

TOBB Ekonomi ve Teknoloji
Üniversitesi Bilgisayar
Mühendisliği Bölümü BIL372
Veri tabanı Sistemleri
Proje Ara Raporu

Rana Demir 201401003

Zeynep Meriç Aşık 201410026

Sabahattin Timur Uzun 191101056

Doruk Ata Özkara 201401019

I. Gerçek Dünya Problemi

Veri tabanı sistemleri dersi kapsamında geliştireceğimiz proje ile sağlık alanında bir gerçek dünya problemini çözmemiz beklenmektedir. Çağımız teknolojilerin sunduğu imkanlardan biri de veri toplama ve kaydetme kolaylığıdır. Her gün sayısız yeni veri toplanıyor. Her türlü detayın titizlikle saklanması icap eden sağlık sektörü de teknoloji yardımıyla veri elde etme ve saklama olanaklarını geliştirmiştir. Ancak verilerin miktarının artması düzensizliğe neden olmakta, kayıtlı bilgiye erişimde zorluklar yaşanmaktadır. Yaşanabilecek sorunları çözmek için iyi tanımlanmış veri tabanları doğru yöntemler kullanılarak tasarlanmalıdır.

Veri tabanı sistemleri dersi öğrencileri olarak amacımız, proje geliştirme sürecindeki her adımda doğru yöntemleri uygulayarak bir gerçek dünya problemine alternatif çözüm geliştirmektedir.

Biz gerçek dünya problemi olarak hastaların ve sağlık personellerinin kayıtlarının düzenlenmesiyle ilgilenmeye karar verdik. Problem, Ankara'nın Çankaya ilçesindeki gerek görülürse ilçenin seçeceğimiz bir mahallesindeki sağlık ocaklarının, aile hekimlerinin ve hekimlere bağlı hastaların verilerinin ilişkilerini tanımlanması, uygun halde kaydedilmesi ihtiyacıdır. Problemi daha detaylı açıklamak için yüzeysel bir örnek verecek olursak; bölgedeki sağlık ocakları bilgileri bir tabloda tutulması lazım, her bir sağlık ocağında çalışan sağlık personellerinin özellikleri başka tablolarda tutulmalı, her bir sağlık ocağına kayıtlı hastalar hastalık teşhisleri ve tedavi yöntemleriyle ilişkilendirilmeli.

II. Gereksinim Analizi

Seçilen miniworld: Çankaya ilçesinde bulunmakta olan sağlık ocakları

Mevcut sistem: Veriler bir txt dosyasında depolanır ve manuel olarak yedeklenir.

