



**Department of Computer Engineering**

**CSE 414 – Databases**

**Spring 2021 – 2022**

**Final Project Report**

17.06.2022

**Esra Eryılmaz**

**171044046**

# ~ Pharmacy Management System ~

## 1) Problem Definition

The main purpose of this project is to manage the database of a pharmacy efficiently. I acquire this aim by creating a database of the available medicines, suppliers of those medicines, employees, the customers who visit the pharmacy, the doctors of those customers (patients) etc.

The pharmacist will use the pharmacy management system to minimize the time and resources by maintaining the details of the medicines and the customers(patients) systemically so that the data can be used in the possible quickest time. At the same time, the resources which are minimized are workforce, money, papers, etc. The system is user-friendly and will help the pharmacist. This Pharmacy Management System will reduce the pharmacist's burden and make the system efficient.

## 2) Implementation Details

I write all my implementations on the Microsoft SQL Server. For the user interface part I used Python.



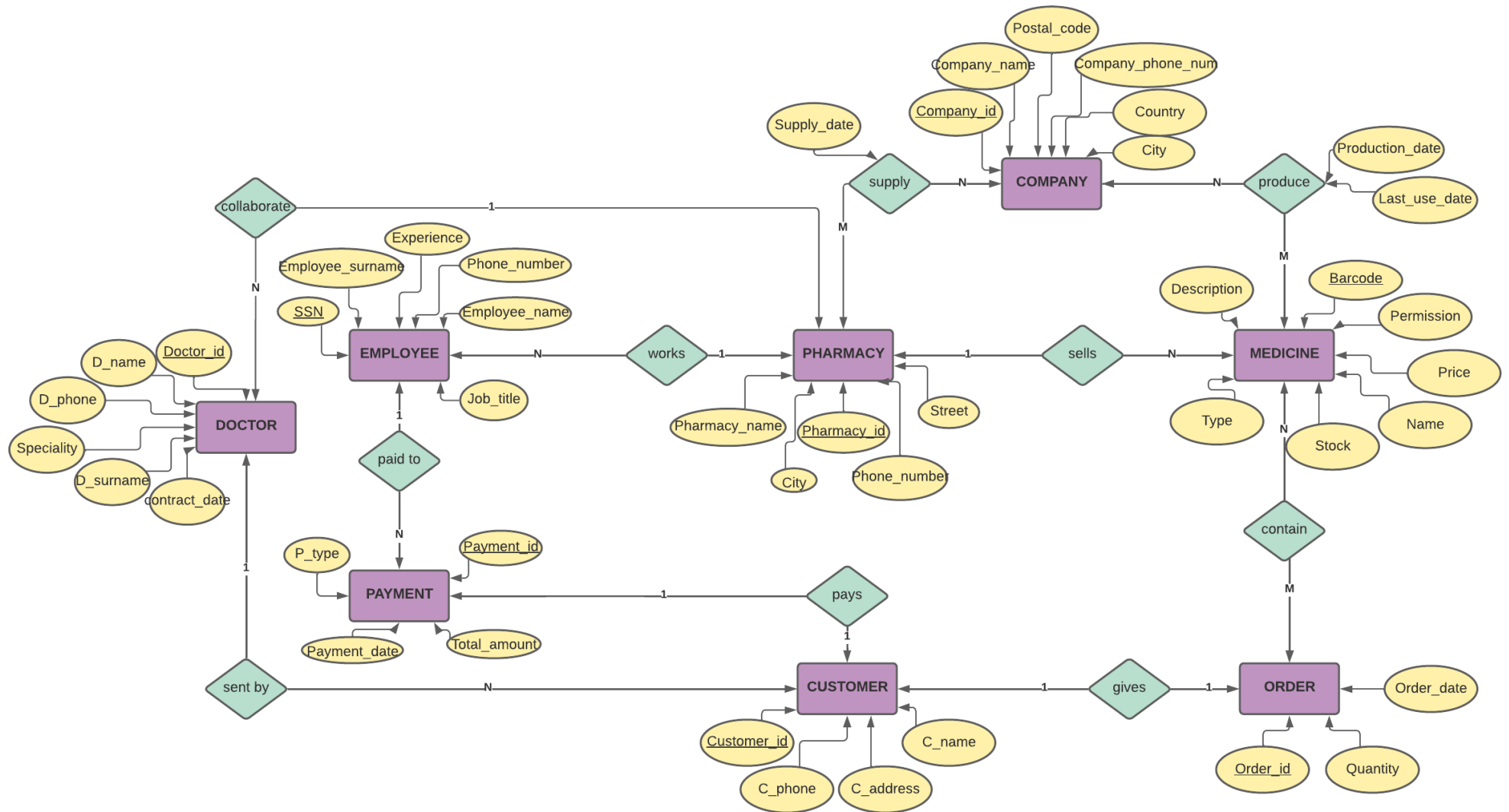
### 3) User Requirements

Pharmacy management system has lots of relations so it has lots of requirements.

