

요양원 관리 시스템 개선을 위한 DB Project -Final Report

Team 1

60201921 김준서

60211902 김성엽

60211647 김민욱

60231199 오동이

60200036 임채아

목 차

1. Group members and their roles
2. project organization and its purpose
3. User Requirements / Business Rules
 - 3.1. User Requirements
 - 3.2. Business Rules
 - 3.3. Important Questions to be Asked to the DB
4. Conceptual design: An (Enhanced) Entity-Relationship Diagram.
5. Logical design
 - 5.1. Third normal form (3NF)
 - 5.2. Relation Model
6. Physical Design/Data dictionary
7. SQL Statements
 - 7.1. DDL
 - 7.2. DML with Useful SQL Queries

1. Group members and their roles

김준서: ER 작성, Relational Model 작성, DML 작성, DDL 작성, 과제 제출

김민욱: ER 작성, Relational Model 작성, DDL 작성, DML 작성

김성엽: ER 작성, Relational Model 작성, DDL 작성, DML 작성

오동이: ER 작성, Relational Model 작성, DDL 작성, DML 작성

임채아: ER 작성, Relational Model 작성, DDL 작성, DML 작성

2. project organization and its purpose

프로젝트 조직 및 목적:

본 프로젝트는 요양원을 대상으로 하는 관리 시스템 개선을 목표로 한다. 팀원들은 데이터베이스 설계, 구현, 테스트, 문서화 등에 협력하며, 입소자의 건강 정보, 생활 지원 서비스 제공 내역, 그리고 직원 관리 등 요양원 운영 전반에 필요한 정보를 효율적으로 관리할 수 있는 데이터베이스 시스템을 구축하고자 한다. 특히 본 프로젝트는 직원 배치 및 근무 현황 파악, 업무 효율성 향상에 초점을 두어, 요양원 운영 체계 전반을 데이터 기반으로 개선하는 것을 목표로 한다.

비즈니스 시나리오:

현재 요양원에서는 다음과 같은 운영 프로세스가 이루어진다.

- 1) 입소자 건강 정보(혈압, 체온, 혈당 등) 및 약물 복용 기록 관리
- 2) 식사 및 일상생활 지원(청소, 세탁 등) 이력 관리
- 3) 직원 근무 일정, 출근 현황, 업무 배정 등 관리

이러한 정보가 주로 수기로 기록되고 문서나 파일 단위로 관리되다 보니, 다음과 같은 문제점이 발생한다.

- 1) 정보 접근성 및 관리 효율성 저하:

수기 기록은 특정 입소자의 건강 변화를 추적하거나, 직원별 근무 현황을 한눈에 파악하는 데 어려움을 초래한다. 직원들은 매번 문서나 별도의 파일을 뒤져봐야 하며, 이는 시간과 노력을 낭비하고, 긴급하게 인력 배치를 조정할 때도 신속한 대응을 어렵게 만든다.

2) 정보 누락 및 오류 가능성:

수기 기록은 정보 누락 또는 오기입이 발생하기 쉽다. 예를 들어, 특정 직원의 근무 시간이나 업무 이력이 누락되거나 잘못 기입될 경우, 관리자가 전체 인력 상황을 정확히 파악하기 어렵고, 이에 따른 인력 배치나 서비스 제공 품질에도 문제가 생길 수 있다.

데이터베이스 구축 필요성 및 기대효과:

1) 체계적이고 신뢰성 있는 정보 관리:

입소자 건강 정보와 생활 지원 서비스 내역, 직원 근무 현황과 배치 정보를 데이터베이스 상에 구조화함으로써, 담당자는 실시간으로 정확한 정보를 쉽게 조회할 수 있다. 이를 통해 관리 효율성과 정확성이 크게 향상된다.

2) 직원 관리 효율성 강화:

직원의 근무 일정, 담당 구역, 업무 배치 정보를 체계적으로 관리함으로써 인력 활용도를 최적화할 수 있다. 예를 들어, 일정 변동 상황에 유연하게 대처하고, 법적으로 요구되는 간호사(또는 간호조무사) 대 입소자 비율을 신속히 확인하며, 추가 인력이 필요한 상황에도 빠르게 대응할 수 있다.

3) 응급 상황 대응력 강화:

데이터베이스에 축적된 건강 정보와 인력 정보를 활용하면, 특정 입소자에게 이상 징후가 감지되었을 때 즉각적으로 해당 지역 담당 직원에게 알림을 전달할 수 있다. 이를 통해 응급 상황에 대한 신속한 대응이 가능해지고, 입소자의 안전을 강화한다.

종합적으로, 본 프로젝트에서 개발하는 데이터베이스 시스템은 기존의 수기 중심 업무 프로세스를 효율적이고 정확한 데이터 중심 방식으로 전환한다. 이를 통해 요양원의 정보 관리 수준이 향상되고, 직원 배치 및 관리 업무가 체계화되어 전반적인 운영 품질이 개선된다.

3. User Requirements / Business Rules

3.1. User Requirements

요양원은 입소자(Resident), 건강 기록(Health Record), 약물(Medication), 정기 검진 서비스(Regularly Clinic), 응급 상황(Emergency), 보호자(Guardian), 직원(Staff), 생활 지원 서비스(Life Support)로 관련된 정보를 관리한다.

입소자(Resident)는 ResidentID로 고유하게 식별되며, 이름, 성별, 생년월일, 키, 몸무게, 혈액형, 입소 날짜와 같은 정보가 기록된다. 각 입소자는 반드시 하나 이상의 건강 기록(Health Record)을 가져야 하며, 각각의 건강 기록은 반드시 하나의 입소자와 연결되어야 한다. 보호자(Guardian)는 GuardianID로 식별되며 이름, 성별, 전화번호, 입소자와의 관계, 주소(도로명, 도시, 주, 우편번호)를 가지고 있다. 입소자는 반드시 한 명의 보호자를 가져야 하며, 각 보호자는 최소 한 명의 입소자와 연결되어야 한다.

입소자의 건강 기록(Health Record)은 고유한 RecordID를 통해 관리되며, 체온, 혈압, 혈당, 날짜 그리고 약물(Medication) 정보와 연결된다. 약물은 주사 약물(Injection)과 경구 약물(Oral Medication)로만 존재하고 한가지 방식만 기록한다.

약물(Medication)은 MedicationID로 고유하게 식별되며, 부작용 유무(Side Effect Status), 투약 시간(Time), 처방 받은 병원의 정보를 가지고 있다. 약물(Medication)은 건강 상태를 기반으로 건강 상태에 맞는 약물이 처방된다. 주사 약물(Injection)은 약물이 주입된 부위(Injection Site)와 투약 방법(Injection Method)을 추가로 기록하며, 경구 약물(Oral Medication)은 식사 상태(Meal_State)와 제형(Form)을 기록한다. 하나의 건강 기록은 여러 약물을 포함할 수 있으며 약물이 없을 수 있다. 그리고 약물은 반드시 하나의 건강 기록과 연결되어야 한다.

직원(Staff)은 StaffID로 고유하게 식별되며, 입소자를 관리하고 지원하는 역할, 그리고 정기 진료 지원을 수행한다. 직원 정보에는 이름, 성별, 생년월일, 역할(Role), 전화번호, 고용 날짜(Hired Date)를 기록한다. 각 직원은 하나 이상의 입소자를 관리해야 하며 최근 입사한 직원은 입소자를 배정받지 않을 수 있다. 각 입소자는

반드시 최소 하나 이상 관리자에 의해 관리되며 여러 명의 직원으로부터 지원을 받을 수 있다.

정기 검진 서비스(Regularly Clinic)는 병원 측에서 요양원으로 방문 진료를 실시하며 Clinic_ID로 식별된다. 또한 정기 검진 서비스는 의사, 날짜, 병원, 처방 여부를 기록한다. 각 직원은 여러 정기 검진 서비스를 관리할 수 있으며 관리하지 않는 직원도 존재한다. 그리고 각 정기 검진 서비스는 반드시 하나의 직원에 의해 관리된다.

생활 지원 서비스(Life Support)은 Support_ID로 식별되며 각 입소자가 받는 식사(아침,점심,저녁), 청소 서비스, 세탁 서비스, 날짜를 기록한다. 각각의 입소자는 반드시 서비스를 받아야 하며, 여러 서비스를 받을 수 있지만, 각 서비스는 반드시 하나의 입소자를 지원해야한다.

응급 상황(Emergency)은 고유한 EmergencyID로 관리되며, 위급 상황 정도(Ktas_Level), 보호자 알림 여부(Guadian_Notify_Status), 날짜와 입원 여부(Hospitalization_Status)를 기록한다. 각 응급 상황은 반드시 하나의 입소자와 연결되며, 한 입소자는 여러 응급 상황을 가질 수 있다.

3.2. Business Rules

- 요양원은 법적으로 입소자 25명당 1명의 간호사(또는 간호조무사)를 배치해야 한다.
- 간호(조무)사는 매일 입소자의 건강상태를 점검하고 건강관리기록부를 작성하여 보관한다.
- 한달에 한 번 입소자 전부(단체로) 병원에 정기 진료를 받는다.
- 정기 진료 외 병원 진료를 전부 응급상황으로 판단하며 레벨에 따라 분류한다.
- 입소자 변동이 있을 시 입소자 정보를 Update 해야 한다.
- 위급 상황 정도(Ktas_Level)는 한국형 응급환자 분류도구를 사용한다.

KTAS(Korean Triage and Acuity Scale : 한국형 응급환자 분류도구)

단계	단계별 정의	대표적인 증상	진료우 선순위
KTAS 1	즉각적인 처치가 필요하며 생명이나 사지를 위협하는(또는 악화 가능성이 높은) 상태	심장마비, 무호흡, 음주와 관련되지 않은 무의식	최우선수 위
KTAS 2	생명 혹은 사지, 신체기능에 잠재적인 위험이 있으며 이에 대한 빠른 치료가 필요한 경우	심근경색, 뇌출혈, 뇌경색	2순위
KTAS 3	치료가 필요한 상태로 진행할 수도 있는 잠재적 가능성을 고려해야 하는 경우	호흡곤란(산소포화도 90%이상) 출혈을 동반한 설사	3순위
KTAS 4	환자의 나이, 통증이나 악화/합병증에 대한 가능성을 고려할 때 1~2시간 안에 처치나 재평가를 시행하면 되는 상태	38도 이상의 발열을 동반한 장염 복통을 동반한 요로감염	4순위
KTAS 5	긴급하지만 응급은 아닌 상태, 만성적인 문제로 인한 것이거나, 악화의 가능성이 낮은 상태	감기, 장염, 설사, 열상(상처)	5순위

3.3. Important Questions to be Asked to the DB

- 입소자 정보
 - 총 입소자의 리스트가 어떻게 되는가?
 - 지난 일주일동안 입소한 입소자의 리스트를 보여주세요
 - 000 입소자의 담당 관리자는 누구인가?
- 입소자 건강 관련 정보
 - 000 입소자의 지난 1개월간 건강 기록 리스트를 보여주세요
 - 000 입소자의 지난 1주일간 혈압값
- 정기 진료
 - 000 담당자가 정기 진료를 간 횟수
 - 11월에 정기 진료를 시행한 날짜
 - 정기 진료를 관리한 staff의 리스트
- 복약
 - 000 입소자의 1주일간 투약한 약물과 횟수를 보여주세요
 - 000 입소자의 12월에 투약한 약물 리스트
- 생활 지원 서비스
 - 000입소자의 1달간 생활 지원 서비스 리스트를 보여주세요

● 보호자 정보

- 000 입소자 보호자의 전화번호가 어떻게 되는가?
- 000 입소자 보호자의 집 주소가 어떻게 되는가?