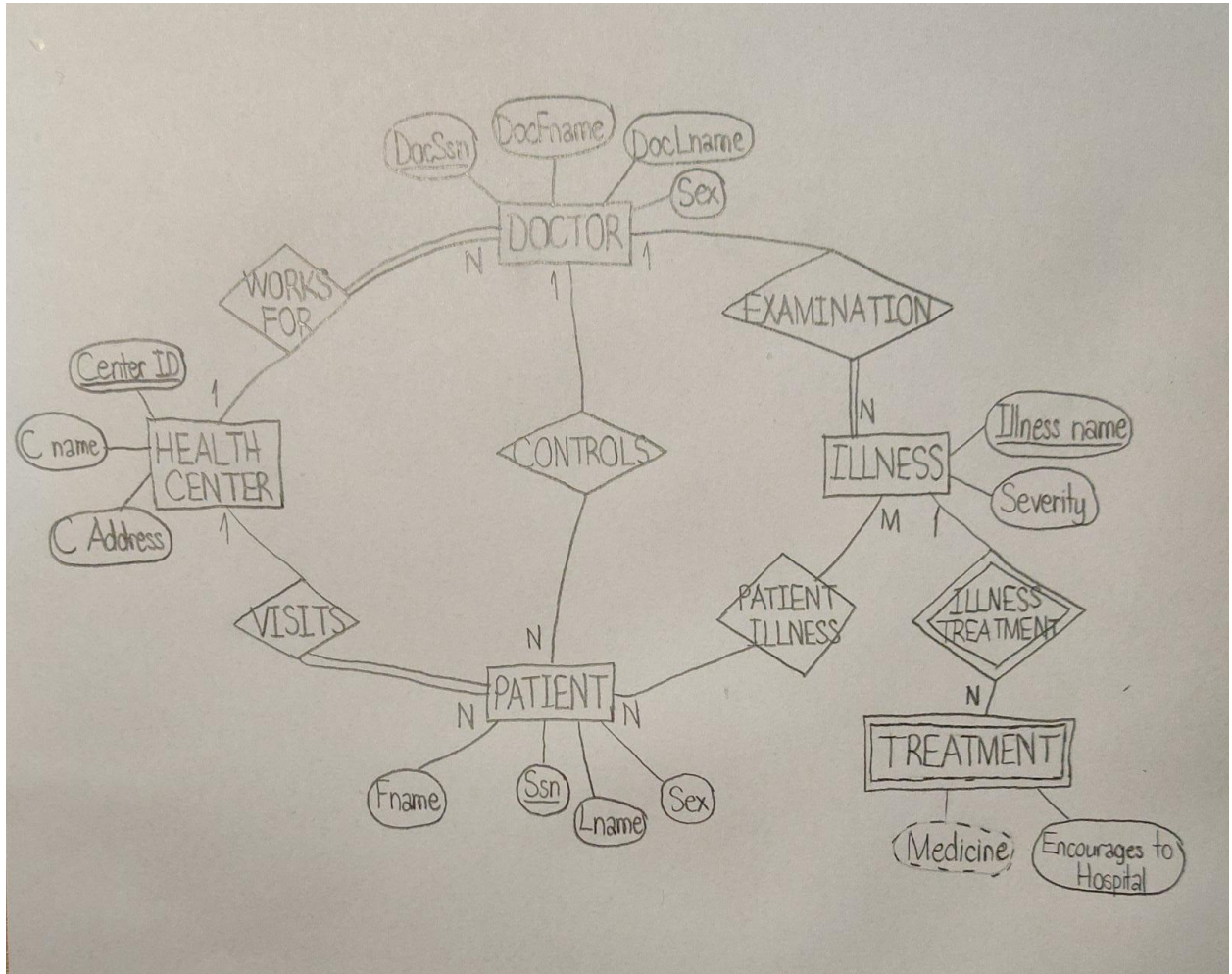
İhtiyaç:

- 8 ya da daha fazla core CPU
- 2 X 16 GB ta da daha fazla RAM
- 5 TB HDD
- Güncel işletim sistemi (tercihen Unix benzeri)
- İnternet bağlantısı
- PostgreSQL server
- Python (3.8 ya da daha yüksek)
- NAS cihazı

Yazılım ve donanım maliyet analizi (yaklaşık):

- CPU (4.400₺)
- RAM (2.600₺)
- HDD (2.000₺)
- OS (Unix benzeri – Ücretsiz / Windows 11 Home Edition – 1.299 ₺ Pro Edition – 1.899 ₺)
- PostgreSQL (Ücretsiz)
- Python (Ücretsiz)
- İnternet (Aylık 450₺)
- NAS Cihazı (7.000₺)

III. Kavramsal tasarım (EER-diyagramları)



Veri tabanında bulunmakta olan varlıklar şu şekilde sıralanabilir: Hekim, hastalık geçmişi, sağlık ocağı, hasta ve tedavi.

Sağlık ocağı, Çankaya’da bulunan her bir sağlık ocağını temsil eden varlık. Diğer tüm varlıkların işlemleri temel olarak burada geçmekte. Anahtar özelliği olarak sağlık ocağı numarası bulunurken başka öznitelikleri ise adı ve adresi.

Hekim, bu sağlık ocaklarında bulunan ve çalışmakta olan her bir doktoru temsil etmekte. Her bir doktor, birbirinden T.C. kimlik numaraları ile ayrılmakta, bunun yanında isim, soy isim, cinsiyet gibi bilgilerine de yer verilmektedir.

Hasta, bu ilçede yer almakta olan sağlık ocaklarına belirli sağlık problemleri sebebiyle başvuran ve burayı ziyaret eden kişileri temsil etmektedir. Bu kişiler, birbirinden tıpkı hekimlerde olduğu gibi T.C. kimlik numaralarıyla ayrılmakta olup isim, soy isim ve cinsiyet bilgilerine de yer verilmektedir.