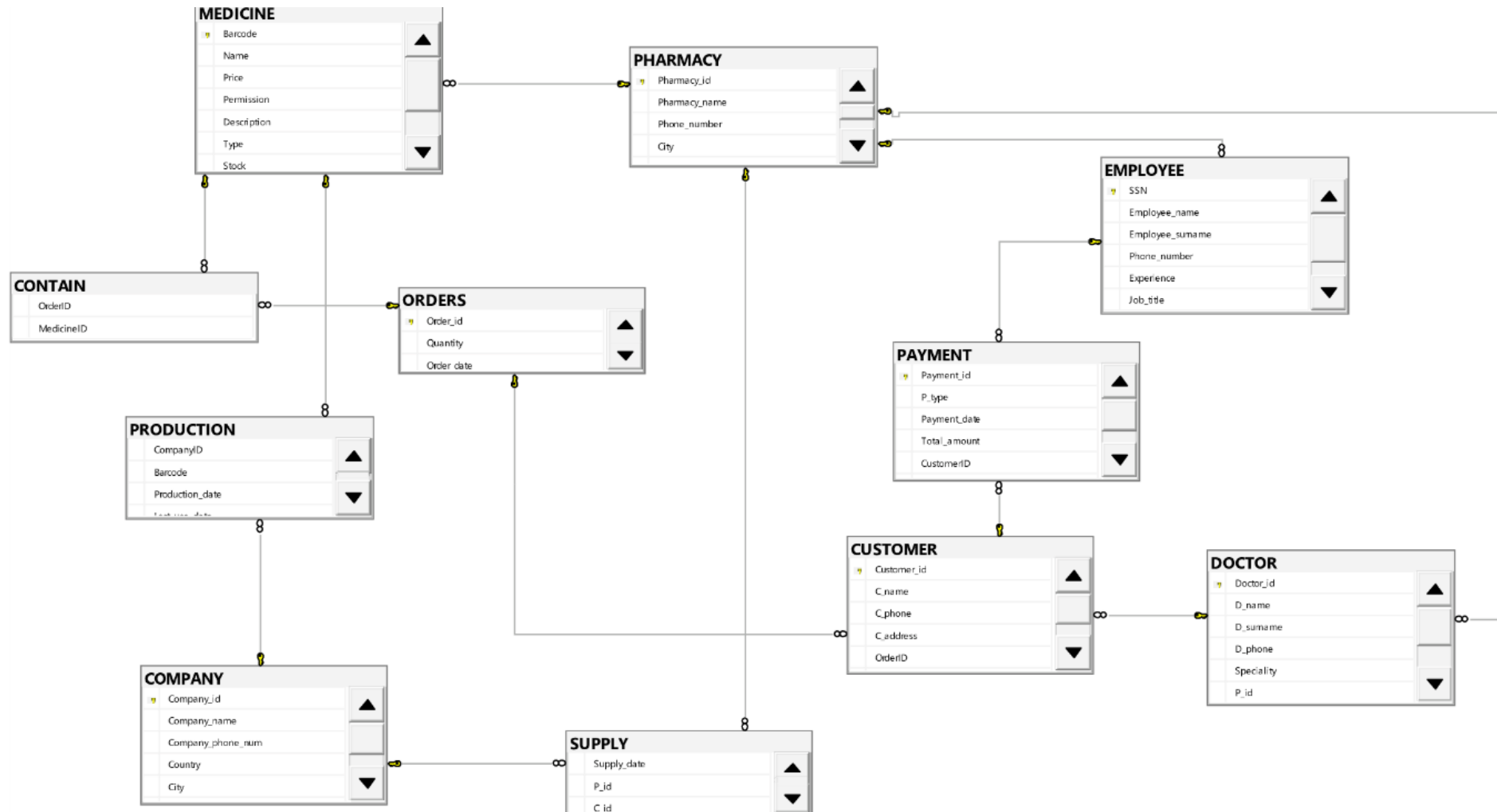
- Pharmacy can sell medicine,
- Pharmacy can supply medicine from the company
- Employees can work in pharmacy,
- Employees can receive payment from the customer,
- Doctors can collaborate with pharmacy,
- Doctors can send customers to the pharmacy,
- Company can produce medicine,
- Order contains medicine,
- Customers can pay payments to the employee,
- Customers can order medicine from the pharmacy.



## 4) E-R Diagram



## 5) Tables



## 5.1) Tables with their values

### • PHARMACY

	Pharmacy_id	Pharmacy_name	Phone_number	City	Street
1	1	Esra Eczanesi	0216900	İstanbul	Çamlıca Caddesi

### • MEDICINE

	Barcode	Name	Price	Permission	Description	Type	Stock	P_id
1	100	Advil	6	1	ağrı kesici	tablet	35	1
2	101	Sebamed	40	1	cilt bakım	krem	15	1
3	102	Aerius	10	1	alerji hâpı	tablet	25	1
4	103	Nexium	20	1	mide ilacı	tablet	20	1
5	104	Norvasc	15	1	tansiyon ilacı	tablet	10	1
6	105	Perebron	9	1	öksürük şurubu	şurup	10	1
7	106	Metpamid	12	1	bulantı şurubu	şurup	10	1
8	107	Naprosyn	47	1	romatizma ilacı	tablet	20	1
9	108	Aspirin	1	1	ağrı kesici	tablet	35	1
10	109	Corasbin	17	1	kan sulandırıcı	tablet	20	1
11	110	Voltaren Emulgel	48	1	romatizma ilacı	jel	10	1
12	111	Ocuvite Complete	22	1	göz vitamini	tablet	10	1
13	112	Otrivin	16	1	burun spreyi	sprey	20	1
14	113	Benexol b12	10	1	vitamin ilacı	tablet	35	1
15	114	Asiviral Krem	10	1	uçuk kremi	krem	20	1
16	115	Terramycin	13	1	yara merhemi	krem	10	1
17	116	Siprogut	9	1	kulak damlası	da...	10	1
18	117	Glifor	19	1	şeker ilacı	tablet	10	1
19	118	Ritalin	30	0	uyuşturucu	tablet	10	1
20	119	Medikinet Retart	40	0	uyuşturucu	tablet	20	1
21	120	Prozac	30	0	sakinleştirici	tablet	10	1

### • EMPLOYEE

	SSN	Employee_name	Employee_surname	Phone_number	Experience	Job_title	P_ID
1	11111	Esra	Eryılmaz	535-45-45	2	Eczacı	1
2	22222	Hasan	Eryılmaz	555-12-12	24	Müdür	1
3	33333	Dilek	Eryılmaz	545-67-67	22	Muhasebeci	1
4	44444	Utku	Eryılmaz	535-55-55	1	Kalfa	1