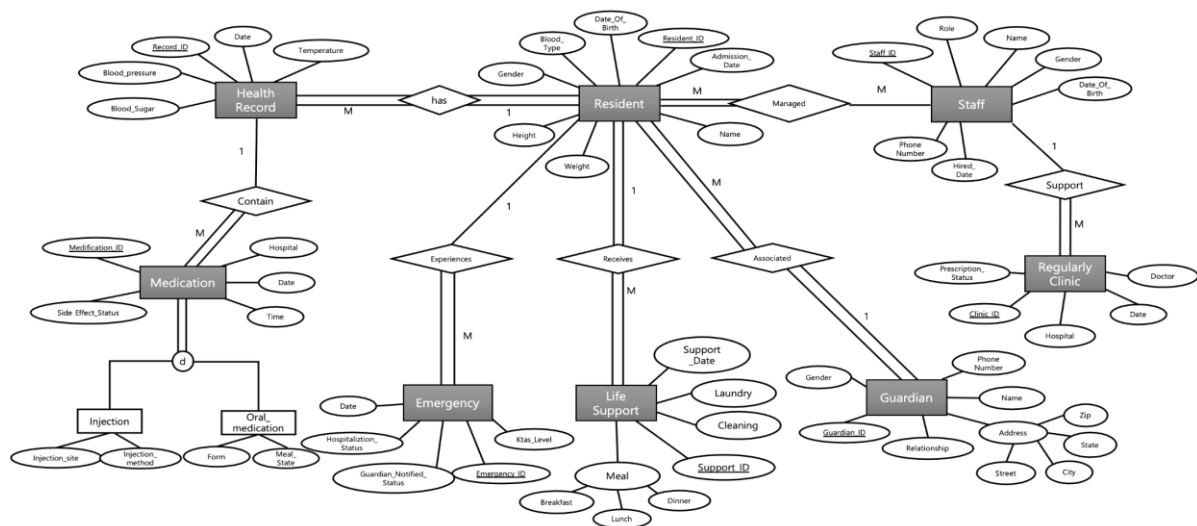
● 직원 정보

- 000 담당자가 담당하는 입소자 리스트를 보여주세요
- 000 담당자의 입사일
- 특정 Role에 해당하는 담당자 리스트

● 응급 상황

- 지난 1개월간 발생한 응급상황 리스트를 보여주세요
- ktas 1번에 해당하는 리스트를 전부 보여주세요

4. Conceptual design: An (Enhanced) Entity-Relationship Diagram.



5. Logical design

5.1. Third normal form (3NF)

- 1NF 만족 : 모든 속성이 Multi-valued, Composite이 아닌 원자값으로 구성되어 있어 1NF를 만족한다.
- 2NF 만족 : 1NF를 만족하며, non-Primary key가 Primary key에 완전히 종속된다.(Partial functional dependency가 없음)
- 3NF 만족 : 2NF를 만족하며 non-Primary key 간의 Transitive Functional dependency가 없다. (이행 함수 종속성이 없음)

5.2. Relation Model

Guardian (Guardian_ID, Name, Gender, Phone_Number, Relationship, Street, City, State, ZIP)

Resident (Resident_ID, Name, Gender, Date_Of_Birth, Height, Weight, Blood_Type, Admission_Date, Guardian_ID)

Foreign key (Guardian_ID) references Guardian (Guardian_ID)

Life Support (Support_ID, Support_Date, Laundry, Cleaning, Breakfast, Lunch, Dinner, Resident_ID)

Foreign key (Resident_ID) references Resident (Resident_ID)

Emergency (Emergency_ID, Date, Ktas_Level, Guardian_Notified_Status, Hospitalization_Status, Resident_ID)

Foreign key (Resident_ID) references Resident (Resident_ID)

Staff (Staff_ID, Role, Name, Gender, Date_of_Birth, Phone_Number, Hired_Date)

Managed (Resident_ID, Staff_ID)

Foreign key (Resident_ID) references Resident (Resident_ID)

Foreign key (Staff_ID) references Staff (Staff_ID)

Regularly Clinic (Clinic_ID, Hospital, Prescription_Status, Doctor, Date, Staff_ID)

Foreign key (Staff_ID) references Staff (Staff_ID)

Health Record (Record_ID, Blood_Pressure, Temperature, Blood_Sugar, Date, ResidentID)

Foreign key (Resident_ID) references Resident (Resident_ID)

Medication (Medication_ID, Side_Effect_Status, Date, Time, Hospital, Record_ID)

Foreign key (Record_ID) references HealthRecord (Record_ID)

Injection (Medication_ID, Injection_site, Injection_method)

Foreign key (Medication_ID) references Medication (Medication_ID)

Oral_Medication (Medication_ID, Form, Meal_Status)

Foreign key (Medication_ID) references Medication (Medication_ID)

6. Physical Design/Data dictionary

Guardian

Attribute	Description	Data Type	Constraint
<u>Guardian_ID</u>	Guardian ID	NUMBER	NOT NULL
Name	Guardian Name	VARCHAR2(100)	NOT NULL
Gender	Guardian Gender	CHAR(1)	CHECK (Gender IN ('M', 'F'))
Phone_Number	Guardian Phone Number	VARCHAR2(15)	
Relationship	Relationship to Resident	VARCHAR2(50)	
Street	Street Address	VARCHAR2(100)	
City	City	VARCHAR2(50)	
State	State	VARCHAR2(50)	
ZIP	ZIP Code	VARCHAR2(10)	

Resident

Attribute	Description	Data Type	Constraint
<u>Resident_ID</u>	Resident ID	NUMBER	NOT NULL
Name	Resident Name	VARCHAR2(100)	NOT NULL
Gender	Gender	CHAR(1)	CHECK (Gender IN ('M', 'F'))
Date_Of_Birth	Date of Birth	DATE	
Height	Height (cm)	NUMBER(5, 2)	
Weight	Weight (kg)	NUMBER(5, 2)	
Blood_Type	Blood Type	CHAR(3)	
Admission_Date	Admission Date	DATE	
Guardian_ID	Guardian ID	NUMBER	NOT NULL, FOREIGN KEY REFERENCES Guardian (Guardian_ID)

Health Record

Attribute	Description	Data Type	Constraint
<u>Record_ID</u>	Record ID	NUMBER	NOT NULL
Blood_Pressure	Blood Pressure	VARCHAR2(10)	
Temperature	Body Temperature	NUMBER(5, 2)	
Blood_Sugar	Blood Sugar	NUMBER(5, 2)	
Record_Date	Record_Date	DATE	
Resident_ID	Resident_ID	NUMBER	NOT NULL, CONSTRAINT fk_Resident_ID_1 FOREIGN KEY (Resident_ID) REFERENCES Resident (Resident_ID)

Medication

Attribute	Description	Data Type	Constraint
<u>Medication_ID</u>	Medication ID	NUMBER	NOT NULL
Side_Effect_Status	Side Effect Status	VARCHAR2(1)	CHECK (Side_Effect_Status IN ('Y', 'N'))
Date	Prescription Date	DATE	
Time	Prescription Time	TIMESTAMP	
Hospital	Prescribing Hospital	VARCHAR2(100)	
Record_ID	Record ID	NUMBER	NOT NULL, CONSTRAINT fk_Record_ID FOREIGN KEY (Record_ID) REFERENCES HealthRecord (Record_ID))

Injection

Attribute	Description	Data Type	Constraint
<u>Injection_Medication_ID</u>	Medication ID	NUMBER	NOT NULL, FOREIGN KEY REFERENCES Medication (Medication_ID)
Injection_Site	Injection Site	VARCHAR2(50)	
Injection_Method	Injection Method	VARCHAR2(50)	

Oral Medication

Attribute	Description	Data Type	Constraint
<u>Oral_Medication_ID</u>	Medication ID	NUMBER	NOT NULL, FOREIGN KEY REFERENCES Medication (Medication_ID)
Oral_Form	Oral Form	VARCHAR2(50)	
Meal_Status	Taken with Meal	VARCHAR2(1)	CHECK (Meal_Status IN ('Y', 'N'))

Life Support

Attribute	Description	Data Type	Constraint
<u>Support_ID</u>	Support ID	NUMBER	NOT NULL
Support_Date	Support Date	DATE	NOT NULL
Laundry	Laundry Service	VARCHAR2(10)	CHECK (Laundry IN ('Yes', 'No'))
Cleaning	Cleaning Service	VARCHAR2(10)	CHECK (Cleaning IN ('Yes', 'No'))
Breakfast	Breakfast Service	VARCHAR2(10)	CHECK (Breakfast IN ('Yes', 'No'))
Lunch	Lunch Service	VARCHAR2(10)	CHECK (Lunch IN ('Yes', 'No'))
Dinner	Dinner Service	VARCHAR2(10)	CHECK (Dinner IN ('Yes', 'No'))
Resident_ID	Resident ID	NUMBER	NOT NULL, FOREIGN KEY REFERENCES Resident (Resident_ID)

Staff

Attribute	Description	Data Type	Constraint
Staff_ID	Staff ID	VARCHAR(10)	NOT NULL
Role	Role	VARCHAR2(50)	NOT NULL, CHECK (Role IN ('Nurse', 'Social Worker'))
Name	Name	VARCHAR2(100)	NOT NULL
Gender	Gender	CHAR(1)	CHECK (Gender IN ('M', 'F'))
Date_Of_Birth	Date of Birth	DATE	
Phone_Number	Phone Number	VARCHAR(15)	
Hired_Date	Hired Date	DATE	

Emergency

Attribute	Description	Data Type	Constraint
<u>Emergency_ID</u>	Emergency ID	NUMBER	NOT NULL
Emergency_Date	Emergency Date	DATE	NOT NULL
Ktas_Level	KTAS Level	NUMBER(1)	CHECK (Ktas_Level BETWEEN 1 AND 5)
Guardian_Notified_Status	Guardian Notified Status	VARCHAR2(3)	CHECK (Guardian_Notified_Status IN ('Yes', 'No')), NOT NULL
Hospitalization_Status	Hospitalization Status	VARCHAR2(3)	CHECK (Hospitalization_Status IN ('Yes', 'No')), NOT NULL
Resident_ID	Resident ID	NUMBER	NOT NULL, FOREIGN KEY REFERENCES Resident (Resident_ID)

Regularly Clinic

Attribute	Description	Data Type	Constraint
<u>Clinic_ID</u>	Clinic ID	NUMBER	NOT NULL,
Hospital	Hospital Name	VARCHAR2(100)	NOT NULL
Prescription_Status	Prescription Status	VARCHAR2(1)	NOT NULL
Doctor	Doctor Name	VARCHAR2(100)	
Date	Clinic Date	DATE	
Staff_ID	Staff ID	VARCHAR2(10)	NOT NULL, FOREIGN KEY REFERENCES Staff (Staff_ID)

Manage

Attribute	Description	Data Type	Constraint
Resident_ID	Resident ID	NUMBER	NOT NULL, FOREIGN KEY → Resident(Resident_ID)
Staff_ID	Staff ID	VARCHAR2(10)	NOT NULL, FOREIGN KEY → Staff(Staff_ID)

7. SQL Statements

7.1. DDL

```
CREATE TABLE Guardian (  
  
    Guardian_ID NUMBER NOT NULL,  
  
    Name VARCHAR2(100) NOT NULL,  
  
    Gender CHAR(1) CHECK (Gender IN ('M', 'F')),  
  
    Phone_Number VARCHAR2(15),  
  
    Relationship VARCHAR2(50),  
  
    Street VARCHAR2(100),  
  
    City VARCHAR2(50),  
  
    State VARCHAR2(50),  
  
    ZIP VARCHAR2(10),  
  
    CONSTRAINT guardian_pk PRIMARY KEY (Guardian_ID)  
  
);
```

```
CREATE TABLE Resident (  
  
    Resident_ID NUMBER NOT NULL,  
  
    Name VARCHAR2(100) NOT NULL,  
  
    Gender CHAR(1) CHECK (Gender IN ('M', 'F')),  
  
    Date_Of_Birth DATE,  
  
    Height NUMBER(5, 2),  
  
    Weight NUMBER(5, 2),
```

```
Blood_Type CHAR(3),

Admission_Date DATE,

Guardian_ID NUMBER NOT NULL,

CONSTRAINT resident_pk PRIMARY KEY (Resident_ID),

CONSTRAINT fk_Guardian_ID FOREIGN KEY (Guardian_ID) REFERENCES Guardian
(Guardian_ID)

);
```

```
CREATE TABLE HealthRecord (

    Record_ID NUMBER NOT NULL,

    Blood_Pressure VARCHAR2(10),

    Temperature NUMBER(5, 2),

    Blood_Sugar NUMBER(5, 2),

    Record_Date DATE,

    Resident_ID NUMBER NOT NULL,

    CONSTRAINT Record_pk PRIMARY KEY (Record_ID),

    CONSTRAINT fk_Resident_ID_1 FOREIGN KEY (Resident_ID) REFERENCES Resident
(Resident_ID)

);
```

```
CREATE TABLE Medication (

    Medication_ID NUMBER NOT NULL,

    Side_Effect_Status VARCHAR2(1) CHECK (Side_Effect_Status IN ('Y', 'N')),
```

```
Prescription_Date DATE,  
  
Prescription_Time TIMESTAMP,  
  
Hospital VARCHAR2(100),  
  
Record_ID NUMBER NOT NULL,  
  
CONSTRAINT Medication_pk PRIMARY KEY (medication_ID),  
  
CONSTRAINT fk_Record_ID FOREIGN KEY (Record_ID) REFERENCES HealthRecord  
(Record_ID)  
);
```

```
CREATE TABLE Injection (  
  
Injection_Medication_ID NUMBER NOT NULL,  
  
Injection_Site VARCHAR2(50),  
  
Injection_Method VARCHAR2(50),  
  
CONSTRAINT Injection_pk PRIMARY KEY (Injection_Medication_ID),  
  
CONSTRAINT fk_Medication_ID_1 FOREIGN KEY (Injection_Medication_ID)  
REFERENCES Medication (Medication_ID)  
);
```

```
CREATE TABLE Oral_Medication (  
  
Oral_Medication_ID NUMBER NOT NULL,  
  
Oral_Form VARCHAR2(50),  
  
Meal_Status VARCHAR2(1) CHECK (Meal_Status IN ('Y', 'N')),  
  
CONSTRAINT Oral_Medication_pk PRIMARY KEY (Oral_Medication_ID),
```

```
CONSTRAINT fk_Medication_ID_2 FOREIGN KEY (Oral_Medication_ID) REFERENCES  
Medication (Medication_ID)  
);
```

```
CREATE TABLE Life_Support (  
  
    Support_ID NUMBER NOT NULL,  
  
    Support_Date DATE NOT NULL,  
  
    Laundry VARCHAR2(10) CHECK (Laundry IN ('Yes', 'No')),  
  
    Cleaning VARCHAR2(10) CHECK (Cleaning IN ('Yes', 'No')),  
  
    Breakfast VARCHAR2(10) CHECK (Breakfast IN ('Yes', 'No')),  
  
    Lunch VARCHAR2(10) CHECK (Lunch IN ('Yes', 'No')),  
  
    Dinner VARCHAR2(10) CHECK (Dinner IN ('Yes', 'No')),  
  
    Resident_ID NUMBER NOT NULL,  
  
    CONSTRAINT Support_pk PRIMARY KEY (Support_ID),  
  
    CONSTRAINT fk_resident_2 FOREIGN KEY (Resident_ID) REFERENCES  
Resident(Resident_ID)  
);
```

```
CREATE TABLE Staff (  
  
    Staff_ID VARCHAR(10) NOT NULL,  
  
    Role VARCHAR2(50) NOT NULL CHECK (Role IN ('Nurse', 'Social Worker')),  
  
    Name VARCHAR2(100) NOT NULL,  
  
    Gender CHAR(1) CHECK (Gender IN ('M', 'F')),
```

```
Date_Of_Birth DATE,  
  
Phone_Number VARCHAR(15),  
  
Hired_Date DATE,  
  
CONSTRAINT Staff_pk PRIMARY KEY (Staff_ID)  
  
);
```

```
CREATE TABLE Emergency (  
  
    Emergency_ID NUMBER NOT NULL,  
  
    Emergency_Date DATE NOT NULL,  
  
    Ktas_Level NUMBER(1) CHECK (Ktas_Level BETWEEN 1 AND 5),  
  
    Guardian_Notified_Status VARCHAR2(3) CHECK (Guardian_Notified_Status IN  
('Yes', 'No')) NOT NULL,  
  
    Hospitalization_Status VARCHAR2(3) CHECK (Hospitalization_Status IN ('Yes', 'No'))  
NOT NULL,  
  
    Resident_ID NUMBER NOT NULL,  
  
    CONSTRAINT Emergency_pk PRIMARY KEY (Emergency_ID),  
  
    CONSTRAINT fk_resident_3 FOREIGN KEY (Resident_ID) REFERENCES  
Resident(Resident_ID)  
  
);
```

```
CREATE TABLE Regularly_Clinic (  
  
    Clinic_ID NUMBER NOT NULL,  
  
    Hospital VARCHAR2(100) NOT NULL,  
  
    Prescription_Status VARCHAR2(1) NOT NULL,
```

```
Doctor VARCHAR2(100),

RDate DATE,

Staff_ID VARCHAR2(10) NOT NULL,

CONSTRAINT Clinic_pk PRIMARY KEY (Clinic_ID),

CONSTRAINT fk_clinic_staff FOREIGN KEY (Staff_ID) REFERENCES Staff(Staff_ID)

);
```

```
CREATE TABLE Manage (

    Resident_ID NUMBER NOT NULL,

    Staff_ID VARCHAR2(10) NOT NULL,

    PRIMARY KEY (Resident_ID, Staff_ID),

    CONSTRAINT fk_manage_resident FOREIGN KEY (Resident_ID) REFERENCES Resident(Resident_ID),

    CONSTRAINT fk_manage_staff FOREIGN KEY (Staff_ID) REFERENCES Staff(Staff_ID)

);
```