Hastaların şikayetçi olma sebebi olan rahatsızlıkları temsil eden varlığın adı ise “Rahatsızlık” tır. Rahatsızlığı tanımlayan özellik, onun ismidir. Bunun yanında rahatsızlık derecesi de önemlidir. Buna bağlı olarak tanımlanabilen “Tedavi” varlığı ise bu hastalıklara üretilebilecek çözümleri içermektedir. Tek başına tanımlanamamasından dolayı zayıf varlık olarak tanımlanmaktadır. Bazı hastalıklar için tedavi yöntemi olarak ilaç uygulanabildiği gibi bazıları için hastalar, hastaneye yönlendirilebilmektedir. Bu yüzden hastaneye yönlendirilme durumu tutulmaktadır. Bu durum doğru olduğunda ilaç kısmı null olarak geçmekte, yanlış olduğunda ise ilaç özelliğine null değer veya bir ilaç ismi girilmektedir.

Sağlık ocağında çalışan hekimin durumu bir ilişkiyle temsil edildiğinde bir hekimin yalnız bir yerde çalışabilmekte, bir sağlık ocağının ise birden fazla hekimi içerebilmekte olduğu bir tasarım izlenmektedir. Herhangi bir hekim, mutlaka bir yerde çalışmak zorunda ve içinde hekim bulunmayan kapalı veya yeni açılmış sağlık ocakları bulunabilmektedir.

Sağlık ocağı ve hastalar arasındaki ilişki ise ziyaret etme adı altında kurulmaktadır. Her hastanın bağlı olduğu sadece bir sağlık ocağı vardır. Sağlık ocakları ise birden fazla hastaya açık olabilmektedir. Tıpkı hekim-sağlık ocağı ilişkisinde olduğu gibi yeni açılmış veya kapalı sağlık ocakları var olabileceği için bunlara bağlı olan hiçbir hasta bulunmama olasılığı bulunmaktadır. Buna tezat olarak her hasta mutlaka bir sağlık ocağına bağlı olmak zorundadır.

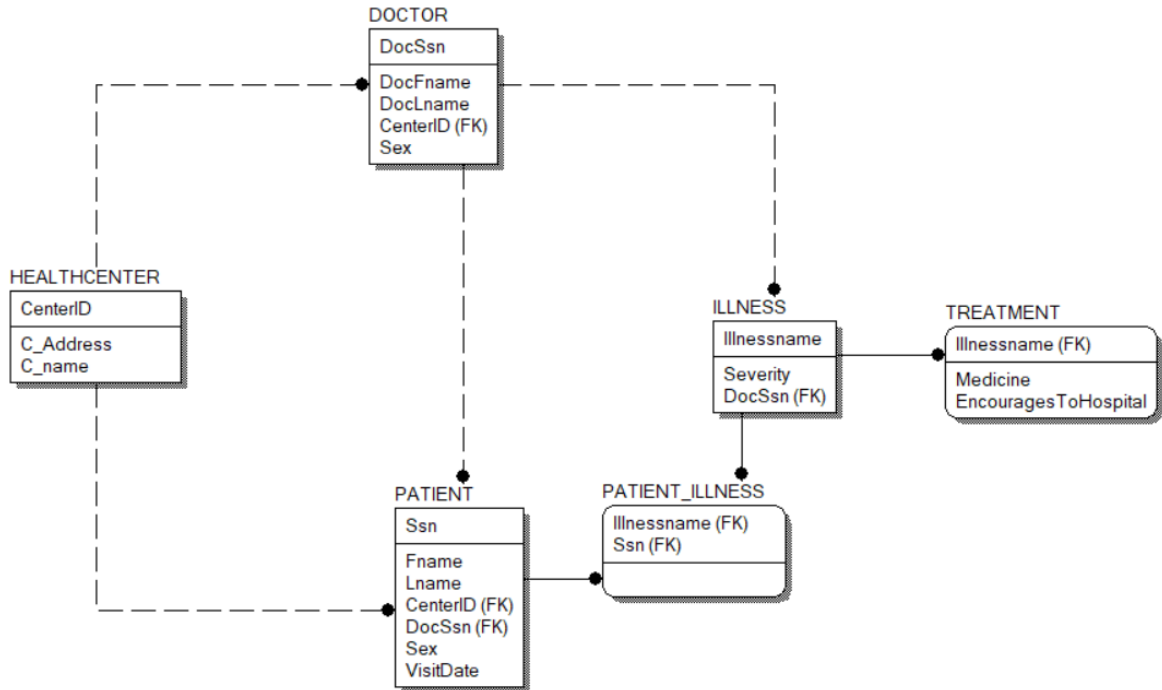
Hekim ile hasta arasındaki ilişkiye “Muayene” adı verilmiş olup bir hekim birden fazla hastayla ilgilenebilirken bir hastanın yalnızca bir adet hekimi vardır. Hekimler yeni atanmış olabileceği için hiç hastası olmayan hekimler bulunabilmektedir. Fakat her hastanın bağlı bulunduğu tam bir tane hekimi bulunmalıdır.