### • DOCTOR

	Doctor_id	D_name	D_surname	D_phone	Speciality	P_id	contract_data
1	10	Mehmet	Öz	777777	Genel Cerrahi	1	2016-06-10
2	11	Ayşe	Demir	666666	Göz Hastalıkları	1	2014-01-05
3	12	Ali	Arslan	555555	Dahiliye	1	2018-02-15
4	13	Meryem	Çelik	444444	Kadın Hastalıkları	1	2015-10-18
5	14	Ahmet	Yılmaz	333333	Çocuk Hastalıkları	1	2017-05-01
6	15	Manolya	Kaya	222222	Psikiyatri	1	2020-09-09
7	16	Burak	Saz	111111	KBB	1	2021-12-22
8	17	Hatice	Koç	888888	Endokrinoloji	1	2015-03-13
9	18	Deniz	Yurt	999999	Dermatoloji	1	2021-07-20
10	19	Yaprak	Şahin	111000	Ortopedi	1	2022-06-10

- COMPANY

	Company_id	Company_name	Company_phone_num	Country	City	Postal_code
1	500	Novartis	+8-67-431	İsviçre	Basel	11
2	501	Pfizer	+7-23-135	ABD	New York	22
3	502	Abdi İbrahim	+9-43-12-432	Türkiye	İstanbul	34
4	503	gsk	+4-23-21-777	İngiltere	Londra	44
5	504	Bayer	+5-90-654	Almanya	Leverkusen	55
6	505	Nobel	+9-32-456	Türkiye	İstanbul	34
7	506	Sanovel	+9-55-990	Türkiye	İstanbul	34
8	507	Sandoz	+8-34-542	İsviçre	Basel	11
9	508	Berko	+9-43-21-111	Türkiye	İstanbul	34
10	509	Sanofi	+3-55-666	Fransa	Paris	66

- CUSTOMER

	Customer_id	C_name	C_phone	C_address	OrderID	DoctorID
1	1000	Yaprak Şahin	555-666	Kadıköy	30	10
2	1001	Safiye Derenoğlu	222-333	Beyoğlu	31	15
3	1002	Sevgi Akın	111-222	Tuzla	32	11
4	1003	Yusuf Yıldız	999-999	Üsküdar	33	15
5	1004	Fatma Kara	333-555	Şişli	34	19
6	1005	Mustafa Çetin	444-111	Beykoz	35	17
7	1006	Emine Kılıç	999-000	Ataşehir	36	13
8	1007	Zeynep Yıldırım	777-555	Bakırköy	37	12
9	1008	Gülben Derenoğlu	888-222	Ümraniye	38	15
10	1009	Onur Yılmaz	111-999	Üsküdar	39	16

- PAYMENT

	Payment_id	P_type	Payment_date	Total_amount	CustomerID	EmployeeID
1	800	kart	2021-03-14	10	1000	33333
2	801	kart	2021-03-14	10	1001	33333
3	802	nakit	2021-03-14	20	1002	33333
4	803	nakit	2021-03-14	15	1003	11111
5	804	nakit	2021-03-14	59	1004	22222
6	805	nakit	2021-03-14	35	1005	44444
7	806	kart	2021-03-14	19	1006	33333
8	807	kart	2021-03-14	29	1007	33333
9	808	kart	2021-03-14	35	1008	33333
10	809	nakit	2021-03-14	5	1009	33333

- ORDER

	Order_id	Quantity	Order_date
1	30	1	2022-06-03
2	31	2	2022-06-03
3	32	2	2022-06-04
4	33	3	2022-06-05
5	34	5	2022-06-07
6	35	2	2022-06-08
7	36	1	2022-06-10
8	37	1	2022-06-10
9	38	3	2022-06-10
10	39	4	2022-06-12

- SUPPLY

	Supply_date	P_id	C_id
1	2022-06-05	1	500
2	2022-06-10	1	501
3	2022-06-15	1	502
4	2022-05-10	1	503
5	2022-04-15	1	504
6	2022-03-25	1	505
7	2022-01-30	1	506
8	2022-06-14	1	507
9	2022-06-03	1	508
10	2022-06-04	1	509

- CONTAIN

	OrderID	MedicineID
8	33	120
9	34	100
10	34	112
11	34	116
12	34	103
13	34	105
14	35	102
15	35	112
16	36	115
17	37	104
18	38	107
19	38	120
20	38	116
21	39	100
22	39	102
23	39	105
24	39	111

- PRODUCTION

	CompanyID	Barcode	Production_date	Last_use_date
1	500	100	2022-01-28	2030-01-28
2	500	101	2021-02-02	2025-02-02
3	500	102	2022-05-15	2030-05-15
4	501	103	2022-01-01	2028-01-01
5	501	104	2020-12-30	2023-12-30
6	502	105	2021-10-11	2026-10-11
7	502	106	2022-02-17	2027-02-17
8	503	107	2021-08-19	2028-08-19
9	503	108	2020-10-09	2029-10-09
10	503	109	2021-08-18	2030-08-18
11	504	110	2021-10-13	2040-10-13
12	505	111	2022-02-04	2023-02-04
13	505	112	2022-01-01	2025-01-01
14	506	113	2022-04-20	2024-04-20
15	506	114	2020-12-12	2030-12-12
16	506	115	2021-05-30	2031-05-30
17	507	116	2021-10-15	2032-10-15
18	508	117	2022-04-10	2032-04-10
19	508	118	2022-06-06	2028-06-06
20	509	119	2022-03-17	2025-03-17
21	509	120	2022-01-13	2025-01-13



## 5) Functional Dependencies

- PHARMACY

Pharmacy\_id  $\rightarrow$  Pharmacy\_name, Phone\_number, City, Street

- COMPANY

Company\_id  $\rightarrow$  Company\_name, Company\_phone\_num, Country, City, Postal\_code

