

MAGISTERE Ingénieur-Economiste

MANAGING DATAGING DATABASE : SQL- NOSQL

EXAM PROJECT

GROUPE 8 :

- ADJANOHOUN Brayann
- ADOUSSINGANDE Céline
- COMLAN Cybelle
- KOUKPONOU Sonagnon

2023-2024

Table des matières

Table des matières	1
Introduction	2
Part 1: Model.....	3
1.1 Dictionary	3
1.2 Conceptual Data Model - CDM (the schema/drawing).....	4
1.2.1 Normalization steps, Hypothesis and constraints.....	6
1.3 Transformation of the CDM into the Logical Data Model – LMD.....	7
1.3.1 Associations of type 1,1	7
1.3.2 Association of type 1, N.....	7
1.3.3 Les associations de type N, N	7
1.3.4 Logical Data Model.....	7
1.4 Creation script for MySQL.....	9
1.5 Insertion Script	20
Part 2 : Queries and SQL solutions	35
2.1 Basic Queries.....	35
2.2 WHERE Clause Queries	37
2.3 ORDER BY queries	39
2.4 Queries with Numeric expressions and functions;.....	42
2.5 GROUP BY Queries.....	44
2.6 Multi-Table queries	46
2.7 Nested queries	49
Part3 : MONGODB.....	51
3.1 The 3 .json files containing all your data (1 file per entitiy)	52
3.2 The insert script (for this, you shall use the mongoimport command to import the 3 .json files).....	52
3.3 Python script.....	52
Conclusion.....	54

Introduction

As part of our group project on database construction, we chose to model a relational database on the CICB hospital management system. It includes information related to the patient, medical and non-medical staff. We also tried to model particular entities such as patient feedback. There are also tables on suppliers to meet subcontracting needs for the purchase of medical equipment and drug supplies.

Part 1: Model

1.1 Dictionary

Attribute	Type	Description
Salary_ADMINISTRATIVE STAFF	INT	Administrative staff salaries
Position_ADMINISTRATIVE STAFF	VARCHAR	Position held by administrative staff.
Name_ADMINISTRATIVE STAFF	VARCHAR	Name of administrative staff member.
ID_AD_STAFF_ADMINISTRATIVE STAFF	VARCHAR	The ID of the administrative staff member.
Gender_ADMINISTRATIVE STAFF	VARCHAR	The gender of the administrative staff member.
Date_APPOINTEMENT	DATE	Appointment date.
ID_BED_BEDS	INT	The ID of Bed
Availability_BEDS	VARCHAR	Available beds
Time_BILLING	TIME	Time of billing
Payment_Method_BILLING	VARCHAR	Invoice payment method.
ID_INVOICE_BILLING	INT	The ID of Invoice billing
Date_BILLING	DATE	The Date of Billing
Date_CONSULT	DATE	The Date of Consultation
Mortality_rate_DISEASE	DECIMAL	Disease mortality rate
DISEASE_NAME_DISEASE	VARCHAR	Disease Name
Status_DOCTORS	VARCHAR	The doctor's status
Salary_DOCTORS	INT	The doctor's salary
Rotation_DOCTORS	VARCHAR	The doctor's Rotation
Name_DOCTORS	VARCHAR	The doctor's Name
ID_Doctor_DOCTORS	INT	The ID of Doctors
Gender_DOCTORS	CHAR	The gender of Doctor
Price_DRUGS	INT	Drug pricing
Name_DRUGS	VARCHAR	Drug Name
ID_DRUG_DRUGS	INT	The ID of Drugs
Reason_EMERGENCY MANAGEMENT	VARCHAR	Reason for emergency hospital admission
ID_EMERGENCY_EMERGENCY MANAGEMENT	INT	The ID of Emergency
Date_of_arrival_EMERGENCY MANAGEMENT	DATE	The patient's date of arrival at the emergency department
Specialty_LABORATORY	VARCHAR	The specialty of Laboratory
ID_LABORATORY_LABORATORY	INT	The ID of Laboratory
Type_of_maintenance__MAINTENANCE	VARCHAR	The Type_of_maintenance
ID_MAINTENANCE_MAINTENANCE	INT	ID of_maintenance
Date_of_Maintenance_MAINTENANCE	VARCHAR	Date of maintenance
Status_MEDICAL EQUIPMENT	VARCHAR	Condition of medical equipment
Name_MEDICAL EQUIPMENT	VARCHAR	Name of medical equipment
ID_EQUIPMENT_MEDICAL EQUIPMENT	INT	ID of medical equipment

