

**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
PROGRAM STUDI S1 ILMU KOMPUTER  
SEMESTER GASAL 2025/2026  
DATABASE (MII212501)**



**LOGICAL SCHEMA & SQL IMPLEMENTATION (GROUP 11)**

**ZHARIF AFZHAN BIN ZHURAINI (25/567441/NPA/19987)  
NUR ADIBAH BINTI MAFILRUDDIN (25/568511/NPA/19988)**

## Clinic Appointment System - Logical Database Schema

### 1. Entities and Attributes

#### Staff

Field	Type	Description
staffID (PK)	INT	Unique staff identifier
username	VARCHAR(50)	Staff username for login
password	VARCHAR(255)	Staff password for login
full_name	VARCHAR(100)	Staff's name
phone_no	VARCHAR(20)	Contact number
email	VARCHAR(100)	Email address

#### Doctor

Field	Type	Description
doctorID (PK)	INT	Unique doctor identifier
full_name	VARCHAR(100)	Doctor's name
specialization	VARCHAR(100)	Field of expertise
phone	VARCHAR(15)	Contact number
email	VARCHAR(100)	Email address
working_days	VARCHAR(50)	Days available (e.g., Mon–Fri)
room_no	VARCHAR(10)	Assigned room number

## Patient

Field	Type	Description
patientID (PK)	INT	Unique patient identifier
full_name	VARCHAR(100)	Patient's name
age	INT	Patient age
gender	VARCHAR(10)	Male/Female
date_of_birth	DATE	Birth date
phone	VARCHAR(15)	Contact number
address	TEXT(255)	Patient's address
emergency_contact	VARCHAR(15)	Emergency contact person

## Appointment

Field	Type	Description
appointmentID (PK)	INT	Unique appointment ID
appointment_number	VARCHAR(20)	Appointment number
reason_for_appointment	TEXT	Reason for the appointment
appointment_time	DATETIME	Time &date of appointment
status	VARCHAR(100)	(Pending, Completed, Cancelled)
patient_come_into_hospital	ENUM('yes', 'no')	Status if the patient come to the appointment or not
doctor_comment	TEXT	Doctor comment
doctor_id (FK)	INT	Linked to Doctor
patient_id (FK)	INT	Linked to Patient

## 2. Relationships

Relationship	Description
Doctor → Appointment	One doctor can handle many appointments
Patient → Appointment	One patient can make many appointments
Staff → Appointment	One staff can create/manage many appointments

## 3. Normalization (up to 3NF)

All tables are normalized up to Third Normal Form (3NF):

- 1NF: No repeating groups; each field holds atomic values.
- 2NF: Every non-key attribute depends entirely on the primary key.
- 3NF: No transitive dependencies between non-key attributes.

## 4. SQL Implementation

```
CREATE DATABASE clinic_db;
USE clinic_db;

-- Staff Table
CREATE TABLE Staff (
    staffID INT AUTO_INCREMENT PRIMARY KEY,
    username VARCHAR(50) NOT NULL UNIQUE,
    password VARCHAR(255) NOT NULL,
    full_name VARCHAR(100),
    phone_no VARCHAR(20),
    email VARCHAR(100)
);

-- Doctor Table
CREATE TABLE Doctor (
    doctorID INT AUTO_INCREMENT PRIMARY KEY,
    full_name VARCHAR(100) NOT NULL,
    specialization VARCHAR(100),
    phone VARCHAR(15),
    email VARCHAR(100),
    working_days VARCHAR(50),
    room_no VARCHAR(10)
);
```

```
-- Patient Table
CREATE TABLE Patient (
  patientID INT AUTO_INCREMENT PRIMARY KEY,
  full_name VARCHAR(100) NOT NULL,
  age INT,
  gender VARCHAR(10),
  date_of_birth DATE,
  phone VARCHAR(15),
  address VARCHAR(255),
  emergency_contact VARCHAR(15)
);

-- Appointment Table
CREATE TABLE Appointment (
  appointmentID INT AUTO_INCREMENT PRIMARY KEY,
  appointment_number VARCHAR(20),
  reason_for_appointment TEXT,
  appointment_time DATETIME,
  status VARCHAR(100),
  patient_come_into_hospital ENUM('yes', 'no'),
  doctor_comment TEXT,
  doctorID INT,
  patientID INT,
  FOREIGN KEY (doctorID) REFERENCES Doctor(doctorID),
  FOREIGN KEY (patientID) REFERENCES Patient(patientID)
);
```