- SUPPLY

Supply\_date  $\rightarrow$  P\_id, C\_id

- ORDERS

Order\_id  $\rightarrow$  Quantity, Order\_date

- MEDICINE

Barcode  $\rightarrow$  Name, Price, Permission, Description, Type, Stock, P\_id

- EMPLOYEE

SSN  $\rightarrow$  Employee\_name, Employee\_surname, Phone\_number, Experience, Job\_title, P\_ID

- CONTAIN

OrderID  $\rightarrow$  MedicineID

- PRODUCTION

CompanyID  $\rightarrow$  Barcode, Production\_date, Last\_use\_date

- DOCTOR

Doctor\_id → D\_name, D\_surname, D\_phone, Speciality, P\_id,  
contract\_date

- CUSTOMER

Customer\_id → C\_name, C\_phone, C\_address, OrderID, DoctorID

- PAYMENT

Payment\_id → P\_type, Payment\_date, Total\_amount, CustomerID,  
EmployeeID

## 6) Normalizations

Normalization is the process of organizing data in a database. This includes creating tables and establishing relationships between those tables according to rules designed both to protect the data and to make the database more flexible by eliminating redundancy and inconsistent dependency.

While designing the pharmacy system, I paid attention to its efficiency. For this reason, my database is already normalized. But I am gonna explain here through some examples considering its unnormalized state.

### 7.1) First Normal Form (1NF)

Rules :

- Each column should contain only one atomic data.
- Each column should have the same type of data.
- Each column should have a unique name.
- Order of data doesn't matter.

For example while creating the tables I separated city and country attributes from each other.

Every column has same types of values.

## **7.2) Second Normal Form (2NF)**

Rules :

- Tables should 1NF.
- They should not have partial dependencies.

For example MedicineID just depends on OrderID, so I make a new table for these two attributes which is called CONTAIN table. For removing all partial dependencies.

## **7.3) Third Normal Form (3NF)**

Rules :

- Tables should be 2NF.
- They should not have transitive dependency.

Payment depends on customer and customer depends on doctor. With this payment depends on doctor too. So I divide all these into 3 different tables. In this way there are no transitive dependency.

## **7.4) Boyce -Codd Normal Form (BCNF)**

Rules :

- Tables should be 3NF.
- For any dependency  $A \rightarrow B$ , A should be a super key.

## 7) User Interface

In this part I make a terminal based interface, it is not a good looking interface. I just show that I connect the database with the high level language. This connectio into a Microsoft SQL Server is done by pyodbc library. pyodbc is an open source Python module that makes accessing ODBC databases simple.

The database connectivity done. So I can show every query on the Python like following examples,

Here is the connection code part :

```
# Connecting to a Microsoft SQL Server with pyodbc
import pyodbc

# define the server and the database
server = 'ESRA'
database = 'PharmacySystem'

# Define the connection string
conx = pyodbc.connect(
    'DRIVER={SQL Server}; \
    SERVER='+ server +'; \
    DATABASE='+ database +';\
    Trusted_Connection=yes;'
)

conx_string = "driver={SQL SERVER}; server=ESRA; database=PharmacySystem; trusted_connection=YES;"

query1 = "SELECT MEDICINE.Barcode, MEDICINE.Name, MEDICINE.Description FROM MEDICINE"

conx = pyodbc.connect(conx_string)
cursor = conx.cursor()
cursor.execute(query1)
data = cursor.fetchall()

for d in data:
    print(d)
```

Output :

```
(100, 'Advil', 'ağrı kesici')
(101, 'Sebamed', 'cilt bakım')
(102, 'Aerius', 'alerji hâpı')
(103, 'Nexium', 'mide ilacı')
(104, 'Norvasc', 'tansiyon ilacı')
(105, 'Perebron', 'öksürük şurubu')
(106, 'Metpamid', 'bulantı şurubu')
(107, 'Naprosyn', 'romatizma ilacı')
(108, 'Aspirin', 'ağrı kesici')
(109, 'Corasbin', 'kan sulandırıcı')
(110, 'Voltaren Emulgel', 'romatizma ilacı')
(111, 'Ocuville Complete', 'göz vitamini')
(112, 'Otrivin', 'burun spreyi')
(113, 'Benexol b12', 'vitamin ilacı')
(114, 'Asiviral Krem', 'uçuk kremi')
(115, 'Terramycin', 'yara merhemi')
(116, 'Siprogut', 'kulak damlası')
(117, 'Glifor', 'şeker ilacı')
(118, 'Ritalin', 'uyuşturucu')
(119, 'Medikinet Retart', 'uyuşturucu')
(120, 'Prozac', 'sakinleştirici')
```

Some other query examples :

```
query2 = "SELECT * FROM PHARMACY"
query3 = "SELECT * FROM MEDICINE"
query4 = "SELECT * FROM EMPLOYEE"
query5 = "SELECT * FROM COMPANY"
query6 = "SELECT * FROM DOCTOR"
query7 = "SELECT * FROM CUSTOMER"
```

### ### PHARMACY

```
❏ conx = pyodbc.connect(conx_string)
   cursor = conx.cursor()
   cursor.execute(query2)
   data = cursor.fetchall()

   for d in data:
       print(d)
```

```
(1, 'Esra Eczanesi', '0216900', 'İstanbul', 'Çamlıca Caddesi')
```