-- Guardian Table

```
INSERT INTO Guardian (Guardian_ID, Name, Gender, Phone_Number, Relationship,
Street, City, State, ZIP) VALUES

(1, '김민수', 'F', '01012345678', '어머니', '행복로 123번길', '서울', '서울특별시', '12345');

INSERT INTO Guardian (Guardian_ID, Name, Gender, Phone_Number, Relationship,
Street, City, State, ZIP) VALUES

(2, '이서현', 'M', '01023456789', '아버지', '사랑길 456번길', '부산', '부산광역시',
'23456');
```

INSERT INTO Guardian (Guardian_ID, Name, Gender, Phone_Number, Relationship, Street, City, State, ZIP) VALUES

(3, '박준영', 'M', '01034567890', '아들', '희망로 789번길', '서울', '서울특별시', '12345');

INSERT INTO Guardian (Guardian_ID, Name, Gender, Phone_Number, Relationship, Street, City, State, ZIP) VALUES

(4, '최지우', 'F', '01045678901', '딸', '기쁨로 101번길', '인천', '인천광역시', '34567');

INSERT INTO Guardian (Guardian_ID, Name, Gender, Phone_Number, Relationship, Street, City, State, ZIP) VALUES

(5, '정예은', 'F', '01056789012', '친구', '햇살로 102번길', '대구', '대구광역시', '56789');

INSERT INTO Guardian (Guardian_ID, Name, Gender, Phone_Number, Relationship, Street, City, State, ZIP) VALUES

(6, '문준호', 'M', '01067890123', '삼촌', '별빛로 103번길', '광주', '광주광역시', '67890');

INSERT INTO Guardian (Guardian_ID, Name, Gender, Phone_Number, Relationship, Street, City, State, ZIP) VALUES

(7, '신지아', 'F', '01078901234', '이모', '푸른길 104번길', '부산', '부산광역시', '23456');

INSERT INTO Guardian (Guardian_ID, Name, Gender, Phone_Number, Relationship, Street, City, State, ZIP) VALUES

(8, '한준혁', 'M', '01089012345', '형', '소망로 105번길', '대전', '대전광역시', '89012');

INSERT INTO Guardian (Guardian_ID, Name, Gender, Phone_Number, Relationship, Street, City, State, ZIP) VALUES

(9, '조유진', 'F', '01090123456', '누나', '바람길 106번길', '서울', '서울특별시', '12345');

INSERT INTO Guardian (Guardian_ID, Name, Gender, Phone_Number, Relationship, Street, City, State, ZIP) VALUES

(10, '이현우', 'M', '01101234567', '사촌', '바다로 107번길', '제주', '제주특별자치도', '01234');

INSERT INTO Guardian (Guardian_ID, Name, Gender, Phone_Number, Relationship, Street, City, State, ZIP) VALUES

(11, '최민아', 'F', '01112345678', '친구', '구름로 108번길', '서울', '서울특별시', '12345');

INSERT INTO Guardian (Guardian_ID, Name, Gender, Phone_Number, Relationship, Street, City, State, ZIP) VALUES

(12, '강민준', 'M', '01123456789', '이웃', '하늘길 109번길', '부산', '부산광역시', '23456');

INSERT INTO Guardian (Guardian_ID, Name, Gender, Phone_Number, Relationship, Street, City, State, ZIP) VALUES

(13, '오하늘', 'F', '01134567890', '누나', '달빛로 110번길', '인천', '인천광역시', '34567');

INSERT INTO Guardian (Guardian_ID, Name, Gender, Phone_Number, Relationship, Street, City, State, ZIP) VALUES

(14, '남현석', 'M', '01145678901', '아버지', '햇빛로 111번길', '대구', '대구광역시', '56789');

INSERT INTO Guardian (Guardian_ID, Name, Gender, Phone_Number, Relationship, Street, City, State, ZIP) VALUES

(15, '윤지수', 'F', '01156789012', '어머니', '들꽃로 112번길', '광주', '광주광역시', '67890');

INSERT INTO Guardian (Guardian_ID, Name, Gender, Phone_Number, Relationship, Street, City, State, ZIP) VALUES

(16, '백지훈', 'M', '01167890123', '할아버지', '평화로 113번길', '서울', '서울특별시', '12345');

INSERT INTO Guardian (Guardian_ID, Name, Gender, Phone_Number, Relationship, Street, City, State, ZIP) VALUES

(17, '임수정', 'F', '01178901234', '할머니', '행운로 114번길', '부산', '부산광역시', '23456');

INSERT INTO Guardian (Guardian_ID, Name, Gender, Phone_Number, Relationship, Street, City, State, ZIP) VALUES

(18, '홍승민', 'M', '01189012345', '삼촌', '사랑로 115번길', '대전', '대전광역시', '89012');

INSERT INTO Guardian (Guardian_ID, Name, Gender, Phone_Number, Relationship, Street, City, State, ZIP) VALUES

(19, '서지영', 'F', '01190123456', '이모', '별빛로 116번길', '광주', '광주광역시', '67890');

INSERT INTO Guardian (Guardian_ID, Name, Gender, Phone_Number, Relationship, Street, City, State, ZIP) VALUES

(20, '정태훈', 'M', '01201234567', '사촌', '소망길 117번길', '서울', '서울특별시', '12345');

INSERT INTO Guardian (Guardian_ID, Name, Gender, Phone_Number, Relationship, Street, City, State, ZIP) VALUES

(21, '한지윤', 'F', '01212345678', '조카', '희망로 118번길', '대구', '대구광역시', '56789');

INSERT INTO Guardian (Guardian_ID, Name, Gender, Phone_Number, Relationship, Street, City, State, ZIP) VALUES

(22, '배정우', 'M', '01223456789', '조카', '평화길 119번길', '서울', '서울특별시', '12345');

INSERT INTO Guardian (Guardian_ID, Name, Gender, Phone_Number, Relationship, Street, City, State, ZIP) VALUES

(23, '이하늘', 'F', '01234567890', '이웃', '하늘길 120번길', '인천', '인천광역시', '34567');

INSERT INTO Guardian (Guardian_ID, Name, Gender, Phone_Number, Relationship, Street, City, State, ZIP) VALUES

(24, '최태양', 'M', '01245678901', '친구', '구름길 121번길', '광주', '광주광역시', '67890');

INSERT INTO Guardian (Guardian_ID, Name, Gender, Phone_Number, Relationship, Street, City, State, ZIP) VALUES

(25, '강소희', 'F', '01256789012', '동료', '햇살길 122번길', '부산', '부산광역시', '23456');