Rahatsızlık ile hekim arasında “Tedavi Etme” ilişkisi bulunmaktadır. Herhangi bir rahatsızlığı yalnızca bir adet hekim kontrol edebilmektedir. Bunun sebebi ise hastanın bağlı olduğu sadece bir hekimin bulunmasıdır. Her hekim ise birden fazla rahatsızlıkla ilgilenebilmektedir. Her rahatsızlığın incelenmesinde mutlaka bir hekim görevlendirilmeli, her hekimin ise kontrol ettiği birden fazla hasta olabileceği için birden fazla rahatsızlık olabilmektedir.

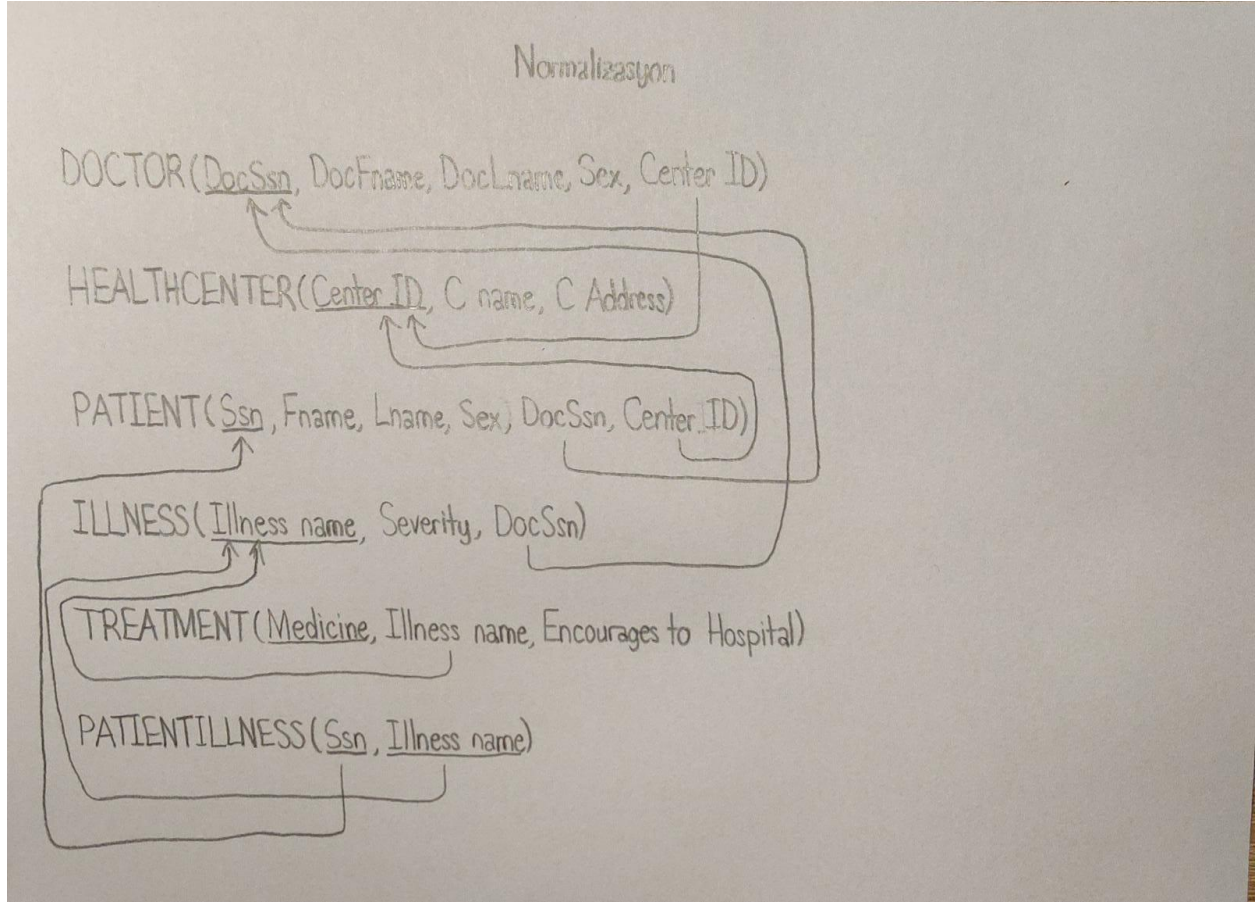
Hasta ile rahatsızlık arasındaki ilişki “Hasta Rahatsızlığı” olarak isimlendirilmiştir. Bir hasta, birden fazla rahatsızlık geçirebilmekte, bir rahatsızlıktan ise birden fazla hasta şikayetçi olabilmektedir. Herhangi bir hastanın bir rahatsızlığı bulunmak zorunda değildir. Herhangi bir hastalığı da aynı şekilde bir hasta yaşamak zorunda değildir.