### ### MEDICINE

```
❏ conx = pyodbc.connect(conx_string)
   cursor = conx.cursor()
   cursor.execute(query3)
   data = cursor.fetchall()

   for d in data:
       print(d)
```

```
(100, 'Advil', 6.0, True, 'ağrı kesici', 'tablet', 35, 1)
(101, 'Sebamed', 40.0, True, 'cilt bakım', 'krem', 15, 1)
(102, 'Aerius', 10.0, True, 'alerji hapi', 'tablet', 25, 1)
(103, 'Nexium', 20.0, True, 'mide ilacı', 'tablet', 20, 1)
(104, 'Norvasc', 15.0, True, 'tansiyon ilacı', 'tablet', 10, 1)
(105, 'Perebron', 9.0, True, 'öksürük şurubu', 'şurup', 10, 1)
(106, 'Metpamid', 12.0, True, 'bulantı şurubu', 'şurup', 10, 1)
(107, 'Naprosyn', 47.0, True, 'romatizma ilacı', 'tablet', 20, 1)
(108, 'Aspirin', 1.0, True, 'ağrı kesici', 'tablet', 35, 1)
(109, 'Corasbin', 17.0, True, 'kan sulandırıcı', 'tablet', 20, 1)
(110, 'Voltaren Emulgel', 48.0, True, 'romatizma ilacı', 'jel', 10, 1)
(111, 'Ocuville Complete', 22.0, True, 'göz vitamini', 'tablet', 10, 1)
(112, 'Otrivin', 16.0, True, 'burun spreyi', 'sprey', 20, 1)
(113, 'Benexol b12', 10.0, True, 'vitamin ilacı', 'tablet', 35, 1)
(114, 'Asiviral Krem', 10.0, True, 'uçuk kremi', 'krem', 20, 1)
(115, 'Terramycin', 13.0, True, 'yara merhemi', 'krem', 10, 1)
(116, 'Siprogut', 9.0, True, 'kulak damlası', 'damla', 10, 1)
(117, 'Glifor', 19.0, True, 'şeker ilacı', 'tablet', 10, 1)
(118, 'Ritalin', 30.0, False, 'uyuşturucu', 'tablet', 10, 1)
(119, 'Medikinet Retart', 40.0, False, 'uyuşturucu', 'tablet', 20, 1)
(120, 'Prozac', 30.0, False, 'sakinleştirici', 'tablet', 10, 1)
```

### ### EMPLOYEE

```

:  ▶ conx = pyodbc.connect(conx_string)
    cursor = conx.cursor()
    cursor.execute(query4)
    data = cursor.fetchall()

    for d in data:
        print(d)

(11111, 'Esra', 'Eryılmaz', '535-45-45', 2, 'Eczacı', 1)
(22222, 'Hasan', 'Eryılmaz', '555-12-12', 24, 'Müdür', 1)
(33333, 'Dilek', 'Eryılmaz', '545-67-67', 22, 'Muhasebeci', 1)
(44444, 'Utku', 'Eryılmaz', '535-55-55', 1, 'Kalfa', 1)

```

~~~~~

### ### COMPANY

```

▶ # conx = pyodbc.connect(conx_string)
  cursor = conx.cursor()
  cursor.execute(query5)
  data = cursor.fetchall()

  for d in data:
      print(d)

(500, 'Novartis', '+8-67-431', 'İsviçre', 'Basel', 11)
(501, 'Pfizer', '+7-23-135', 'ABD', 'New York', 22)
(502, 'Abdi İbrahim', '+9-43-12-432', 'Türkiye', 'İstanbul', 34)
(503, 'gsk', '+4-23-21-777', 'İngiltere', 'Londra', 44)
(504, 'Bayer', '+5-90-654', 'Almanya', 'Leverkusen', 55)
(505, 'Nobel', '+9-32-456', 'Türkiye', 'İstanbul', 34)
(506, 'Sanovel', '+9-55-990', 'Türkiye', 'İstanbul', 34)
(507, 'Sandoz', '+8-34-542', 'İsviçre', 'Basel', 11)
(508, 'Berko', '+9-43-21-111', 'Türkiye', 'İstanbul', 34)
(509, 'Sanofi', '+3-55-666', 'Fransa', 'Paris', 66)

```

~~~~~

### ### DOCTOR

```

:  ▶ conx = pyodbc.connect(conx_string)
    cursor = conx.cursor()
    cursor.execute(query6)
    data = cursor.fetchall()

    for d in data:
        print(d)

(10, 'Mehmet', 'Öz', '777777', 'Genel Cerrahi', 1, '2016-06-10')
(11, 'Ayşe', 'Demir', '666666', 'Göz Hastalıkları', 1, '2014-01-05')
(12, 'Ali', 'Arslan', '555555', 'Dahiliye', 1, '2018-02-15')
(13, 'Meryem', 'Çelik', '444444', 'Kadın Hastalıkları', 1, '2015-10-18')
(14, 'Ahmet', 'Yılmaz', '333333', 'Çocuk Hastalıkları', 1, '2017-05-01')
(15, 'Manolya', 'Kaya', '222222', 'Psikiyatri', 1, '2020-09-09')
(16, 'Burak', 'Saz', '111111', 'KBB', 1, '2021-12-22')
(17, 'Hatice', 'Koç', '888888', 'Endokrinoloji', 1, '2015-03-13')
(18, 'Deniz', 'Yurt', '999999', 'Dermatoloji', 1, '2021-07-20')
(19, 'Yaprak', 'Şahin', '111000', 'Ortopedi', 1, '2022-06-10')

```

~~~~~

### ### CUSTOMER

```

❏ conx = pyodbc.connect(conx_string)
   cursor = conx.cursor()
   cursor.execute(query7)
   data = cursor.fetchall()

   for d in data:
       print(d)

(1000, 'Yaprak Şahin', '555-666', 'Kadıköy', 30, 10)
(1001, 'Safiye Derenoğlu', '222-333', 'Beyoğlu', 31, 15)
(1002, 'Sevgi Akın', '111-222', 'Tuzla', 32, 11)
(1003, 'Yusuf Yıldız', '999-999', 'Üsküdar', 33, 15)
(1004, 'Fatma Kara', '333-555', 'Şişli', 34, 19)
(1005, 'Mustafa Çetin', '444-111', 'Beykoz', 35, 17)
(1006, 'Emine Kılıç', '999-000', 'Ataşehir', 36, 13)
(1007, 'Zeynep Yıldırım', '777-555', 'Bakırköy', 37, 12)
(1008, 'Gülben Derenoğlu', '888-222', 'Ümraniye', 38, 15)
(1009, 'Onur Yılmaz', '111-999', 'Üsküdar', 39, 16)

```

### ### Show medicines which will expire before '2025-05-05'