Medical_History_MEDICAL FILE	TEXT	The patient's medical history.
ID_FILE_MEDICAL FILE	INT	The ID of medical history.
File_Status_MEDICAL FILE	VARCHAR	File_Status_MEDICAL FILE
Diagnosis_MEDICAL FILE	VARCHAR	Diagnosis of MEDICAL FILE
Status_NURSES	VARCHAR	Status of nurses
Salary_NURSES	INT	Salary of nurses
Rotation_NURSES	VARCHAR	Nurses rotation
Name_Nurses	VARCHAR	Name of Nurses
ID_NURSES_NURSES	INT	ID of Nurses
Gender_NURSES	VARCHAR	Gender of Nurses
Sex_PATIENTS	CHAR	Gender of Patients
Name_PATIENTS	VARCHAR	Name of Patients
ID_PATIENTS_PATIENTS	INT	ID of Patients
Age_PATIENTS	INT	Patient's age
Address_PATIENTS	VARCHAR	Address of Patients
Name_PHARMACY	VARCHAR	Name of Pharmacy
ID_PHARMACY_PHARMACY	INT	ID of Pharmacy
Status_ROOMS	VARCHAR	Status of Rooms
Number_of_beds_ROOMS	INT	Number of beds in ROOMS
ID_ROOMS_ROOMS	INT	ID of Rooms
Type_SERVICES	VARCHAR	Type of service
ID_SERVICE_SERVICES	INT	ID of SERVICE
Name_SUPPLIERS	VARCHAR	Name of Suppliers
Medicine_SUPPLIERS	VARCHAR	Medicine_SUPPLIERS
Medical_equipment_SUPPLIERS	VARCHAR	Medical equipment supplied
ID_SUPPLIER_SUPPLIERS	INT	ID of Suppliers
Type_of_Vacation_VACATION MANAGEMENT	VARCHAR	Type of vacations.
Start_day_VACATION MANAGEMENT	DATE	Dates of begun vacations.
ID_VACATION_VACATION MANAGEMENT	INT	The ID of vacations.
End_Day_VACATION MANAGEMENT	DATE	Start and end dates of staff vacations.

1.2 Conceptual Data Model - CDM (the schema/drawing).

Here we deal only with entity-association relationships. The foreign keys are specified in the tables at the MLD level.



1.2.1 Normalization steps, Hypothesis and constraints

1.2.1.1 Normalization steps, Hypothesis

At the Patient Level:

At the BCCI hospital, a patient can have at least one disease and at most several diseases. The latter may not be hospitalized at all or hospitalized several times. Some patients are admitted to a single emergency. The emergency here refers to the case faced by the doctors. Moreover, the patient must have a single and unique medical record. They can make several medical appointments with the administrative service. This service is not only responsible for organizing the hospital employees but also for managing leaves and absences.

At the Doctors Level:

Each doctor must belong to a single service and can treat at least one disease. Doctors may not be assisted by nurses. Each doctor performs at least one analysis in the laboratory. We assume that the CICB hospital contains several laboratories distributed by services. Each laboratory is equipped with one or more pieces of equipment. These pieces of equipment are delivered by one or more suppliers. The said suppliers are also responsible for providing medications.