-- Resident Table

```
INSERT INTO Resident (Resident_ID, Name, Gender, Date_Of_Birth, Height, Weight, Blood_Type, Admission_Date, Guardian_ID) VALUES
```

```
(1, '김철수', 'M', TO_DATE('1950-01-01', 'YYYY-MM-DD'), 175.5, 70.3, 'O+', TO_DATE('2023-01-10', 'YYYY-MM-DD'), 1);
```

```
INSERT INTO Resident (Resident_ID, Name, Gender, Date_Of_Birth, Height, Weight, Blood_Type, Admission_Date, Guardian_ID) VALUES
```

```
(2, '이영희', 'F', TO_DATE('1945-05-12', 'YYYY-MM-DD'), 160.2, 60.1, 'A+', TO_DATE('2023-01-15', 'YYYY-MM-DD'), 2);
```

```
INSERT INTO Resident (Resident_ID, Name, Gender, Date_Of_Birth, Height, Weight, Blood_Type, Admission_Date, Guardian_ID) VALUES
```

```
(3, '박민수', 'M', TO_DATE('1948-07-25', 'YYYY-MM-DD'), 180.0, 75.5, 'B+', TO_DATE('2023-02-10', 'YYYY-MM-DD'), 3);
```

```
INSERT INTO Resident (Resident_ID, Name, Gender, Date_Of_Birth, Height, Weight, Blood_Type, Admission_Date, Guardian_ID) VALUES
```

```
(4, '최은정', 'F', TO_DATE('1940-11-30', 'YYYY-MM-DD'), 165.3, 58.4, 'AB+', TO_DATE('2023-02-25', 'YYYY-MM-DD'), 4);
```

```
INSERT INTO Resident (Resident_ID, Name, Gender, Date_Of_Birth, Height, Weight, Blood_Type, Admission_Date, Guardian_ID) VALUES
```

```
(5, '홍길동', 'M', TO_DATE('1935-09-12', 'YYYY-MM-DD'), 170.8, 68.2, 'O-', TO_DATE('2023-03-05', 'YYYY-MM-DD'), 5);
```

```
INSERT INTO Resident (Resident_ID, Name, Gender, Date_Of_Birth, Height, Weight, Blood_Type, Admission_Date, Guardian_ID) VALUES
```

```
(6, '윤미경', 'F', TO_DATE('1958-02-10', 'YYYY-MM-DD'), 168.5, 65.0, 'A-', TO_DATE('2023-04-01', 'YYYY-MM-DD'), 6);
```

```
INSERT INTO Resident (Resident_ID, Name, Gender, Date_Of_Birth, Height, Weight, Blood_Type, Admission_Date, Guardian_ID) VALUES
```

```
(7, '김준호', 'M', TO_DATE('1947-03-15', 'YYYY-MM-DD'), 178.0, 80.5, 'B-', TO_DATE('2023-04-05', 'YYYY-MM-DD'), 7);
```

```
INSERT INTO Resident (Resident_ID, Name, Gender, Date_Of_Birth, Height, Weight, Blood_Type, Admission_Date, Guardian_ID) VALUES
```

```
(8, '이정아', 'F', TO_DATE('1952-06-25', 'YYYY-MM-DD'), 162.0, 55.5, 'O+', TO_DATE('2023-04-10', 'YYYY-MM-DD'), 8);
```

```
INSERT INTO Resident (Resident_ID, Name, Gender, Date_Of_Birth, Height, Weight, Blood_Type, Admission_Date, Guardian_ID) VALUES
```

```
(9, '박성호', 'M', TO_DATE('1944-09-12', 'YYYY-MM-DD'), 172.4, 72.2, 'AB-', TO_DATE('2023-04-15', 'YYYY-MM-DD'), 9);
```

```
INSERT INTO Resident (Resident_ID, Name, Gender, Date_Of_Birth, Height, Weight, Blood_Type, Admission_Date, Guardian_ID) VALUES
```

```
(10, '최윤희', 'F', TO_DATE('1956-10-08', 'YYYY-MM-DD'), 161.0, 59.0, 'A+', TO_DATE('2023-04-20', 'YYYY-MM-DD'), 10);
```

```
INSERT INTO Resident (Resident_ID, Name, Gender, Date_Of_Birth, Height, Weight, Blood_Type, Admission_Date, Guardian_ID) VALUES
```

```
(11, '정우진', 'M', TO_DATE('1942-11-23', 'YYYY-MM-DD'), 180.5, 85.3, 'B+', TO_DATE('2023-04-25', 'YYYY-MM-DD'), 11);
```

```
INSERT INTO Resident (Resident_ID, Name, Gender, Date_Of_Birth, Height, Weight, Blood_Type, Admission_Date, Guardian_ID) VALUES
```

```
(12, '한지민', 'F', TO_DATE('1954-12-01', 'YYYY-MM-DD'), 164.0, 60.0, 'O-', TO_DATE('2023-05-01', 'YYYY-MM-DD'), 12);
```

```
INSERT INTO Resident (Resident_ID, Name, Gender, Date_Of_Birth, Height, Weight, Blood_Type, Admission_Date, Guardian_ID) VALUES
```

```
(13, '남태식', 'M', TO_DATE('1951-01-15', 'YYYY-MM-DD'), 177.8, 75.0, 'A+', TO_DATE('2023-05-05', 'YYYY-MM-DD'), 13);
```

```
INSERT INTO Resident (Resident_ID, Name, Gender, Date_Of_Birth, Height, Weight, Blood_Type, Admission_Date, Guardian_ID) VALUES
```

```
(14, '서진아', 'F', TO_DATE('1949-03-10', 'YYYY-MM-DD'), 170.2, 58.6, 'AB+', TO_DATE('2023-05-10', 'YYYY-MM-DD'), 14);
```

```
INSERT INTO Resident (Resident_ID, Name, Gender, Date_Of_Birth, Height, Weight, Blood_Type, Admission_Date, Guardian_ID) VALUES
```

```
(15, '최성민', 'M', TO_DATE('1953-04-18', 'YYYY-MM-DD'), 182.5, 78.4, 'O+', TO_DATE('2023-05-15', 'YYYY-MM-DD'), 15);
```

```
INSERT INTO Resident (Resident_ID, Name, Gender, Date_Of_Birth, Height, Weight, Blood_Type, Admission_Date, Guardian_ID) VALUES
```

```
(16, '강민아', 'F', TO_DATE('1955-06-22', 'YYYY-MM-DD'), 160.4, 50.2, 'A-', TO_DATE('2023-05-20', 'YYYY-MM-DD'), 16);
```

```
INSERT INTO Resident (Resident_ID, Name, Gender, Date_Of_Birth, Height, Weight, Blood_Type, Admission_Date, Guardian_ID) VALUES
```

```
(17, '이석준', 'M', TO_DATE('1946-08-30', 'YYYY-MM-DD'), 174.3, 73.1, 'B+', TO_DATE('2023-05-25', 'YYYY-MM-DD'), 17);
```

```
INSERT INTO Resident (Resident_ID, Name, Gender, Date_Of_Birth, Height, Weight, Blood_Type, Admission_Date, Guardian_ID) VALUES
```

```
(18, '송지연', 'F', TO_DATE('1943-11-12', 'YYYY-MM-DD'), 167.6, 63.8, 'O-', TO_DATE('2023-06-01', 'YYYY-MM-DD'), 18);
```

```
INSERT INTO Resident (Resident_ID, Name, Gender, Date_Of_Birth, Height, Weight, Blood_Type, Admission_Date, Guardian_ID) VALUES
```

```
(19, '유현수', 'M', TO_DATE('1941-02-19', 'YYYY-MM-DD'), 176.9, 69.0, 'A+', TO_DATE('2023-06-05', 'YYYY-MM-DD'), 19);
```

```
INSERT INTO Resident (Resident_ID, Name, Gender, Date_Of_Birth, Height, Weight, Blood_Type, Admission_Date, Guardian_ID) VALUES
```

```
(20, '차예은', 'F', TO_DATE('1959-03-25', 'YYYY-MM-DD'), 162.8, 58.5, 'B-', TO_DATE('2023-06-10', 'YYYY-MM-DD'), 20);
```

```
INSERT INTO Resident (Resident_ID, Name, Gender, Date_Of_Birth, Height, Weight, Blood_Type, Admission_Date, Guardian_ID) VALUES
```

```
(21, '박준형', 'M', TO_DATE('1955-05-16', 'YYYY-MM-DD'), 183.0, 80.0, 'AB+', TO_DATE('2023-06-15', 'YYYY-MM-DD'), 21);
```

```
INSERT INTO Resident (Resident_ID, Name, Gender, Date_Of_Birth, Height, Weight, Blood_Type, Admission_Date, Guardian_ID) VALUES
```

```
(22, '김수진', 'F', TO_DATE('1958-07-29', 'YYYY-MM-DD'), 165.1, 59.7, 'O+', TO_DATE('2023-06-20', 'YYYY-MM-DD'), 22);
```

```
INSERT INTO Resident (Resident_ID, Name, Gender, Date_Of_Birth, Height, Weight, Blood_Type, Admission_Date, Guardian_ID) VALUES
```

```
(23, '이민호', 'M', TO_DATE('1962-09-03', 'YYYY-MM-DD'), 178.6, 76.8, 'A-', TO_DATE('2023-06-25', 'YYYY-MM-DD'), 23);
```

```
INSERT INTO Resident (Resident_ID, Name, Gender, Date_Of_Birth, Height, Weight, Blood_Type, Admission_Date, Guardian_ID) VALUES
```

```
(24, '최소연', 'F', TO_DATE('1965-10-12', 'YYYY-MM-DD'), 164.2, 61.0, 'B+', TO_DATE('2023-07-01', 'YYYY-MM-DD'), 24);
```

```
INSERT INTO Resident (Resident_ID, Name, Gender, Date_Of_Birth, Height, Weight, Blood_Type, Admission_Date, Guardian_ID) VALUES
```

```
(25, '정현우', 'M', TO_DATE('1957-11-22', 'YYYY-MM-DD'), 172.5, 70.0, 'O-', TO_DATE('2023-07-05', 'YYYY-MM-DD'), 25);
```

-- HealthRecord Table

```
INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar, Record_Date, Resident_ID) VALUES
```

```
(1, '120/80', 36.5, 95.0, TO_DATE('2023-01-12', 'YYYY-MM-DD'), 1);
```

```
INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar, Record_Date, Resident_ID) VALUES
```

```
(2, '130/85', 37.0, 100.5, TO_DATE('2023-01-17', 'YYYY-MM-DD'), 2);
```

```
INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar, Record_Date, Resident_ID) VALUES
```

```
(3, '125/78', 36.8, 98.5, TO_DATE('2023-02-12', 'YYYY-MM-DD'), 3);
```

INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar, Record_Date, Resident_ID) VALUES

(4, '140/90', 37.2, 105.0, TO_DATE('2023-02-20', 'YYYY-MM-DD'), 4);

INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar, Record_Date, Resident_ID) VALUES

(5, '115/75', 36.7, 92.0, TO_DATE('2023-03-02', 'YYYY-MM-DD'), 5);

INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar, Record_Date, Resident_ID) VALUES

(6, '120/80', 36.6, 97.0, TO_DATE('2023-03-10', 'YYYY-MM-DD'), 6);

INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar, Record_Date, Resident_ID) VALUES

(7, '128/82', 36.9, 102.0, TO_DATE('2023-03-15', 'YYYY-MM-DD'), 7);

INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar, Record_Date, Resident_ID) VALUES

(8, '122/78', 36.5, 94.5, TO_DATE('2023-03-18', 'YYYY-MM-DD'), 8);

INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar, Record_Date, Resident_ID) VALUES

(9, '135/88', 37.1, 101.2, TO_DATE('2023-03-22', 'YYYY-MM-DD'), 9);

INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar, Record_Date, Resident_ID) VALUES

(10, '118/76', 36.4, 93.5, TO_DATE('2023-03-25', 'YYYY-MM-DD'), 10);

INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar, Record_Date, Resident_ID) VALUES

(11, '126/80', 36.7, 99.3, TO_DATE('2023-04-02', 'YYYY-MM-DD'), 11);

INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar, Record_Date, Resident_ID) VALUES

(12, '130/84', 37.0, 104.0, TO_DATE('2023-04-06', 'YYYY-MM-DD'), 12);

INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar, Record_Date, Resident_ID) VALUES

(13, '124/79', 36.9, 98.0, TO_DATE('2023-04-11', 'YYYY-MM-DD'), 13);

INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar, Record_Date, Resident_ID) VALUES

(14, '132/85', 37.2, 107.5, TO_DATE('2023-04-14', 'YYYY-MM-DD'), 14);

INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar, Record_Date, Resident_ID) VALUES

(15, '120/78', 36.8, 96.0, TO_DATE('2023-04-17', 'YYYY-MM-DD'), 15);

INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar, Record_Date, Resident_ID) VALUES

(16, '121/77', 36.6, 94.5, TO_DATE('2023-04-20', 'YYYY-MM-DD'), 16);

INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar, Record_Date, Resident_ID) VALUES

(17, '129/81', 36.9, 99.2, TO_DATE('2023-04-23', 'YYYY-MM-DD'), 17);

INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar, Record_Date, Resident_ID) VALUES

(18, '133/86', 37.1, 103.0, TO_DATE('2023-04-26', 'YYYY-MM-DD'), 18);

INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar, Record_Date, Resident_ID) VALUES

(19, '122/80', 36.5, 95.8, TO_DATE('2023-04-30', 'YYYY-MM-DD'), 19);

INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar, Record_Date, Resident_ID) VALUES

(20, '125/79', 36.7, 97.5, TO_DATE('2023-05-02', 'YYYY-MM-DD'), 20);

```
INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar,  
Record_Date, Resident_ID) VALUES
```

```
(21, '118/74', 36.5, 92.0, TO_DATE('2023-05-05', 'YYYY-MM-DD'), 21);
```

```
INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar,  
Record_Date, Resident_ID) VALUES
```

```
(22, '130/84', 37.0, 100.5, TO_DATE('2023-05-08', 'YYYY-MM-DD'), 22);
```

```
INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar,  
Record_Date, Resident_ID) VALUES
```

```
(23, '125/79', 36.7, 96.5, TO_DATE('2023-05-12', 'YYYY-MM-DD'), 23);
```

```
INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar,  
Record_Date, Resident_ID) VALUES
```

```
(24, '135/88', 37.1, 105.0, TO_DATE('2023-05-15', 'YYYY-MM-DD'), 24);
```

```
INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar,  
Record_Date, Resident_ID) VALUES
```

```
(25, '121/75', 36.8, 94.0, TO_DATE('2023-05-18', 'YYYY-MM-DD'), 25);
```

```
INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar,  
Record_Date, Resident_ID) VALUES
```

```
(26, '125/82', 36.8, 98.5, TO_DATE('2023-06-05', 'YYYY-MM-DD'), 6);
```

```
INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar,  
Record_Date, Resident_ID) VALUES
```

```
(27, '118/78', 37.2, 92.0, TO_DATE('2023-06-15', 'YYYY-MM-DD'), 6);
```

```
INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar,  
Record_Date, Resident_ID)
```

```
VALUES (28, '120/80', 36.5, 90.0, TO_DATE('2023-07-10', 'YYYY-MM-DD'), 1);
```

```
INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar,  
Record_Date, Resident_ID)
```



```
VALUES (29, '122/84', 36.6, 92.5, TO_DATE('2023-08-05', 'YYYY-MM-DD'), 1);

INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar,
Record_Date, Resident_ID)

VALUES (30, '118/76', 36.7, 95.0, TO_DATE('2023-09-15', 'YYYY-MM-DD'), 1);

INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar,
Record_Date, Resident_ID)

VALUES (31, '125/85', 36.8, 98.0, TO_DATE('2023-07-20', 'YYYY-MM-DD'), 9);

INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar,
Record_Date, Resident_ID)

VALUES (32, '126/88', 36.9, 97.5, TO_DATE('2023-08-10', 'YYYY-MM-DD'), 9);

INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar,
Record_Date, Resident_ID)

VALUES (33, '124/83', 37.0, 96.5, TO_DATE('2023-09-05', 'YYYY-MM-DD'), 9);

INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar,
Record_Date, Resident_ID)

VALUES (34, '120/80', 37.0, 100.0, TO_DATE('2023-07-10', 'YYYY-MM-DD'), 22);

INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar,
Record_Date, Resident_ID)

VALUES (35, '110/75', 36.5, 90.0, TO_DATE('2023-08-01', 'YYYY-MM-DD'), 24);

INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar,
Record_Date, Resident_ID)

VALUES (36, '115/78', 36.8, 95.0, TO_DATE('2023-08-15', 'YYYY-MM-DD'), 24);

INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar,
Record_Date, Resident_ID)

VALUES (37, '125/85', 37.1, 102.0, TO_DATE('2023-09-01', 'YYYY-MM-DD'), 24);
```

```
INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar,  
Record_Date, Resident_ID)
```

```
VALUES (38, '117/79', 36.6, 93.0, TO_DATE('2023-09-10', 'YYYY-MM-DD'), 24);
```

```
INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar,  
Record_Date, Resident_ID)
```

```
VALUES (39, '122/80', 36.9, 97.0, TO_DATE('2023-10-01', 'YYYY-MM-DD'), 25);
```

```
INSERT INTO HealthRecord (Record_ID, Blood_Pressure, Temperature, Blood_Sugar,  
Record_Date, Resident_ID)
```

```
VALUES (40, '128/85', 37.2, 105.0, TO_DATE('2023-10-15', 'YYYY-MM-DD'), 25);
```