```

: ❏ conx = pyodbc.connect(conx_string)
   cursor = conx.cursor()
   cursor.execute(query8)
   data = cursor.fetchall()

   for d in data:
       print(d)

('Sebamed', )
('Norvasc', )
('Ocuville Complete', )
('Otrivin', )
('Benexol b12', )
('Medikinet Retart', )
('Prozac', )

```

### #### VIEW EXAMPLE

```

❏ conx = pyodbc.connect(conx_string)
   cursor = conx.cursor()
   cursor.execute(query10)
   data = cursor.fetchall()

   for d in data:
       print(d)

('Mehmet', 'Öz', '777777', 'Genel Cerrahi', 'Yaprak Şahin', 'Kadıköy', '555-666')
('Manolya', 'Kaya', '222222', 'Psikiyatri', 'Safiye Derenoğlu', 'Beyoğlu', '222-333')
('Ayşe', 'Demir', '666666', 'Göz Hastalıkları', 'Sevgi Akın', 'Tuzla', '111-222')
('Manolya', 'Kaya', '222222', 'Psikiyatri', 'Yusuf Yıldız', 'Üsküdar', '999-999')
('Yaprak', 'Şahin', '111000', 'Ortopedi', 'Fatma Kara', 'Şişli', '333-555')
('Hatice', 'Koç', '888888', 'Endokrinoloji', 'Mustafa Çetin', 'Beykoz', '444-111')
('Meryem', 'Çelik', '444444', 'Kadın Hastalıkları', 'Emine Kılıç', 'Ataşehir', '999-000')
('Ali', 'Arslan', '555555', 'Dahiliye', 'Zeynep Yıldırım', 'Bakırköy', '777-555')
('Manolya', 'Kaya', '222222', 'Psikiyatri', 'Gülben Derenoğlu', 'Ümraniye', '888-222')
('Burak', 'Saz', '111111', 'KBB', 'Onur Yılmaz', 'Üsküdar', '111-999')

```

## 8) SQL Joins

- Outer Right Join

Show medicines which will expire before 2025-05-05

```
SELECT MEDICINE.Name
FROM MEDICINE
RIGHT JOIN PRODUCTION ON MEDICINE.Barcode = PRODUCTION.Barcode
Where PRODUCTION.Last_use_date < '2025-05-05'
```

|   | Name             |
|---|------------------|
| 1 | Sebamed          |
| 2 | Norvasc          |
| 3 | Ocuvite Complete |
| 4 | Otrivin          |
| 5 | Benexol b12      |
| 6 | Medikinet Retart |
| 7 | Prozac           |

- Outer Left Join

Companies which supplied in 'Türkiye'

```
SELECT COMPANY.Company_id, Company_name, Country, City
FROM COMPANY
LEFT JOIN SUPPLY ON COMPANY.Company_id = SUPPLY.C_id
WHERE COMPANY.Country in ('Türkiye');
```

|   | Company_id | Company_name | Country | City     |
|---|------------|--------------|---------|----------|
| 1 | 502        | Abdi İbrahim | Türkiye | İstanbul |
| 2 | 505        | Nobel        | Türkiye | İstanbul |
| 3 | 506        | Sanovel      | Türkiye | İstanbul |
| 4 | 508        | Berko        | Türkiye | İstanbul |

- Full Outer Join

Show 'Psikiyatri' customers (patients)

```
SELECT Customer_id, C_name, C_phone, C_address
FROM CUSTOMER
FULL OUTER JOIN DOCTOR ON DOCTOR.Doctor_id = CUSTOMER.DoctorID
WHERE DOCTOR.Speciality in ('Psikiyatri');
```

|   | Customer_id | C_name           | C_phone | C_address |
|---|-------------|------------------|---------|-----------|
| 1 | 1001        | Safiye Derenoğlu | 222-333 | Beyoğlu   |
| 2 | 1003        | Yusuf Yıldız     | 999-999 | Üsküdar   |
| 3 | 1008        | Gülben Derenoğlu | 888-222 | Ümraniye  |



## 10) Views

- View 1

View of customers' informations with their doctors

```
CREATE VIEW view1 AS
SELECT DOCTOR.D_name, DOCTOR.D_surname, DOCTOR.D_phone,
DOCTOR.Speciality, CUSTOMER.C_name, CUSTOMER.C_address, CUSTOMER.C_phone
FROM DOCTOR, CUSTOMER
WHERE DOCTOR.Doctor_id = CUSTOMER.DoctorID;
```

```
SELECT * FROM view1;
```

|    | D_name  | D_surname | D_phone | Speciality         | C_name           | C_address | C_phone |
|----|---------|-----------|---------|--------------------|------------------|-----------|---------|
| 1  | Mehmet  | Öz        | 777777  | Genel Cerrahi      | Yaprak Şahin     | Kadıköy   | 555-666 |
| 2  | Manolya | Kaya      | 222222  | Psikiyatri         | Safiye Derenoğlu | Beyoğlu   | 222-333 |
| 3  | Ayşe    | Demir     | 666666  | Göz Hastalıkları   | Sevgi Akın       | Tuzla     | 111-222 |
| 4  | Manolya | Kaya      | 222222  | Psikiyatri         | Yusuf Yıldız     | Üsküdar   | 999-999 |
| 5  | Yaprak  | Şahin     | 111000  | Ortopedi           | Fatma Kara       | Şişli     | 333-555 |
| 6  | Hatice  | Koç       | 888888  | Endokrinoloji      | Mustafa Çetin    | Beykoz    | 444-111 |
| 7  | Meryem  | Çelik     | 444444  | Kadın Hastalıkları | Emine Kılıç      | Ataşehir  | 999-000 |
| 8  | Ali     | Arslan    | 555555  | Dahiliye           | Zeynep Yıldırım  | Bakırköy  | 777-555 |
| 9  | Manolya | Kaya      | 222222  | Psikiyatri         | Gülben Derenoğlu | Ümraniye  | 888-222 |
| 10 | Burak   | Saz       | 111111  | KBB                | Onur Yılmaz      | Üsküdar   | 111-999 |

- View 2

Combine and view MEDICINE - PRODUCTION tables