Rahatsızlık ve tedavi arasında bulunan ilişki, tedaviyi tanımlayabilen varlık rahatsızlık olduğu için zayıf bir ilişkidir. Bir rahatsızlığın birden fazla tedavisi olabilmektedir. Aynı zamanda hiçbir tedavi olmama olasılığı da bulunmaktadır.

IV. Mantıksal tasarım ve şema diyagramları



*Erwin’de yapılan ER diyagramı şekildeki gibidir.



*İlişkisel modelleme, başka bir varlığa referans eden öznitelikleri gösterir.

V. Tasarımın Uyarlanması

Uyarlanan Yazılım Donanımı Ortamı Bilgileri

İşlemci: Intel(R) Core(TM) i7-7500U CPU @ 2.70GHz 2.90 GHz

Takılı RAM: 8,00 GB (kullanılabilir: 7,85 GB)

Sistem türü: 64 bit işletim sistemi, x64 tabanlı işlemci

VTYS Bilgileri

PostgreSQL ya da Postgres, özgür ve açık kaynak kodlu, SQL destekli bir ilişkisel veri tabanı yönetim sistemidir. pgAdmin, PostgreSQL için zengin özelliklere sahip en popüler açık kaynak yönetim ve geliştirme platformudur.

Projede veri tabanı yönetim sistemi olarak PostgreSQL'in 13.8.1 kullanılmaktadır. PostgreSQL 13, aşağıdakiler dahil birçok yeni özellik ve geliştirme içerir:

- B-ağacı dizin girişlerinin tekilleştirilmesinden alan tasarrufu ve performans kazanımları
- Toplamaları veya bölümlenmiş tabloları kullanan sorgular için iyileştirilmiş performans
- Genişletilmiş istatistikler kullanılırken daha iyi sorgu planlaması
- İndekslerin paralel vakumlanması
- Artımlı sıralama

SQL Tabloları

Veri tabanı, aşağıda da görüldüğü gibi dördü varlık, biri zayıf varlık, biri çoklu ilişki (M to N) olarak altı tablodan oluşmaktadır.

Tablolar sırasıyla;

	centerid [PK] integer	c_address character varying (60)	c_name character varying (60)
1	1	Maltepe, Güzaltan Sk....	Çankaya 1 Nolu Aile S...
2	2	Çukurambar, Öğretme...	Çankaya Çukurambar ...
3	3	Ertuğrulgazi, Süngüba...	Çankaya Şahin Asm

Oluşturulan sağlık ocağı (HEALTHCENTER) tablosu, kurumun eşsiz(unique) numarasını, adresini ve adını tutar. Sağlık ocakları birbirlerinden numaralarıyla ayrılır.

	docssn [PK] character (11)	docfname character varying (20)	doclname character varying (20)	centerid integer	sex character (1)
1	11112222001	Arzu	Çetin	1	F
2	11112222002	Mehmet	Ovacık	2	M
3	11112222003	Sevgi	Yılmaz	3	F

Her sağlık ocağında çalışan doktorlar (DOCTOR) vardır ve her doktorun, adı, soyadı, cinsiyeti, çalıştığı sağlık ocağının numarası ve herkesten farklı olan TC kimlik numarası tutulur.

	ssn [PK] character (11)	fname character varying (20)	lname character varying (20)	centerid integer	docssn character (11)	sex character (1)	visitdate date
1	11119999001	Barış	Akyol	1	11112222001	M	2022-03-17
2	11119999002	Murat	Başaran	2	11112222002	M	2022-07-27
3	11119999003	Selma	Deniz	3	11112222003	F	2021-12-25