We also assume that the hospital has several pharmacies containing identifiable medications with an expiration date. A supplier may not provide medications or may provide several medications. We assume that each pharmacy contains the medications delivered by the suppliers.

1.2.1.2 Constraints we couldn't consider into our model

This model is not perfect. Indeed, we could have considered several other elements in the description of our hospital. These include: the adverse event management system, which is used to manage medical incidents. There are also research and teaching centers. Many hospitals have research and teaching centers in order to train and properly educate the next generation of healthcare professionals. We could also add communication infrastructures to facilitate communication between the various medical services.”

1.3 Transformation of the CDM into the Logical Data Model – LMD

To transform the MDC into MDL, we group the entity-association relationships by type of associations:

1.3.1 Associations of type 1,1

Merging tables and removing the unique ID. We have two 1,1 relationships in our model:

1. Between the entity **PATIENTS** and **MEDICAL_FILE**
2. Between the entity **MEDICAL_EQUIPMENT** and **MAINTENANCE**

1.3.2 Association of type 1, N

Addition in table 1 in entity 1 of a foreign key which is the primary key of entity N. We have 4 types of 1, N relationships in our model:

1. **DOCTORS AND SERVICES**
2. **PATIENTS and ROOMS**
3. **PATIENTS and EMERGENCY**
4. **ROOMS and BEDS**

1.3.3 Les associations de type N, N

Addition of a new table containing the two primary keys + the attributes of the association. The rest of the associations are N, N type associations. There are 16 of them.

1.3.4 Logical Data Model

The different tables of our model are presented as follows:

1. **SERVICES** (**ID_SERVICE**, Type)
2. **DOCTORS** (**ID_Doctor**, Name, Gender, Salary, Rotation, Status, **#ID_SERVICE**)
3. **DISEASE** (**DISEASE_NAME**, Mortality rate,)
4. **TREAT** (**#ID DOCTOR, DISEASE_NAME**)

5. **PATIENTS** (ID PATIENTS, Name, Sex, Age, Address, Medical History, File Status, Diagnosis, #ID_ROOMS, #ID_EMERGENCY)
6. **CONSULT** (#ID DOCTOR, ID PATIENTS, Date_Conult)
7. **LABORATORY** (ID LABORATORY, Specialty)
8. **ANALYSE** (#ID Doctor, ID LABORATORY)
9. **NURSES** (ID NURSES, Name, Gender, Status, Rotation, Salary)
10. **ASSIST** (#ID DOCTOR, ID NURSES)
11. **SUFFER** (#ID PATIENTS, DISEASE NAME)
12. **FOLLOW** (#ID PATIENTS, ID NURSES)
13. **MEDICAL EQUIPMENT** (ID EQUIPMENT, Name, Status, type of Maintenance, Date of Maintenance, price of Maintenance)
14. **EQUIP** (#ID LABORATORY, ID EQUIPMENT)
15. **ROOMS** (ID ROOMS, Number of beds, Status, #ID_BED)
16. **EMERGENCY MANAGEMENT** (ID EMERGENCY, Reason, Date of Arrival)
17. **BILLING** (ID INVOICE, Payment Method, Date, Time)
18. **PAY** (#ID PATIENTS, ID INVOICE)
19. **ADMINISTRATIVE STAFF** (ID AD STAFF, Name, Gender, Position, Salary)
20. **APPOINTEMENT** (#ID PATIENTS, ID AD STAFF, Date_Appointment)

- 21. SUPPLIERS (ID_SUPPLIER, Name, Medicine)
- 22. DELIVRER (#ID_SUPPLIER, ID_EQUIPMENT)
- 23. VACATION MANAGEMENT (ID_VACATION, Type of Vacation, Start day, End Day)
- 24. MANAGE (#ID_AD STAFF, ID_VACATION)
- 25. DRUGS (ID_DRUGS, Name, Price)
- 26. PROVIDES (#ID_SUPPLIER, ID_DRUGS)
- 27. PHARMACY (#ID_PHARMACY, Name of Pharmacy)
- 28. CONTAINED (#ID_PHARMACY, ID_DRUGS)
- 29. BEDS (ID_BEDS, Availability, #ID_ROOMS)