--Medication Table

```
INSERT INTO Medication (MEDICATION_ID, SIDE_EFFECT_STATUS,  
PRESCRIPTION_DATE, PRESCRIPTION_TIME, HOSPITAL, RECORD_ID)
```

```
VALUES (1, 'N', TO_DATE('2023-01-12', 'YYYY-MM-DD'), TO_TIMESTAMP('2023-01-12  
10:00', 'YYYY-MM-DD HH24:MI'), '서울병원', 1);
```

```
INSERT INTO Medication (MEDICATION_ID, SIDE_EFFECT_STATUS,  
PRESCRIPTION_DATE, PRESCRIPTION_TIME, HOSPITAL, RECORD_ID)
```

```
VALUES (2, 'Y', TO_DATE('2023-03-10', 'YYYY-MM-DD'), TO_TIMESTAMP('2023-03-10  
11:30', 'YYYY-MM-DD HH24:MI'), '중앙병원', 6);
```

```
INSERT INTO Medication (MEDICATION_ID, SIDE_EFFECT_STATUS,  
PRESCRIPTION_DATE, PRESCRIPTION_TIME, HOSPITAL, RECORD_ID)
```

```
VALUES (3, 'N', TO_DATE('2023-03-22', 'YYYY-MM-DD'), TO_TIMESTAMP('2023-03-22  
14:00', 'YYYY-MM-DD HH24:MI'), '세브란스병원', 9);
```

```
INSERT INTO Medication (MEDICATION_ID, SIDE_EFFECT_STATUS,  
PRESCRIPTION_DATE, PRESCRIPTION_TIME, HOSPITAL, RECORD_ID)
```

```
VALUES (4, 'Y', TO_DATE('2023-04-17', 'YYYY-MM-DD'), TO_TIMESTAMP('2023-04-17  
09:15', 'YYYY-MM-DD HH24:MI'), '서울병원', 15);
```

```
INSERT INTO Medication (MEDICATION_ID, SIDE_EFFECT_STATUS,  
PRESCRIPTION_DATE, PRESCRIPTION_TIME, HOSPITAL, RECORD_ID)
```

```
VALUES (5, 'N', TO_DATE('2023-05-08', 'YYYY-MM-DD'), TO_TIMESTAMP('2023-05-08  
13:45', 'YYYY-MM-DD HH24:MI'), '삼성병원', 22);
```

```
INSERT INTO Medication (MEDICATION_ID, SIDE_EFFECT_STATUS,  
PRESCRIPTION_DATE, PRESCRIPTION_TIME, HOSPITAL, RECORD_ID)
```

```
VALUES (6, 'N', TO_DATE('2023-05-15', 'YYYY-MM-DD'), TO_TIMESTAMP('2023-05-15  
15:45', 'YYYY-MM-DD HH24:MI'), '서울아산병원', 24);
```

```
INSERT INTO Medication (MEDICATION_ID, SIDE_EFFECT_STATUS,  
PRESCRIPTION_DATE, PRESCRIPTION_TIME, HOSPITAL, RECORD_ID)
```

```
VALUES (7, 'N', TO_DATE('2023-06-05', 'YYYY-MM-DD'), TO_TIMESTAMP('2023-05-20  
09:15', 'YYYY-MM-DD HH24:MI'), '서울병원', 26);
```

```
INSERT INTO Medication (MEDICATION_ID, SIDE_EFFECT_STATUS,  
PRESCRIPTION_DATE, PRESCRIPTION_TIME, HOSPITAL, RECORD_ID)
```

```
VALUES (8, 'Y', TO_DATE('2023-06-15', 'YYYY-MM-DD'), TO_TIMESTAMP('2023-06-18  
14:45', 'YYYY-MM-DD HH24:MI'), '중앙병원', 27);
```

```
INSERT INTO Medication (MEDICATION_ID, SIDE_EFFECT_STATUS,  
PRESCRIPTION_DATE, PRESCRIPTION_TIME, HOSPITAL, RECORD_ID)
```

```
VALUES (9, 'N', TO_DATE('2023-07-10', 'YYYY-MM-DD'), TO_TIMESTAMP('2023-07-10  
10:30', 'YYYY-MM-DD HH24:MI'), '서울병원', 28);
```

```
INSERT INTO Medication (MEDICATION_ID, SIDE_EFFECT_STATUS,  
PRESCRIPTION_DATE, PRESCRIPTION_TIME, HOSPITAL, RECORD_ID)
```

```
VALUES (10, 'Y', TO_DATE('2023-08-05', 'YYYY-MM-DD'), TO_TIMESTAMP('2023-08-05  
14:00', 'YYYY-MM-DD HH24:MI'), '중앙병원', 29);
```

```
INSERT INTO Medication (MEDICATION_ID, SIDE_EFFECT_STATUS,  
PRESCRIPTION_DATE, PRESCRIPTION_TIME, HOSPITAL, RECORD_ID)
```

```
VALUES (11, 'N', TO_DATE('2023-09-15', 'YYYY-MM-DD'), TO_TIMESTAMP('2023-09-15  
09:45', 'YYYY-MM-DD HH24:MI'), '세브란스병원', 30);
```

```
INSERT INTO Medication (MEDICATION_ID, SIDE_EFFECT_STATUS,  
PRESCRIPTION_DATE, PRESCRIPTION_TIME, HOSPITAL, RECORD_ID)
```

```
VALUES (12, 'Y', TO_DATE('2023-07-20', 'YYYY-MM-DD'), TO_TIMESTAMP('2023-07-20  
11:15', 'YYYY-MM-DD HH24:MI'), '서울병원', 31);
```

```
INSERT INTO Medication (MEDICATION_ID, SIDE_EFFECT_STATUS,  
PRESCRIPTION_DATE, PRESCRIPTION_TIME, HOSPITAL, RECORD_ID)
```

```
VALUES (13, 'N', TO_DATE('2023-08-10', 'YYYY-MM-DD'), TO_TIMESTAMP('2023-08-10  
13:30', 'YYYY-MM-DD HH24:MI'), '중앙병원', 32);
```

```
INSERT INTO Medication (MEDICATION_ID, SIDE_EFFECT_STATUS,  
PRESCRIPTION_DATE, PRESCRIPTION_TIME, HOSPITAL, RECORD_ID)
```

```
VALUES (14, 'Y', TO_DATE('2023-09-05', 'YYYY-MM-DD'), TO_TIMESTAMP('2023-09-05  
15:00', 'YYYY-MM-DD HH24:MI'), '세브란스병원', 33);
```

```
INSERT INTO Medication (MEDICATION_ID, SIDE_EFFECT_STATUS,  
PRESCRIPTION_DATE, PRESCRIPTION_TIME, HOSPITAL, RECORD_ID)
```

```
VALUES (15, 'N', TO_DATE('2023-07-10', 'YYYY-MM-DD'), TO_TIMESTAMP('2023-07-10  
10:30', 'YYYY-MM-DD HH24:MI'), '서울병원', 34);
```

```
INSERT INTO Medication (MEDICATION_ID, SIDE_EFFECT_STATUS,  
PRESCRIPTION_DATE, PRESCRIPTION_TIME, HOSPITAL, RECORD_ID)
```

```
VALUES (16, 'Y', TO_DATE('2023-08-01', 'YYYY-MM-DD'), TO_TIMESTAMP('2023-08-01  
09:00', 'YYYY-MM-DD HH24:MI'), '중앙병원', 35);
```

```
INSERT INTO Medication (MEDICATION_ID, SIDE_EFFECT_STATUS,  
PRESCRIPTION_DATE, PRESCRIPTION_TIME, HOSPITAL, RECORD_ID)
```

```
VALUES (17, 'N', TO_DATE('2023-08-15', 'YYYY-MM-DD'), TO_TIMESTAMP('2023-08-15  
14:30', 'YYYY-MM-DD HH24:MI'), '서울병원', 36);
```

```
INSERT INTO Medication (MEDICATION_ID, SIDE_EFFECT_STATUS,  
PRESCRIPTION_DATE, PRESCRIPTION_TIME, HOSPITAL, RECORD_ID)
```

```
VALUES (18, 'Y', TO_DATE('2023-09-01', 'YYYY-MM-DD'), TO_TIMESTAMP('2023-09-01  
11:45', 'YYYY-MM-DD HH24:MI'), '중앙병원', 37);
```

```
INSERT INTO Medication (MEDICATION_ID, SIDE_EFFECT_STATUS,  
PRESCRIPTION_DATE, PRESCRIPTION_TIME, HOSPITAL, RECORD_ID)
```

```
VALUES (19, 'N', TO_DATE('2023-09-10', 'YYYY-MM-DD'), TO_TIMESTAMP('2023-09-10  
16:20', 'YYYY-MM-DD HH24:MI'), '서울병원', 38);
```

```
INSERT INTO Medication (MEDICATION_ID, SIDE_EFFECT_STATUS,  
PRESCRIPTION_DATE, PRESCRIPTION_TIME, HOSPITAL, RECORD_ID)
```

```
VALUES (20, 'N', TO_DATE('2023-10-01', 'YYYY-MM-DD'), TO_TIMESTAMP('2023-10-01  
09:10', 'YYYY-MM-DD HH24:MI'), '중앙병원', 39);
```

```
INSERT INTO Medication (MEDICATION_ID, SIDE_EFFECT_STATUS,  
PRESCRIPTION_DATE, PRESCRIPTION_TIME, HOSPITAL, RECORD_ID)
```

```
VALUES (21, 'Y', TO_DATE('2023-10-15', 'YYYY-MM-DD'), TO_TIMESTAMP('2023-10-15  
13:45', 'YYYY-MM-DD HH24:MI'), '서울병원', 40);
```

```
INSERT INTO Medication (MEDICATION_ID, SIDE_EFFECT_STATUS,  
PRESCRIPTION_DATE, PRESCRIPTION_TIME, HOSPITAL, RECORD_ID)
```

```
VALUES (22, 'Y', TO_DATE('2024-12-10', 'YYYY-MM-DD'), TO_TIMESTAMP('2024-12-10  
18:20', 'YYYY-MM-DD HH24:MI'), '세브란스병원', 21);
```

