis wird geändert und diese Änderung mit einer Transaktion erstellen.

```
DECLARE @TranName nvarchar(20)
SELECT @TranName = 'UpdateProduktTran'
BEGIN TRANSACTION @TranName
UPDATE Produkt
SET ProduktPreis=64.99 WHERE GTINNummer = '1234567891233';
COMMIT TRANSACTION @TranName;
```

```
1  DECLARE @TranName nvarchar(20)
2  SELECT @TranName='UpdateProduktTran'
3  BEGIN TRANSACTION @TranName
4  UPDATE Produkt
5  SET ProduktPreis=64.99 WHERE GTINNummer='1234567891233';
6  COMMIT TRANSACTION @TranName;
```

## Messages

02:58:54      Started executing query at Line 1  
(1 row affected)  
Total execution time: 00:00:00.018

- 2) Die folgende Abfrage zeigt, dass, vor dem Löschen eines Fahrzeug einen Sicherungspunkt genommen und das Fahrzeug gelöscht hat, dann einen zweiten Sicherungspunkt genommen und ein neues Fahrzeug hinzugefügt hat und in seinem Endzustand an die Position zurückkehrt, an der das Fahrzeug nicht gelöscht wurde und die neues Fahrzeug wurde nicht hinzugefügt.

```
DECLARE @TranName nvarchar(20)
SELECT @TranName='DeleteAndInsert'
BEGIN TRANSACTION @TranName
SAVE TRANSACTION SP1
DELETE Fahrzeug WHERE Fahrzeug.Nummernschild='26YA2813';
SAVE TRANSACTION SP2
INSERT INTO Fahrzeug VALUES ('26TA2813','Renault','Megane','2022');
ROLLBACK TRANSACTION SP1
COMMIT
```

```
1  DECLARE @TranName nvarchar(20)
2  SELECT @TranName='DeleteAndInsert'
3  BEGIN TRANSACTION @TranName
4  SAVE TRANSACTION SP1
5  DELETE Fahrzeug WHERE Fahrzeug.Nummernschild='26YA2813';
6  SAVE TRANSACTION SP2
7  INSERT INTO Fahrzeug VALUES ('26TA2813','Renault','Megane','2022');
8  ROLLBACK TRANSACTION SP1
9  COMMIT
```

## Messages

03:00:29      Started executing query at Line 1  
(1 row affected)  
(1 row affected)  
Total execution time: 00:00:00.009

## TRIGGER

- 1) Die folgende Abfrage zeigt dass, die Rabatt wird auslösen wenn diesem auslösung ermöglichen.

```
CREATE TRIGGER Rabatt ON Produkt
AFTER INSERT AS UPDATE Produkt SET ProduktPreis=ProduktPreis- ProduktPreis * 10/100
WHERE Kategorie='Milchprodukte'
```

```
1 CREATE TRIGGER Rabatt ON Produkt
2 AFTER INSERT AS UPDATE Produkt SET ProduktPreis=ProduktPreis- ProduktPreis * 10/100
3 WHERE Kategorie='Milchprodukte'
```

### Messages

20:38:20      Started executing query at Line 1  
Commands completed successfully.  
Total execution time: 00:00:00.031

## STORED PROCEDURE

- 1) Die folgende Abfrage zeigt dass, die Alle Supermarkte in dieser Stadt wird zeigen wenn die Variable von Stadt Wählen.

```
CREATE PROCEDURE SelectAlleSupermarkt @Stadt nvarchar(25) AS SELECT * FROM
Supermarkt WHERE Stadt = @Stadt;
GO
EXEC SelectAlleSupermarkt @Stadt = N'Istanbul';
```

```
1 CREATE PROCEDURE SelectAlleSupermarkt @Stadt nvarchar(25) AS SELECT * FROM Supermarkt WHERE Stadt = @Stadt;
2 GO
3 EXEC SelectAlleSupermarkt @Stadt = N'Istanbul';
```

### Results   Messages

	SupermarktID	SupermarktName	Adresse	Stadt	Distrikt	KoordinatenX	KoordinatenY	Type
1	34001	MOONGROSS Acıbadem	Acıbadem mh. Acıbadem cd...	İstanbul	Kadıköy	29,0374584197998	41,001033782959	MMM
2	34002	MOONGROSS Barbaros	Barbaros mh. Azizboy sk. ...	İstanbul	Ataşehir	29,0980968475342	40,9971694946289	MM
3	34003	MOONGROSS KemerCountry	Göktürk mh. Kemerburgaz y...	İstanbul	Eyüp	28,8838119506836	41,1797065734863	MMMM
4	34004	MOONGROSS Kalamış	Fenerbahçe mh. Cemil Topu...	İstanbul	Kadıköy	29,0473175048828	40,973934173584	MMMM