1.4 Creation script for MySQL

DROP DATABASE IF EXISTS Hospital_CICB;

CREATE DATABASE Hospital_CICB;

use Hospital_CICB;

DROP TABLE IF EXISTS SERVICES;

CREATE TABLE SERVICES (

ID_SERVICE INT(10) NOT NULL PRIMARY KEY,

Type VARCHAR(100));

DROP TABLE IF EXISTS DOCTORS;

CREATE TABLE DOCTORS (

ID_Doctor INT NOT NULL PRIMARY KEY,

Name VARCHAR(100),

Gender CHAR(1),

Salary INT,

Rotation VARCHAR(100),

Status VARCHAR(100),

ID_SERVICE INT,

**FOREIGN KEY (ID_SERVICE) REFERENCES SERVICES(ID_SERVICE) ON
DELETE CASCADE**

);

DROP TABLE IF EXISTS DISEASE;

CREATE TABLE DISEASE (

DISEASE_NAME VARCHAR(100) PRIMARY KEY,

Mortality_rate DECIMAL(5, 2)

);

DROP TABLE IF EXISTS TREAT;

CREATE TABLE TREAT (

ID_DOCTOR INT NOT NULL,

DISEASE_NAME VARCHAR(100),

PRIMARY KEY (ID_DOCTOR, DISEASE_NAME),

FOREIGN KEY (ID_DOCTOR) REFERENCES DOCTORS(ID_Doctor)ON DELETE CASCADE,

FOREIGN KEY (DISEASE_NAME) REFERENCES DISEASE(DISEASE_NAME)ON DELETE CASCADE

);

DROP TABLE IF EXISTS LABORATORY;

CREATE TABLE LABORATORY (

ID_LABORATORY INT NOT NULL PRIMARY KEY,

Specialty VARCHAR(100)

);

DROP TABLE IF EXISTS ANALYSE;

CREATE TABLE ANALYSE (

ID_Doctor INT NOT NULL,

ID_LABORATORY INT NOT NULL,

PRIMARY KEY (ID_Doctor, ID_LABORATORY),

FOREIGN KEY (ID_Doctor) REFERENCES DOCTORS(ID_Doctor)ON DELETE CASCADE,

FOREIGN KEY (ID_LABORATORY) REFERENCES LABORATORY(ID_LABORATORY)ON DELETE CASCADE

);

DROP TABLE IF EXISTS NURSES;