-- Injection

```
INSERT INTO Injection (Injection_Medication_ID, Injection_Site, Injection_Method)
```

```
VALUES (1, '왼팔', '근육 주사');
```

```
INSERT INTO Injection (Injection_Medication_ID, Injection_Site, Injection_Method)
```

```
VALUES (3, '오른쪽 허벅지', '피하 주사');
```

```
INSERT INTO Injection (Injection_Medication_ID, Injection_Site, Injection_Method)
```

```
VALUES (5, '복부', '정맥 주사');
```

```
INSERT INTO Injection (Injection_Medication_ID, Injection_Site, Injection_Method)
```

```
VALUES (7, '엉덩이', '수액 주사');
```

```
INSERT INTO Injection (Injection_Medication_ID, Injection_Site, Injection_Method)
```

```
VALUES (9, '왼팔', '근육 주사');
```

```
INSERT INTO Injection (Injection_Medication_ID, Injection_Site, Injection_Method)
```

```
VALUES (11, '오른쪽 허벅지', '피하 주사');
```

```
INSERT INTO Injection (Injection_Medication_ID, Injection_Site, Injection_Method)
```

```
VALUES (13, '복부', '정맥 주사');
```

```
INSERT INTO Injection (Injection_Medication_ID, Injection_Site, Injection_Method)
```

```
VALUES (15, '엉덩이', '수액 주사');
```

```
INSERT INTO Injection (Injection_Medication_ID, Injection_Site, Injection_Method)
```

```
VALUES (17, '왼팔', '근육 주사');
```

```
INSERT INTO Injection (Injection_Medication_ID, Injection_Site, Injection_Method)
```

```
VALUES (19, '오른쪽 허벅지', '피하 주사');
```

```
INSERT INTO Injection (Injection_Medication_ID, Injection_Site, Injection_Method)
```

```
VALUES (21, '복부', '정맥 주사');
```

```
-- Oral Medication
```

```
INSERT INTO Oral_Medication (Oral_Medication_ID, Oral_Form, Meal_Status)
```

```
VALUES (2, '캡슐', 'Y');
```

```
INSERT INTO Oral_Medication (Oral_Medication_ID, Oral_Form, Meal_Status)
```

```
VALUES (4, '물약', 'Y');
```

```
INSERT INTO Oral_Medication (Oral_Medication_ID, Oral_Form, Meal_Status)
```

```
VALUES (6, '물약', 'N');
```

```
INSERT INTO Oral_Medication (Oral_Medication_ID, Oral_Form, Meal_Status)
```

```
VALUES (8, '캡슐', 'Y');
```

```
INSERT INTO Oral_Medication (Oral_Medication_ID, Oral_Form, Meal_Status)
```

```
VALUES (10, '물약', 'Y');
```

```
INSERT INTO Oral_Medication (Oral_Medication_ID, Oral_Form, Meal_Status)
```

```
VALUES (12, '캡슐', 'N');
```

```
INSERT INTO Oral_Medication (Oral_Medication_ID, Oral_Form, Meal_Status)
```

```
VALUES (14, '물약', 'Y');
```

```
INSERT INTO Oral_Medication (Oral_Medication_ID, Oral_Form, Meal_Status)
```

```
VALUES (16, '캡슐', 'Y');
```

```
INSERT INTO Oral_Medication (Oral_Medication_ID, Oral_Form, Meal_Status)
```

```
VALUES (18, '물약', 'N');
```

```
INSERT INTO Oral_Medication (Oral_Medication_ID, Oral_Form, Meal_Status)
```

```
VALUES (20, '캡슐', 'Y');
```

```
INSERT INTO Oral_Medication (Oral_Medication_ID, Oral_Form, Meal_Status)
```

```
VALUES (22, '물약', 'Y');
```

```
-- Life_support
```

```
INSERT INTO Life_Support (Support_ID, Support_Date, Laundry, Cleaning, Breakfast,  
Lunch, Dinner, Resident_ID)
```

```
VALUES (1, TO_DATE('2023-12-01', 'YYYY-MM-DD'), 'Yes', 'No', 'Yes', 'No', 'Yes', 1);
```

```
INSERT INTO Life_Support (Support_ID, Support_Date, Laundry, Cleaning, Breakfast,  
Lunch, Dinner, Resident_ID)
```

```
VALUES (2, TO_DATE('2023-12-02', 'YYYY-MM-DD'), 'No', 'Yes', 'No', 'Yes', 'Yes', 2);
```

```
INSERT INTO Life_Support (Support_ID, Support_Date, Laundry, Cleaning, Breakfast,  
Lunch, Dinner, Resident_ID)
```

```
VALUES (3, TO_DATE('2023-12-03', 'YYYY-MM-DD'), 'Yes', 'Yes', 'Yes', 'Yes', 'No', 3);
```

```
INSERT INTO Life_Support (Support_ID, Support_Date, Laundry, Cleaning, Breakfast,  
Lunch, Dinner, Resident_ID)
```

```
VALUES (4, TO_DATE('2023-12-04', 'YYYY-MM-DD'), 'No', 'No', 'Yes', 'Yes', 'Yes', 4);
```

```
INSERT INTO Life_Support (Support_ID, Support_Date, Laundry, Cleaning, Breakfast,  
Lunch, Dinner, Resident_ID)
```

```
VALUES (5, TO_DATE('2023-12-05', 'YYYY-MM-DD'), 'Yes', 'No', 'No', 'Yes', 'No', 5);
```

```
INSERT INTO Life_Support (Support_ID, Support_Date, Laundry, Cleaning, Breakfast,  
Lunch, Dinner, Resident_ID)
```

```
VALUES (6, TO_DATE('2023-12-06', 'YYYY-MM-DD'), 'Yes', 'Yes', 'Yes', 'Yes', 'Yes', 6);
```

```
INSERT INTO Life_Support (Support_ID, Support_Date, Laundry, Cleaning, Breakfast,  
Lunch, Dinner, Resident_ID)
```

```
VALUES (7, TO_DATE('2023-12-07', 'YYYY-MM-DD'), 'No', 'Yes', 'No', 'No', 'Yes', 7);
```

```
INSERT INTO Life_Support (Support_ID, Support_Date, Laundry, Cleaning, Breakfast,  
Lunch, Dinner, Resident_ID)
```

```
VALUES (8, TO_DATE('2023-12-08', 'YYYY-MM-DD'), 'Yes', 'No', 'Yes', 'Yes', 'Yes', 8);
```

```
INSERT INTO Life_Support (Support_ID, Support_Date, Laundry, Cleaning, Breakfast,  
Lunch, Dinner, Resident_ID)
```

```
VALUES (9, TO_DATE('2023-12-09', 'YYYY-MM-DD'), 'Yes', 'Yes', 'No', 'Yes', 'No', 9);
```

```
INSERT INTO Life_Support (Support_ID, Support_Date, Laundry, Cleaning, Breakfast,  
Lunch, Dinner, Resident_ID)
```


VALUES (10, TO_DATE('2023-12-10', 'YYYY-MM-DD'), 'No', 'Yes', 'Yes', 'No', 'Yes', 10);

INSERT INTO Life_Support (Support_ID, Support_Date, Laundry, Cleaning, Breakfast, Lunch, Dinner, Resident_ID)

VALUES (11, TO_DATE('2023-12-11', 'YYYY-MM-DD'), 'Yes', 'No', 'Yes', 'Yes', 'Yes', 11);

INSERT INTO Life_Support (Support_ID, Support_Date, Laundry, Cleaning, Breakfast, Lunch, Dinner, Resident_ID)

VALUES (12, TO_DATE('2023-12-12', 'YYYY-MM-DD'), 'No', 'Yes', 'No', 'Yes', 'Yes', 12);

INSERT INTO Life_Support (Support_ID, Support_Date, Laundry, Cleaning, Breakfast, Lunch, Dinner, Resident_ID)

VALUES (13, TO_DATE('2023-12-13', 'YYYY-MM-DD'), 'Yes', 'Yes', 'Yes', 'No', 'Yes', 13);

INSERT INTO Life_Support (Support_ID, Support_Date, Laundry, Cleaning, Breakfast, Lunch, Dinner, Resident_ID)

VALUES (14, TO_DATE('2023-12-14', 'YYYY-MM-DD'), 'No', 'No', 'No', 'Yes', 'No', 14);

INSERT INTO Life_Support (Support_ID, Support_Date, Laundry, Cleaning, Breakfast, Lunch, Dinner, Resident_ID)

VALUES (15, TO_DATE('2023-12-15', 'YYYY-MM-DD'), 'Yes', 'Yes', 'Yes', 'Yes', 'Yes', 15);

INSERT INTO Life_Support (Support_ID, Support_Date, Laundry, Cleaning, Breakfast, Lunch, Dinner, Resident_ID)

VALUES (16, TO_DATE('2023-12-16', 'YYYY-MM-DD'), 'Yes', 'Yes', 'Yes', 'Yes', 'No', 16);

INSERT INTO Life_Support (Support_ID, Support_Date, Laundry, Cleaning, Breakfast, Lunch, Dinner, Resident_ID)

VALUES (17, TO_DATE('2023-12-17', 'YYYY-MM-DD'), 'No', 'Yes', 'Yes', 'Yes', 'Yes', 17);

INSERT INTO Life_Support (Support_ID, Support_Date, Laundry, Cleaning, Breakfast, Lunch, Dinner, Resident_ID)

VALUES (18, TO_DATE('2023-12-18', 'YYYY-MM-DD'), 'Yes', 'Yes', 'No', 'No', 'Yes', 18);

```

INSERT INTO Life_Support (Support_ID, Support_Date, Laundry, Cleaning, Breakfast,
Lunch, Dinner, Resident_ID)

VALUES (19, TO_DATE('2023-12-19', 'YYYY-MM-DD'), 'No', 'No', 'Yes', 'Yes', 'No', 19);

INSERT INTO Life_Support (Support_ID, Support_Date, Laundry, Cleaning, Breakfast,
Lunch, Dinner, Resident_ID)

VALUES (20, TO_DATE('2023-12-20', 'YYYY-MM-DD'), 'Yes', 'Yes', 'Yes', 'No', 'Yes', 20);

INSERT INTO Life_Support (Support_ID, Support_Date, Laundry, Cleaning, Breakfast,
Lunch, Dinner, Resident_ID)

VALUES (21, TO_DATE('2023-12-21', 'YYYY-MM-DD'), 'No', 'Yes', 'Yes', 'Yes', 'Yes', 21);

INSERT INTO Life_Support (Support_ID, Support_Date, Laundry, Cleaning, Breakfast,
Lunch, Dinner, Resident_ID)

VALUES (22, TO_DATE('2023-12-22', 'YYYY-MM-DD'), 'Yes', 'No', 'Yes', 'Yes', 'Yes', 22);

INSERT INTO Life_Support (Support_ID, Support_Date, Laundry, Cleaning, Breakfast,
Lunch, Dinner, Resident_ID)

VALUES (23, TO_DATE('2023-12-23', 'YYYY-MM-DD'), 'Yes', 'Yes', 'Yes', 'No', 'No', 23);

INSERT INTO Life_Support (Support_ID, Support_Date, Laundry, Cleaning, Breakfast,
Lunch, Dinner, Resident_ID)

VALUES (24, TO_DATE('2023-12-24', 'YYYY-MM-DD'), 'No', 'No', 'Yes', 'Yes', 'Yes', 24);

INSERT INTO Life_Support (Support_ID, Support_Date, Laundry, Cleaning, Breakfast,
Lunch, Dinner, Resident_ID)

VALUES (25, TO_DATE('2023-12-25', 'YYYY-MM-DD'), 'Yes', 'Yes', 'No', 'Yes', 'No', 25);

-- Staff

INSERT INTO Staff (Staff_ID, Role, Name, Gender, Date_Of_Birth, Phone_Number,
Hired_Date)

```

VALUES ('N001', 'Nurse', '김아름', 'F', '1985-06-15', '010-1234-5678', '2023-01-01');

INSERT INTO Staff (Staff_ID, Role, Name, Gender, Date_Of_Birth, Phone_Number, Hired_Date)

VALUES ('S001', 'Social Worker', '이준호', 'M', '1990-08-22', '010-9876-5432', '2022-05-15');

INSERT INTO Staff (Staff_ID, Role, Name, Gender, Date_Of_Birth, Phone_Number, Hired_Date)

VALUES ('N002', 'Nurse', '박수진', 'F', '1978-03-11', '010-5555-1234', '2020-09-30');

INSERT INTO Staff (Staff_ID, Role, Name, Gender, Date_Of_Birth, Phone_Number, Hired_Date)

VALUES ('S002', 'Social Worker', '최민수', 'M', '1988-12-05', '010-1111-2222', '2021-11-20');

INSERT INTO Staff (Staff_ID, Role, Name, Gender, Date_Of_Birth, Phone_Number, Hired_Date)

VALUES ('N003', 'Nurse', '정다운', 'F', '1986-04-18', '010-3333-4444', '2021-06-10');

INSERT INTO Staff (Staff_ID, Role, Name, Gender, Date_Of_Birth, Phone_Number, Hired_Date)

VALUES ('S003', 'Social Worker', '한지훈', 'M', '1995-02-09', '010-8888-7777', '2023-08-15');

INSERT INTO Staff (Staff_ID, Role, Name, Gender, Date_Of_Birth, Phone_Number, Hired_Date)

VALUES ('N004', 'Nurse', '오예림', 'F', '1980-07-25', '010-6666-5555', '2019-05-01');

INSERT INTO Staff (Staff_ID, Role, Name, Gender, Date_Of_Birth, Phone_Number, Hired_Date)

VALUES ('N005', 'Nurse', '유하영', 'F', '1983-03-30', '010-4444-3333', '2018-11-19');

INSERT INTO Staff (Staff_ID, Role, Name, Gender, Date_Of_Birth, Phone_Number, Hired_Date)

```
VALUES ('S004', 'Social Worker', '김영준', 'M', '1992-10-17', '010-3333-2222', '2022-02-15');
```

```
INSERT INTO Staff (Staff_ID, Role, Name, Gender, Date_Of_Birth, Phone_Number, Hired_Date)
```

```
VALUES ('N006', 'Nurse', '이승재', 'M', '1975-01-01', '010-2222-1111', '2020-03-25');
```

```
INSERT INTO Staff (Staff_ID, Role, Name, Gender, Date_Of_Birth, Phone_Number, Hired_Date)
```

```
VALUES ('S005', 'Social Worker', '박소현', 'F', '1987-05-10', '010-1111-3333', '2021-07-30');
```

```
INSERT INTO Staff (Staff_ID, Role, Name, Gender, Date_Of_Birth, Phone_Number, Hired_Date)
```

```
VALUES ('N007', 'Nurse', '조민석', 'M', '1984-08-15', '010-8888-6666', '2020-12-11');
```

```
INSERT INTO Staff (Staff_ID, Role, Name, Gender, Date_Of_Birth, Phone_Number, Hired_Date)
```

```
VALUES ('S006', 'Social Worker', '윤가영', 'F', '1990-12-12', '010-6666-4444', '2019-09-14');
```

```
INSERT INTO Staff (Staff_ID, Role, Name, Gender, Date_Of_Birth, Phone_Number, Hired_Date)
```

```
VALUES ('N008', 'Nurse', '송하민', 'M', '1993-09-01', '010-4444-5555', '2023-04-05');
```

```
INSERT INTO Staff (Staff_ID, Role, Name, Gender, Date_Of_Birth, Phone_Number, Hired_Date)
```

```
VALUES ('S007', 'Social Worker', '임수정', 'F', '1989-06-21', '010-2222-4444', '2022-12-01');
```

```
INSERT INTO Staff (Staff_ID, Role, Name, Gender, Date_Of_Birth, Phone_Number, Hired_Date)
```

```
VALUES ('N009', 'Nurse', '이하연', 'F', '1998-03-02', '010-4124-1145', '2023-06-12');
```