Her hasta (PATIENT), tek bir sağlık ocağının içindeki tek bir doktora kayıtlıdır. Bu doktorun kimlik numarası ve sağlık ocağının numarası tutulur. Aynı zamanda hastaya ait TC kimlik numarası, ad, soy ad, cinsiyet ve son muayene günü tutulur.

	illnessname [PK] character varying (30)	severity integer	docssn character (11)
1	Karın ağrısı	6	11112222001
2	Soğuk Algınlığı	4	11112222002
3	Zatürre	[null]	11112222003

Hastalık (ILLNESS) varlığı, her hastanın geçirdiği hastalık adını ve ciddiyetini tutar. Hastalığın ciddiyeti 1'den 10'a şiddetin yüksekliğini belirtir.

	illnessname [PK] character varying (20)	ssn [PK] character (11)
1	Karın ağrısı	11119999001
2	Soğuk Algınlığı	11119999002
3	Zatürre	11119999003

Hasta-Hastalık (PATIENT_ILLNESS) arasındaki ilişki çoklu ilişkidir ve bir tabloda gösterilir. Her hastalığın teşhisi ve teşhisi koyan doktor bu tabloya dahildir.

	illnessname [PK] character varying (20)	medicine character varying (30)	encouragestohospital boolean
1	Karın ağrısı	Parol	false
2	Soğuk Algınlığı	Dolven	false
3	Zatürre	Klacid	true

Tedavi (TREATMENT), hastalığın tanımladığı (identifying) bir zayıf varlıktır. Bu yüzden ana anahtarı (primary key) hastalıktan foreign key olarak gelen hastalık adıdır. Öznitelikleri tedavi için verilen ilaç ve sağlık ocağındaki doktorun hastaneye sevk edip etmediğini tutan bir boolean'dır.

* Projenin ER diyagramına baęlı SQL ile yazılmıř řema, tablo oluřturma ve ilk veri y¼kleme kodu ara raporun sonuna eklenmiřtir. Veri tabanı ve tablolar PostgreSQL kullanılarak oluřturulmuřtur.

SQL File

```
CREATE SCHEMA HEALTHCENTER_SYSTEM;
```

```
CREATE TABLE HEALTHCENTER_SYSTEM.HEALTHCENTER
```

```
(  
    CenterID INT NOT NULL ,  
    C_Address VARCHAR(60) ,  
    C_name VARCHAR(60) ,  
    PRIMARY KEY (CenterID)
```

```
);
```

```
CREATE TABLE HEALTHCENTER_SYSTEM.DOCTOR
```

```
(  
    DocSsn CHAR(11) NOT NULL ,  
    DocFname VARCHAR(20) ,  
    DocLname VARCHAR(20) ,  
    CenterID INT NOT NULL ,  
    Sex CHAR(1) ,  
    PRIMARY KEY (DocSsn) ,  
    FOREIGN KEY (CenterID) REFERENCES HEALTHCENTER_SYSTEM.HEALTHCENTER(CenterID)
```

```
);
```

```
CREATE TABLE HEALTHCENTER_SYSTEM.PATIENT
```

```
(  
    Ssn CHAR(11) NOT NULL ,  
    Fname VARCHAR(20) ,  
    Lname VARCHAR(20) ,  
    CenterID INT NOT NULL ,  
    DocSsn CHAR(11) NOT NULL ,  
    Sex CHAR(1) ,  
    VisitDate date ,  
    PRIMARY KEY (Ssn) ,  
    FOREIGN KEY (CenterID) REFERENCES HEALTHCENTER_SYSTEM.HEALTHCENTER(CenterID),  
    FOREIGN KEY (DocSsn) REFERENCES HEALTHCENTER_SYSTEM.DOCTOR(DocSsn)
```

```
);
```

```

CREATE TABLE HEALTHCENTER_SYSTEM.ILLNESS
(
    Illnessname VARCHAR(30) NOT NULL ,
    Severity INT ,
    DocSsn CHAR(11) NOT NULL ,
    PRIMARY KEY (Illnessname) ,
    FOREIGN KEY (DocSsn) REFERENCES HEALTHCENTER_SYSTEM.DOCTOR(DocSsn)
);

CREATE TABLE HEALTHCENTER_SYSTEM.PATIENT_ILLNESS
(
    Illnessname VARCHAR(20) NOT NULL ,
    Ssn CHAR(11) NOT NULL ,
    FOREIGN KEY (Illnessname) REFERENCES HEALTHCENTER_SYSTEM.ILLNESS(Illnessname) ,
    FOREIGN KEY (Ssn) REFERENCES HEALTHCENTER_SYSTEM.PATIENT(Ssn) ,
    PRIMARY KEY (Ssn, Illnessname)
);

CREATE TABLE HEALTHCENTER_SYSTEM.TREATMENT
(
    Illnessname VARCHAR(20) NOT NULL ,
    Medicine VARCHAR(30) ,
    PRIMARY KEY (Illnessname),
    FOREIGN KEY (Illnessname) REFERENCES HEALTHCENTER_SYSTEM.ILLNESS(Illnessname)
);

--DATA INSERTION

INSERT INTO HEALTHCENTER_SYSTEM.HEALTHCENTER
VALUES (1,'Maltepe, Güzaltan Sk. No:2, 06570 Çankaya/Ankara','Çankaya 1 Nolu Aile Sağlığı Merkezi')
INSERT INTO HEALTHCENTER_SYSTEM.HEALTHCENTER
VALUES (2,'Çukurambar, Öğretmenler Cd. no:6, 06510 Çankaya/Ankara','Çankaya Çukurambar Aile Sağlık Merkezi')
INSERT INTO HEALTHCENTER_SYSTEM.HEALTHCENTER
VALUES (3,'Ertuğrulgazi, Süngübayırı Sk., 06590 Çankaya/Ankara','Çankaya Şahin Asm')
--
INSERT INTO HEALTHCENTER_SYSTEM.DOCTOR VALUES ('11112222001','Arzu','Çetin',1,'F');
INSERT INTO HEALTHCENTER_SYSTEM.DOCTOR VALUES ('11112222002','Mehmet','Ovacık',2,'M');