```
CREATE VIEW view2 AS
SELECT MEDICINE.Barcode, MEDICINE.Name, MEDICINE.Description,
MEDICINE.Permission, MEDICINE.Price, MEDICINE.Stock, MEDICINE.Type,
PRODUCTION.Production_date, PRODUCTION.Last_use_date
FROM MEDICINE, PRODUCTION
WHERE MEDICINE.Barcode = PRODUCTION.Barcode;
```

|    | Barcode | Name         | Description     | Permission | Price | Stock | Type   | Production_date | Last_use_date |
|----|---------|--------------|-----------------|------------|-------|-------|--------|-----------------|---------------|
| 1  | 100     | Advil        | ağrı kesici     | 1          | 6     | 35    | tablet | 2022-01-28      | 2030-01-28    |
| 2  | 101     | Sebamed      | cilt bakım      | 1          | 40    | 15    | krem   | 2021-02-02      | 2025-02-02    |
| 3  | 102     | Aerius       | alerji hapi     | 1          | 10    | 25    | tablet | 2022-05-15      | 2030-05-15    |
| 4  | 103     | Nexium       | mide ilacı      | 1          | 20    | 20    | tablet | 2022-01-01      | 2028-01-01    |
| 5  | 104     | Norvasc      | tansiyon ilacı  | 1          | 15    | 10    | tablet | 2020-12-30      | 2023-12-30    |
| 6  | 105     | Perebron     | öksürük şurubu  | 1          | 9     | 10    | şurup  | 2021-10-11      | 2026-10-11    |
| 7  | 106     | Metpamid     | bulantı şurubu  | 1          | 12    | 10    | şurup  | 2022-02-17      | 2027-02-17    |
| 8  | 107     | Naprosyn     | romatizma ilacı | 1          | 47    | 20    | tablet | 2021-08-19      | 2028-08-19    |
| 9  | 108     | Aspirin      | ağrı kesici     | 1          | 1     | 35    | tablet | 2020-10-09      | 2029-10-09    |
| 10 | 109     | Corasbin     | kan sulandırıcı | 1          | 17    | 20    | tablet | 2021-08-18      | 2030-08-18    |
| 11 | 110     | Voltare...   | romatizma ilacı | 1          | 48    | 10    | jel    | 2021-10-13      | 2040-10-13    |
| 12 | 111     | Ocuvite ...  | göz vitamini    | 1          | 22    | 10    | tablet | 2022-02-04      | 2023-02-04    |
| 13 | 112     | Otrivin      | burun spreyi    | 1          | 16    | 20    | sprey  | 2022-01-01      | 2025-01-01    |
| 14 | 113     | Benexol...   | vitamin ilacı   | 1          | 10    | 35    | tablet | 2022-04-20      | 2024-04-20    |
| 15 | 114     | Asiviral ... | uçuk kremi      | 1          | 10    | 20    | krem   | 2020-12-12      | 2030-12-12    |
| 16 | 115     | Terramy...   | yara merhemi    | 1          | 13    | 10    | krem   | 2021-05-30      | 2031-05-30    |
| 17 | 116     | Siprogut     | kulak damlası   | 1          | 9     | 10    | da...  | 2021-10-15      | 2032-10-15    |
| 18 | 117     | Glifor       | şeker ilacı     | 1          | 19    | 10    | tablet | 2022-04-10      | 2032-04-10    |
| 19 | 118     | Ritalin      | uyuşturucu      | 0          | 30    | 10    | tablet | 2022-06-06      | 2028-06-06    |
| 20 | 119     | Medikin...   | uyuşturucu      | 0          | 40    | 20    | tablet | 2022-03-17      | 2025-03-17    |
| 21 | 120     | Prozac       | sakinleştirici  | 0          | 30    | 10    | tablet | 2022-01-13      | 2025-01-13    |

- View 3

Combine and view EMPLOYEE - PAYMENT tables

```
CREATE VIEW view3 AS
SELECT
EMPLOYEE.SSN,EMPLOYEE.Employee_name,EMPLOYEE.Employee_surname,EMPLOYEE.Experience,EMPLOYEE.Job_title,EMPLOYEE.Phone_number,PAYMENT.Payment_id,PAYMENT.P_type,PAYMENT.Payment_date,PAYMENT.CustomerID,PAYMENT.Total_amount
FROM EMPLOYEE,PAYMENT
WHERE EMPLOYEE.SSN = PAYMENT.EmployeeID;
```

|    | SSN   | Employee_name | Employee_surname | Experience | Job_title  | Phone_number | Payment_id | P_type | Payment_date | CustomerID | Total_amount |
|----|-------|---------------|------------------|------------|------------|--------------|------------|--------|--------------|------------|--------------|
| 1  | 33333 | Dilek         | Eryılmaz         | 22         | Muhasebeci | 545-67-67    | 800        | kart   | 2021-03-14   | 1000       | 10           |
| 2  | 33333 | Dilek         | Eryılmaz         | 22         | Muhasebeci | 545-67-67    | 801        | kart   | 2021-03-14   | 1001       | 10           |
| 3  | 33333 | Dilek         | Eryılmaz         | 22         | Muhasebeci | 545-67-67    | 802        | nakit  | 2021-03-14   | 1002       | 20           |
| 4  | 11111 | Esra          | Eryılmaz         | 2          | Eczacı     | 535-45-45    | 803        | nakit  | 2021-03-14   | 1003       | 15           |
| 5  | 22222 | Hasan         | Eryılmaz         | 24         | Müdür      | 555-12-12    | 804        | nakit  | 2021-03-14   | 1004       | 59           |
| 6  | 44444 | Utku          | Eryılmaz         | 1          | Kalfa      | 535-55-55    | 805        | nakit  | 2021-03-14   | 1005       | 35           |
| 7  | 33333 | Dilek         | Eryılmaz         | 22         | Muhasebeci | 545-67-67    | 806        | kart   | 2021-03-14   | 1006       | 19           |
| 8  | 33333 | Dilek         | Eryılmaz         | 22         | Muhasebeci | 545-67-67    | 807        | kart   | 2021-03-14   | 1007       | 29           |
| 9  | 33333 | Dilek         | Eryılmaz         | 22         | Muhasebeci | 545-67-67    | 808        | kart   | 2021-03-14   | 1008       | 35           |
| 10 | 33333 | Dilek         | Eryılmaz         | 22         | Muhasebeci | 545-67-67    | 809        | nakit  | 2021-03-14   | 1009       | 5            |

- View 4

Combine and view EMPLOYEE - PHARMACY tables