--Emergency

```
INSERT INTO Emergency (Emergency_ID, Emergency_Date, Ktas_Level,  
Guardian_Notified_Status, Hospitalization_Status, Resident_ID)
```

```
VALUES (1, TO_DATE('2023-03-20', 'YYYY-MM-DD'), 3, 'Yes', 'No', 5);
```

```
INSERT INTO Emergency (Emergency_ID, Emergency_Date, Ktas_Level,  
Guardian_Notified_Status, Hospitalization_Status, Resident_ID)
```

```
VALUES (2, TO_DATE('2023-04-29', 'YYYY-MM-DD'), 4, 'No', 'Yes', 9);
```

```
INSERT INTO Emergency (Emergency_ID, Emergency_Date, Ktas_Level,  
Guardian_Notified_Status, Hospitalization_Status, Resident_ID)
```

```
VALUES (3, TO_DATE('2023-05-02', 'YYYY-MM-DD'), 2, 'Yes', 'No', 2);
```

```
INSERT INTO Emergency (Emergency_ID, Emergency_Date, Ktas_Level,  
Guardian_Notified_Status, Hospitalization_Status, Resident_ID)
```

```
VALUES (4, TO_DATE('2023-05-22', 'YYYY-MM-DD'), 5, 'No', 'Yes', 14);
```

```
INSERT INTO Emergency (Emergency_ID, Emergency_Date, Ktas_Level,  
Guardian_Notified_Status, Hospitalization_Status, Resident_ID)
```

```
VALUES (5, TO_DATE('2023-06-13', 'YYYY-MM-DD'), 3, 'Yes', 'No', 11);
```

```
INSERT INTO Emergency (Emergency_ID, Emergency_Date, Ktas_Level,  
Guardian_Notified_Status, Hospitalization_Status, Resident_ID)
```

```
VALUES (6, TO_DATE('2023-06-30', 'YYYY-MM-DD'), 4, 'Yes', 'Yes', 18);
```

```
INSERT INTO Emergency (Emergency_ID, Emergency_Date, Ktas_Level,  
Guardian_Notified_Status, Hospitalization_Status, Resident_ID)
```

```
VALUES (7, TO_DATE('2023-07-08', 'YYYY-MM-DD'), 2, 'No', 'No', 22);
```

```
INSERT INTO Emergency (Emergency_ID, Emergency_Date, Ktas_Level,  
Guardian_Notified_Status, Hospitalization_Status, Resident_ID)
```

```
VALUES (8, TO_DATE('2023-07-24', 'YYYY-MM-DD'), 1, 'Yes', 'Yes', 8);
```

```
INSERT INTO Emergency (Emergency_ID, Emergency_Date, Ktas_Level,  
Guardian_Notified_Status, Hospitalization_Status, Resident_ID)
```

```
VALUES (9, TO_DATE('2023-08-15', 'YYYY-MM-DD'), 1, 'Yes', 'No', 12);
```

```
INSERT INTO Emergency (Emergency_ID, Emergency_Date, Ktas_Level,  
Guardian_Notified_Status, Hospitalization_Status, Resident_ID)
```

```
VALUES (10, TO_DATE('2023-08-20', 'YYYY-MM-DD'), 1, 'No', 'Yes', 15);
```

```
--Regularly_Clinic
```

```
INSERT INTO Regularly_Clinic (Clinic_ID, Hospital, Prescription_Status, Doctor, RDate,  
Staff_ID)
```

```
VALUES (101, '중앙병원', 'T', '김현주', TO_DATE('2023-11-15', 'YYYY-MM-DD'), 'S001');
```

```
INSERT INTO Regularly_Clinic (Clinic_ID, Hospital, Prescription_Status, Doctor, RDate,  
Staff_ID)
```

```
VALUES (102, '중앙병원', 'F', '박재현', TO_DATE('2023-12-01', 'YYYY-MM-DD'), 'S002');
```

```
INSERT INTO Regularly_Clinic (Clinic_ID, Hospital, Prescription_Status, Doctor, RDate,  
Staff_ID)
```

```
VALUES (103, '서쪽클리닉', 'T', '이민호', TO_DATE('2023-10-20', 'YYYY-MM-DD'),  
'S003');
```

```
INSERT INTO Regularly_Clinic (Clinic_ID, Hospital, Prescription_Status, Doctor, RDate,  
Staff_ID)
```

```
VALUES (104, '동쪽의료원', 'T', '정예지', TO_DATE('2023-11-10', 'YYYY-MM-DD'),  
'S004');
```

```
INSERT INTO Regularly_Clinic (Clinic_ID, Hospital, Prescription_Status, Doctor, RDate,  
Staff_ID)
```

```
VALUES (105, '북부병원', 'T', '최은지', TO_DATE('2023-11-18', 'YYYY-MM-DD'), 'S005');
```

```
INSERT INTO Regularly_Clinic (Clinic_ID, Hospital, Prescription_Status, Doctor, RDate, Staff_ID)
```

```
VALUES (106, '서울중앙병원', 'F', '한유진', TO_DATE('2023-11-25', 'YYYY-MM-DD'), 'S006');
```

```
INSERT INTO Regularly_Clinic (Clinic_ID, Hospital, Prescription_Status, Doctor, RDate, Staff_ID)
```

```
VALUES (107, '서쪽클리닉', 'T', '오민영', TO_DATE('2023-09-15', 'YYYY-MM-DD'), 'S007');
```

```
INSERT INTO Regularly_Clinic (Clinic_ID, Hospital, Prescription_Status, Doctor, RDate, Staff_ID)
```

```
VALUES (108, '동쪽의료원', 'T', '윤지호', TO_DATE('2023-12-02', 'YYYY-MM-DD'), 'S001');
```

```
INSERT INTO Regularly_Clinic (Clinic_ID, Hospital, Prescription_Status, Doctor, RDate, Staff_ID)
```

```
VALUES (109, '남부클리닉', 'T', '강수연', TO_DATE('2023-11-12', 'YYYY-MM-DD'), 'S002');
```

```
INSERT INTO Regularly_Clinic (Clinic_ID, Hospital, Prescription_Status, Doctor, RDate, Staff_ID)
```

```
VALUES (110, '서울중앙병원', 'F', '최진우', TO_DATE('2023-11-19', 'YYYY-MM-DD'), 'S003');
```

```
INSERT INTO Regularly_Clinic (Clinic_ID, Hospital, Prescription_Status, Doctor, RDate, Staff_ID)
```

```
VALUES (111, '서쪽클리닉', 'T', '임다연', TO_DATE('2023-11-07', 'YYYY-MM-DD'), 'S004');
```

```
INSERT INTO Regularly_Clinic (Clinic_ID, Hospital, Prescription_Status, Doctor, RDate, Staff_ID)
```

```
VALUES (112, '동쪽의료원', 'T', '조성윤', TO_DATE('2023-10-15', 'YYYY-MM-DD'), 'S005');
```

```
INSERT INTO Regularly_Clinic (Clinic_ID, Hospital, Prescription_Status, Doctor, RDate, Staff_ID)
```

```
VALUES (113, '북부병원', 'T', '황지현', TO_DATE('2023-11-03', 'YYYY-MM-DD'), 'S006');
```

```
INSERT INTO Regularly_Clinic (Clinic_ID, Hospital, Prescription_Status, Doctor, RDate, Staff_ID)
```

```
VALUES (114, '남부클리닉', 'T', '이정원', TO_DATE('2023-11-30', 'YYYY-MM-DD'), 'S007');
```

```
INSERT INTO Regularly_Clinic (Clinic_ID, Hospital, Prescription_Status, Doctor, RDate, Staff_ID)
```

```
VALUES (115, '서울중앙병원', 'F', '박민정', TO_DATE('2023-11-25', 'YYYY-MM-DD'), 'S001');
```

```
--Manage
```

```
INSERT INTO Manage (Resident_ID, Staff_ID)
```

```
VALUES (1, 'N001');
```

```
INSERT INTO Manage (Resident_ID, Staff_ID)
```

```
VALUES (2, 'N005');
```

```
INSERT INTO Manage (Resident_ID, Staff_ID)
```

```
VALUES (3, 'N006');
```

```
INSERT INTO Manage (Resident_ID, Staff_ID)
```

```
VALUES (4, 'N003');
```

```
INSERT INTO Manage (Resident_ID, Staff_ID)
```

```
VALUES (5, 'N004');
```

```
INSERT INTO Manage (Resident_ID, Staff_ID)
```


VALUES (6, 'N002');

INSERT INTO Manage (Resident_ID, Staff_ID)

VALUES (7, 'N002');

INSERT INTO Manage (Resident_ID, Staff_ID)

VALUES (8, 'N005');

INSERT INTO Manage (Resident_ID, Staff_ID)

VALUES (9, 'N001');

INSERT INTO Manage (Resident_ID, Staff_ID)

VALUES (10, 'N003');

INSERT INTO Manage (Resident_ID, Staff_ID)

VALUES (11, 'N004');

INSERT INTO Manage (Resident_ID, Staff_ID)

VALUES (12, 'N005');

INSERT INTO Manage (Resident_ID, Staff_ID)

VALUES (13, 'N006');

INSERT INTO Manage (Resident_ID, Staff_ID)

VALUES (14, 'N007');

INSERT INTO Manage (Resident_ID, Staff_ID)

VALUES (15, 'N001');

INSERT INTO Manage (Resident_ID, Staff_ID)

VALUES (16, 'N007');

INSERT INTO Manage (Resident_ID, Staff_ID)

VALUES (17, 'N004');

INSERT INTO Manage (Resident_ID, Staff_ID)

VALUES (18, 'N003');

INSERT INTO Manage (Resident_ID, Staff_ID)

VALUES (19, 'N002');

INSERT INTO Manage (Resident_ID, Staff_ID)

VALUES (20, 'N002');

INSERT INTO Manage (Resident_ID, Staff_ID)

VALUES (21, 'N005');

INSERT INTO Manage (Resident_ID, Staff_ID)

VALUES (22, 'N006');

INSERT INTO Manage (Resident_ID, Staff_ID)

VALUES (23, 'N007');

INSERT INTO Manage (Resident_ID, Staff_ID)

VALUES (24, 'N002');

INSERT INTO Manage (Resident_ID, Staff_ID)

VALUES (25, 'N004');

7.2. DML with Useful SQL Queries

1. 약을 투여한 입소자와 몇번 투여 했는지

```
SELECT

    Resident.Resident_ID,

    Resident.Name AS Resident_Name,

    COUNT(Medication.Medication_ID) AS Total_Medication_Count -- 투약 횟수

FROM

    Resident,

    HealthRecord,

    Medication

WHERE

    Resident.Resident_ID = HealthRecord.Resident_ID

    AND HealthRecord.Record_ID = Medication.Record_ID

GROUP BY

    Resident.Resident_ID, Resident.Name

ORDER BY

    Resident.Resident_ID;
```

대답하는 내용:

이 쿼리는 모든 입소자에 대해, 각 입소자의 약물 투여 횟수를 보여줍니다.
입소자별로 약물이 투여된 횟수를 확인하여 전체 약물 사용 현황을 파악할 수 있습니다.

중요성과 유용성:

- 약물 사용 패턴 분석: 모든 입소자가 받은 약물과 투여 횟수를 확인함으로써, 시설 내 약물 사용 패턴을 분석하고, 과다 투약이나 투약 누락을 방지할 수 있습니다.

2. 000 담당자가 담당하는 입소자 리스트를 보여주세요 -- 이 쿼리는 특정 직원이 관리 중인 입소자의 ID와 이름을 보여줍니다.

SELECT

Manage.Staff_ID,

Staff.Name AS Staff_Name,

Resident.Resident_ID,

Resident.Name AS Resident_Name

FROM

Manage, Staff, Resident

WHERE

Manage.Staff_ID = Staff.Staff_ID

AND Manage.Resident_ID = Resident.Resident_ID

AND Staff.Name = '김아름'; -- 특정 직원 이름

대답하는 내용:

이 쿼리는 특정 직원이 관리하고 있는 입소자들의 ID와 이름을 조회합니다. 예시로 "김아름"이라는 직원이 담당하는 입소자 리스트를 반환합니다.

중요성과 유용성:

- 책임 분배와 업무 효율성 분석: 직원별로 관리하는 입소자들을 명확히 파악하여 각 직원의 업무 분담을 조정하고, 효율성을 높일 수 있습니다.

- 업무 최적화: 담당 직원의 업무 과중 여부를 확인하고, 필요한 경우 업무 재조정이나 추가 배정이 가능하게 합니다.
- 업무 성과 분석: 담당한 입소자 수와 관리 정도를 바탕으로 직원의 업무 성과를 분석하고 평가할 수 있습니다.