```

```

INSERT INTO HEALTHCENTER_SYSTEM.DOCTOR VALUES ('11112222003','Sevgi','Yılmaz',3,'F');

--

INSERT INTO HEALTHCENTER_SYSTEM.PATIENT
VALUES ('11119999001','Barış','Akyol',1,'11112222001','M','2022-03-17');

INSERT INTO HEALTHCENTER_SYSTEM.PATIENT
VALUES ('11119999002','Murat','Başaran',2,'11112222002','M','2022-07-27');

INSERT INTO HEALTHCENTER_SYSTEM.PATIENT
VALUES ('11119999003','Selma','Deniz',3,'11112222003','F','2021-12-25');

--

INSERT INTO HEALTHCENTER_SYSTEM.ILLNESS VALUES ('Karın ağrısı',6,'11112222001');
INSERT INTO HEALTHCENTER_SYSTEM.ILLNESS VALUES ('Soğuk Algınlığı',4,'11112222002');
INSERT INTO HEALTHCENTER_SYSTEM.ILLNESS VALUES ('Zatürre',NULL,'11112222003');

--

INSERT INTO HEALTHCENTER_SYSTEM.TREATMENT VALUES ('Karın ağrısı','Parol',false);
INSERT INTO HEALTHCENTER_SYSTEM.TREATMENT VALUES ('Soğuk Algınlığı','Dolven',false);
INSERT INTO HEALTHCENTER_SYSTEM.TREATMENT VALUES ('Zatürre','Klacid',true);

--

INSERT INTO HEALTHCENTER_SYSTEM.PATIENT_ILLNESS VALUES ('Karın ağrısı','11119999001');
INSERT INTO HEALTHCENTER_SYSTEM.PATIENT_ILLNESS VALUES ('Soğuk Algınlığı','11119999002');
INSERT INTO HEALTHCENTER_SYSTEM.PATIENT_ILLNESS VALUES ('Zatürre','11119999003');

--SHOW TABLE

SELECT * FROM healthcenter_system.healthcenter ORDER BY centerid ASC

SELECT * FROM healthcenter_system.DOCTOR

ORDER BY centerid ASC

SELECT * FROM healthcenter_system.PATIENT ORDER BY centerid ASC

SELECT * FROM healthcenter_system.ILLNESS ORDER BY DocSsn ASC

SELECT * FROM healthcenter_system.TREATMENT

SELECT * FROM healthcenter_system.PATIENT_ILLNESS

```