```
CREATE VIEW view4 AS
SELECT PHARMACY.Pharmacy_id, PHARMACY.Pharmacy_name, PHARMACY.City,
EMPLOYEE.P_ID,EMPLOYEE.Employee_name,EMPLOYEE.Employee_surname,EMPLOYEE.Job_title,EMPLOYEE.SSN,EMPLOYEE.Experience
FROM PHARMACY, EMPLOYEE
WHERE PHARMACY.Pharmacy_id = EMPLOYEE.P_ID;
```

|   | Pharmacy_id | Pharmacy_name | City     | P_ID | Employee_name | Employee_surname | Job_title  | SSN   | Experience |
|---|-------------|---------------|----------|------|---------------|------------------|------------|-------|------------|
| 1 | 1           | Esra Eczanesi | İstanbul | 1    | Esra          | Eryılmaz         | Eczacı     | 11111 | 2          |
| 2 | 1           | Esra Eczanesi | İstanbul | 1    | Hasan         | Eryılmaz         | Müdür      | 22222 | 24         |
| 3 | 1           | Esra Eczanesi | İstanbul | 1    | Dilek         | Eryılmaz         | Muhasebeci | 33333 | 22         |
| 4 | 1           | Esra Eczanesi | İstanbul | 1    | Utku          | Eryılmaz         | Kalfa      | 44444 | 1          |

- View 5

View the medicines that have permission to be sold with receipt.

```
CREATE VIEW view5 AS
SELECT
MEDICINE.Barcode,MEDICINE.Name,MEDICINE.Description,MEDICINE.Permission,
MEDICINE.Type,MEDICINE.Price,MEDICINE.Stock
FROM MEDICINE
WHERE MEDICINE.Permission in (0);
```

|   | Barcode | Name             | Description    | Permission | Type   | Price | Stock |
|---|---------|------------------|----------------|------------|--------|-------|-------|
| 1 | 118     | Ritalin          | uyuşturucu     | 0          | tablet | 30    | 10    |
| 2 | 119     | Medikinet Retart | uyuşturucu     | 0          | tablet | 40    | 20    |
| 3 | 120     | Prozac           | sakinleştirici | 0          | tablet | 30    | 10    |

## 11) Additional Details

- I used inheritance while designing the system. For example EMPLOYEE is a subclass of the PHARMACY. It belongs to the pharmacy.
- When company supplies medicine to the pharmacy, the medicine stocks are updated. It is a atomic transaction.
- Also my E-R diagram have some weak entities. For example CONTAIN, SUPPLY, and PRODUCTION tables are weak entities. They depend on some other tables, without other tables they cannot exist.  
COMPANY supply medicine to the PHARMACY. Without company or pharmacy; supply cannot exist.

### 11.1) Additional Queries to see the more details of the system.

- Sorting medicines by sales quantity

```
SELECT Count(CONTAIN.MedicineID) as Total, MEDICINE.Name
from CONTAIN
LEFT JOIN MEDICINE ON CONTAIN.MedicineID = MEDICINE.Barcode
Group by MEDICINE.Name
order by Total DESC;
```

|    | Total | Name         |
|----|-------|--------------|
| 1  | 4     | Advil        |
| 2  | 3     | Prozac       |
| 3  | 2     | Ritalin      |
| 4  | 2     | Siprogut     |
| 5  | 2     | Otrivin      |
| 6  | 2     | Perebron     |
| 7  | 2     | Aerius       |
| 8  | 1     | Asiviral ... |
| 9  | 1     | Aspirin      |
| 10 | 1     | Napros...    |
| 11 | 1     | Nexium       |
| 12 | 1     | Norvasc      |
| 13 | 1     | Ocuvite...   |
| 14 | 1     | Terram...    |

- Who has experience less than or equal 2 years as an EMPLOYEE?

```
SELECT Employee_name, Employee_surname, Job_title, Experience
FROM EMPLOYEE
WHERE Experience<=2
```

|   | Employee_name | Employee_surname | Job_title | Experience |
|---|---------------|------------------|-----------|------------|
| 1 | Esra          | Eryılmaz         | Eczacı    | 2          |
| 2 | Utku          | Eryılmaz         | Kalfa     | 1          |

- Search Aspirin in MEDICINE table

```
SELECT Name, Barcode, Price, Description, Stock
from MEDICINE
Where Name in ('Aspirin');
```

|   | Name    | Barcode | Price | Description | Stock |
|---|---------|---------|-------|-------------|-------|
| 1 | Aspirin | 108     | 1     | ağrı kesici | 35    |