3. staff가 정기진료를 관리한 횟수(전체 리스트)

```
SELECT

    Staff.Staff_ID,

    Staff.Name AS Staff_Name,

    COUNT(*) AS Regular_Checkup_Count

FROM

    Regularly_Clinic, Staff

WHERE

    Regularly_Clinic.Staff_ID = Staff.Staff_ID

GROUP BY

    Staff.Staff_ID, Staff.Name;
```

대답하는 내용:

이 쿼리는 모든 직원들이 관리한 정기 진료 서비스의 횟수를 반환합니다. 각 직원별로 관리한 정기 진료의 횟수와 해당 직원의 이름을 확인할 수 있습니다.

중요성과 유용성:

- 업무 배치 최적화: 직원별로 정기 진료를 관리한 횟수를 파악하여 업무 배치를 최적화할 수 있습니다.
- 직원 기여도 평가: 각 직원이 관리한 정기 진료 횟수를 기반으로 기여도를 평가하고, 업무 분배를 개선할 수 있습니다.

- 업무 성과 관리: 직원들의 업무 성과를 명확히 파악하여 필요시 교육이나 재조정을 통한 성과 향상이 가능합니다.

4. 정현우 입소자의 담당 관리자는 누구인가? -- 이 쿼리는 특정 입소자의 담당 관리자의 이름과 연락처를 보여줍니다.

```
SELECT Manage.Staff_ID, Staff.Name AS Staff_Name, staff.Phone_Number,  
resident.name as resident_Name
```

```
FROM
```

```
Manage, staff, resident
```

```
where
```

```
Manage.Staff_ID = Staff.Staff_ID
```

```
and manage.resident_ID = resident.resident_id
```

```
and resident.name = '정현우';
```

대답하는 내용:

이 쿼리는 특정 입소자의 담당 관리자인 직원의 이름과 연락처를 조회합니다. 예시로 "정현우"라는 입소자의 담당 관리자를 확인할 수 있습니다.

중요성과 유용성:

- 신속한 대응: 입소자에 대한 긴급한 상황이나 중요한 결정을 내려야 할 때 담당 관리자의 연락처를 즉시 확인하여 빠르게 조치할 수 있습니다.
- 업무 효율성 향상: 담당 관리자가 누구인지 명확히 파악함으로써, 입소자 관리 및 지원 업무가 원활하게 이루어지도록 할 수 있습니다.
- 입소자 맞춤형 지원: 담당 관리자의 정보 제공을 통해 입소자와 보호자와의 의사소통을 강화하고, 맞춤형 의료 서비스를 제공할 수 있습니다.

5. ktas 1번 응급상황을 겪은 입소자 리스트를 전부 보여주세요 -- 이 쿼리는 KTAS 레벨 1 응급 상황을 경험한 입소자의 ID와 이름을 보여줍니다.

```
SELECT

    Emergency.Resident_ID,

    Resident.Name AS Resident_Name,

    Emergency.Emergency_Date,

    Emergency.Guardian_Notified_Status,

    Emergency.Hospitalization_Status

FROM

    Emergency, resident

WHERE

    Emergency.Resident_ID = Resident.Resident_ID

    and Emergency.Ktas_Level = 1;
```

대답하는 내용:

이 쿼리는 KTAS(응급실 분류 시스템) 레벨 1로 분류된 응급 상황을 경험한 입소자들의 ID, 이름, 응급 상황 발생일, 보호자 알림 여부, 입원 여부를 보여줍니다.

중요성과 유용성:

- 응급 대응 개선: KTAS 1번 응급 상황을 겪은 입소자들을 파악함으로써 시설의 응급 대응 프로토콜을 개선하고 강화할 수 있습니다.
- 위험 관리: 응급 상황이 빈번한 입소자를 추적하여 위험 요소를 사전에 식별하고 예방할 수 있습니다.
- 의료 서비스 품질 개선: 응급 상황에 대한 데이터 분석을 통해 의료 서비스의 품질을 높이고, 응급 상황에 대한 대처 능력을 향상시킬 수 있습니다.

6. "정우진 입소자의 응급상황 리스트"

```
SELECT  
  
Resident.Resident_ID, Resident.Name AS Resident_Name,  
Emergency.Emergency_Date, Emergency.Ktas_Level,  
Emergency.Hospitalization_Status  
  
FROM  
  
Resident, Emergency  
  
WHERE  
  
Resident.Resident_ID = Emergency.Resident_ID  
  
and Resident.NAME = '정우진';
```

대답하는 내용: 특정한 입소자(예: Resident_ID = '000')가 과거에 겪었던 응급상황(Emergency) 기록을 보여준다. 응급상황 발생일, 긴급도(Ktas_Level), 보호자 알림 여부, 입원 여부 등의 정보를 조회할 수 있다.

중요성과 유용성:

- 개별 입소자 관리 측면: 특정 입소자의 응급상황 히스토리를 통해 해당 입소자의 건강 상태 변화를 추적하고, 향후 응급 대처 계획 수립에 활용할 수 있다.
- 응급 대응 프로토콜 개선: 응급상황 빈도나 유형을 파악함으로써, 시설의 응급 대처 방침을 재정비하거나 개선하는 근거자료로 활용할 수 있다.

7. "남태식 입소자가 받은 생활 지원 서비스 리스트"

SELECT

Resident.Resident_ID, Resident.Name AS Resident_Name,
Life_Support.Support_ID, Life_Support.Support_Date, Life_Support.Laundry,
Life_Support.Cleaning, Life_Support.Breakfast, Life_Support.Lunch,
Life_Support.Dinner

FROM

Resident, Life_Support

WHERE

Resident.Resident_ID = Life_Support.Resident_ID AND Resident.Name = '남태식';

대답하는 내용: 특정 입소자(Resident_ID = '000')에게 제공되었던 생활 지원 서비스(Life Support) 내역을 보여준다. 날짜별로 제공된 식사, 청소, 세탁 등의 서비스를 조회할 수 있다.

중요성과 유용성:

- 서비스 품질 관리: 특정 입소자에게 어떤 서비스가 제공되었는지 확인함으로써 서비스 누락 여부나 만족도를 점검할 수 있다.
- 서비스 기록 및 청구 관리: 각 입소자별로 제공된 서비스를 기록하여 비용 청구나 회계 관리 시에 증빙 자료로 활용할 수 있다.

8. "이영희 입소자의 보호자 정보"

```
SELECT

Resident .Resident_ID, Resident .Name AS Resident_Name, Guardian.Guardian_ID,
Guardian.Name AS Guardian_Name, Guardian.Relationship,
Guardian.Phone_Number

FROM

Resident, Guardian

WHERE

Resident.Guardian_ID = Guardian.Guardian_ID

and Resident.Name = '이영희';
```

대답하는 내용: 특정 입소자의 보호자(Guardian) 정보를 조회한다. 보호자의 이름, 전화번호, 주소, 입소자와의 관계 등의 정보를 확인할 수 있다.

중요성과 유용성:

- 비상 연락체계 확보: 응급 상황이나 긴급 연락이 필요한 경우 보호자 정보를 빠르게 조회할 수 있어 즉각적인 소통이 가능하다.
- 입소자 관리 및 의사결정 지원: 의사결정이 필요한 상황(치료 결정, 비용 문제 등)에서 보호자와 신속한 협의가 가능하다.

9. "최근 한달 부작용이 있는 약물을 처방받은 입소자 리스트"

```
SELECT

    DISTINCT Resident.Resident_ID, Resident.Name AS Resident_Name

FROM

    Resident, HealthRecord, Medication

WHERE

    Resident.Resident_ID = HealthRecord.Resident_ID

    AND HealthRecord.Record_ID = Medication.Record_ID

    AND Medication.Side_Effect_Status = 'Y'

    AND Medication.Prescription_Date >= ADD_MONTHS(SYSDATE, -1);
```

대답하는 내용: 최근 한 달 내에 부작용(Side Effect Status = True 등)이 보고된 약물을 처방받은 모든 입소자 목록을 반환한다.

중요성과 유용성:

- 약물 안전성 모니터링: 특정 기간 동안 부작용 약물 처방 이력이 있는 입소자들을 파악하여 약물 관리 및 처방 정책을 재검토할 수 있다.
- 입소자 보호 및 치료 개선: 해당 입소자들에게 대체 약물 처방, 투약 방법 변경, 추가 관찰 등의 대응 방안을 수립할 수 있다.

10. "김철수 입소자의 건강 기록 내역

SELECT

HealthRecord.Record_ID, HealthRecord.Blood_Pressure,
HealthRecord.Temperature, HealthRecord.Blood_Sugar,
HealthRecord.Record_Date

FROM

HealthRecord, Resident

WHERE

HealthRecord.Resident_ID = Resident.Resident_ID AND Resident.Name =
'김철수';

대답하는 내용: 특정 입소자(Resident_ID = '000')의 과거 Health Record 데이터(혈압, 체온, 혈당, 측정 날짜, 관련 약물 정보 등)를 조회한다.

중요성과 유용성:

- 건강 추세 파악: 입소자의 지난 건강 상태 변화를 시간축으로 파악하여 적절한 치료나 관리 전략을 세울 수 있다.
- 맞춤형 의료 서비스 제공: 건강 상태 변화를 바탕으로 맞춤형 간호나 정기 진료 계획 수립, 필요 시 보호자 및 의사와 협의 가능하다.

11. "입소자를 배정받지 못한 간호사 리스트와 입사일"

SELECT

Staff.Staff_ID, Staff.Name, Staff.Hired_Date

FROM

Staff

WHERE

Staff.Role = 'Nurse' AND Staff.Staff_ID NOT IN (SELECT Staff_ID FROM Manage);

대답하는 내용: 현재까지 어떤 Nurse(간호사)들이 아직 맡은 입소자(Managed 데이터가 없는 경우)가 없으며, 그 직원의 입사일(Hired_Date)이 언제인지 조회한다.

중요성과 유용성:

- **인력 관리 및 업무 배정:** 어떤 직원이 아직 업무(입소자 관리)를 할당받지 않았는지 파악함으로써 적절한 인력 재배치나 신입 직원 교육 계획 수립에 도움을 준다.
- **운영 효율성 개선:** 인력 배치를 효율적으로 조정하여 시설 운영의 효율성을 높이고, 인력 자원을 적절히 활용할 수 있다.

12. "약물 복용 시 부작용이 있던 입소자 리스트"

(1) 주사를 맞고 부작용이 있었던 입소자 리스트

```
SELECT

    Resident.resident_ID,

    resident.name as resident_name,

    Medication.MEDICATION_ID,

    Injection.Injection_Method

FROM Medication, healthRecord, resident, injection

where Medication.RECORD_ID = Healthrecord.RECORD_ID

and HealthRecord.resident_ID = Resident.resident_ID

and medication.medicament_ID = injection.injection_medicament_ID

and Medication.SIDE_EFFECT_STATUS = 'Y';
```

(2) 경구 약물을 복용하고 부작용이 있었던 입소자 리스트

```
SELECT

    Resident.resident_ID,

    resident.name as resident_name,

    Medication.MEDICATION_ID,

    Oral_medication.Oral_Form

FROM Medication, healthRecord, resident, Oral_medication

where Medication.RECORD_ID = Healthrecord.RECORD_ID

and HealthRecord.resident_ID = Resident.resident_ID
```

and medication.medication_ID = Oral_Medication.Oral_medication_ID

and Medication.SIDE_EFFECT_STATUS = 'Y';

대답하는 내용: 약물을 복용한 입소자들 중 부작용이 있었던 입소자의 ID, 이름과 어떤 약물(주사 혹은 경구 약물)을 복용했는지 조회한다.

중요성과 유용성:

- 입소자의 약물 제한: 어떤 입소자가 어떤 약물을 복용하여 부작용이 있었는지를 판단하여 특정 입소자가 복용하면 안되는 약물을 제한하고 다른 방법으로 약물을 제공한다.
- 맞춤형 약물 처방: 해당 입소자에게 부작용이 있었던 약물, 투약방법 데이터를 기반으로 부작용이 없는 대체 약물, 투약 방법등을 변경 및 제공하여 부작용에 대한 대처가 가능하다.

과제 보고서

보고서 및 논문 윤리 서약

1. 나는 보고서 및 논문의 내용을 조작하지 않겠습니다.
 2. 나는 다른 사람의 보고서 및 논문의 내용을 내 것처럼 무단으로 복사하지 않겠습니다.
 3. 나는 다른 사람의 보고서 및 논문의 내용을 참고하거나 인용할 시 참고 및 인용 형식을 갖추고 출처를 반드시 밝히겠습니다.
 4. 나는 보고서 및 논문을 대신하여 작성하도록 청탁하지도 청탁받지도 않겠습니다.
- 나는 보고서 및 논문 작성 시 위법 행위를 하지 않고, 명지인으로서 또한 공학인으로서 나의 양심과 명예를 지킬 것을 약속합니다.



보고서명 : DB 설계 및 구현 2 최종 레포트

학 과 : 응용소프트웨어학과

과 목 : DB 설계 및 구현 2

담당교수 : 김일주 교수님

학 번 : 60201921 성 명 : 김준서

학 번 : 60211902 성 명 : 김성엽

학 번 : 60211647 성 명 : 김민욱

학 번 : 60231199 성 명 : 오동이

학 번 : 60200036 성 명 : 임채아