CREATE TABLE NURSES (

ID_NURSES INT NOT NULL PRIMARY KEY,

```
Name VARCHAR(100),  
Gender CHAR(1),  
Status VARCHAR(100),  
Rotation VARCHAR(100),  
Salary INT  
);
```

```
DROP TABLE IF EXISTS ASSIST;
```

```
CREATE TABLE ASSIST (  
    ID_DOCTOR INT NOT NULL,  
    ID_NURSES INT NOT NULL,  
    PRIMARY KEY (ID_DOCTOR, ID_NURSES),  
    FOREIGN KEY (ID_DOCTOR) REFERENCES DOCTORS(ID_Doctor),  
    FOREIGN KEY (ID_NURSES) REFERENCES NURSES(ID_NURSES)  
);
```

```
DROP TABLE IF EXISTS MEDICAL_EQUIPMENT;
```

```
CREATE TABLE MEDICAL_EQUIPMENT (  
    ID_EQUIPMENT INT NOT NULL PRIMARY KEY,  
    Name VARCHAR(100),  
    Status VARCHAR(100),  
    Type_of_Maintenance VARCHAR(100),  
    Date_of_Maintenance DATE,  
    Price_of_Maintenance INT  
);
```

```

DROP TABLE IF EXISTS EQUIP;

CREATE TABLE EQUIP (
    ID_LABORATORY INT NOT NULL,
    ID_EQUIPMENT INT NOT NULL,
    PRIMARY KEY (ID_LABORATORY, ID_EQUIPMENT),
    FOREIGN KEY (ID_LABORATORY) REFERENCES
LABORATORY(ID_LABORATORY) ON DELETE CASCADE,
    FOREIGN KEY (ID_EQUIPMENT) REFERENCES
MEDICAL_EQUIPMENT(ID_EQUIPMENT) ON DELETE CASCADE
);

DROP TABLE IF EXISTS EMERGENCY_MANAGEMENT ;

CREATE TABLE EMERGENCY_MANAGEMENT (
    ID_EMERGENCY INT NOT NULL PRIMARY KEY,
    Reason VARCHAR(100),
    Date_of_Arrival DATE
);

DROP TABLE IF EXISTS BILLING ;

CREATE TABLE BILLING (
    ID_INVOICE INT NOT NULL PRIMARY KEY,
    Payment_Method VARCHAR(100),
    Date DATE,
    Time TIME
);

DROP TABLE IF EXISTS ADMINISTRATIVE_STAFF ;

```

```
CREATE TABLE ADMINISTRATIVE_STAFF (  
    ID_AD_STAFF INT NOT NULL PRIMARY KEY,  
    Name VARCHAR(100),  
    Gender CHAR(1),  
    Position VARCHAR(100),  
    Salary INT  
);
```

```
DROP TABLE IF EXISTS SUPPLIERS ;
```

```
CREATE TABLE SUPPLIERS (  
    ID_SUPPLIER INT NOT NULL PRIMARY KEY,  
    Name VARCHAR(100),  
    Medicine VARCHAR(100)  
);
```

```
DROP TABLE IF EXISTS DELIVRER ;
```

```
CREATE TABLE DELIVRER (  
    ID_SUPPLIER INT NOT NULL,  
    ID_EQUIPMENT INT NOT NULL,  
    PRIMARY KEY (ID_SUPPLIER, ID_EQUIPMENT),  
    FOREIGN KEY (ID_SUPPLIER) REFERENCES SUPPLIERS(ID_SUPPLIER) ON  
DELETE CASCADE,  
    FOREIGN KEY (ID_EQUIPMENT) REFERENCES  
MEDICAL_EQUIPMENT(ID_EQUIPMENT) ON DELETE CASCADE  
);
```

DROP TABLE IF EXISTS VACATION_MANAGEMENT ;

CREATE TABLE VACATION_MANAGEMENT (
ID_VACATION INT NOT NULL PRIMARY KEY,
Type_of_Vacation VARCHAR(100),
Start_day DATE,
End_Day DATE
);

DROP TABLE IF EXISTS MANAGE ;

CREATE TABLE MANAGE (
ID_AD_STAFF INT NOT NULL,
ID_VACATION INT NOT NULL,
PRIMARY KEY (ID_AD_STAFF, ID_VACATION),
FOREIGN KEY (ID_AD_STAFF) REFERENCES
ADMINISTRATIVE_STAFF(ID_AD_STAFF) ON DELETE CASCADE,
FOREIGN KEY (ID_VACATION) REFERENCES
VACATION_MANAGEMENT(ID_VACATION) ON DELETE CASCADE
);

DROP TABLE IF EXISTS DRUGS ;

CREATE TABLE DRUGS (
ID_DRUGS INT NOT NULL PRIMARY KEY,
Name VARCHAR(100),
Expiry_date DATE,
Price INT

);

DROP TABLE IF EXISTS PROVIDES ;

```
CREATE TABLE PROVIDES (  
    ID_SUPPLIER INT NOT NULL,  
    ID_DRUGS INT NOT NULL,  
    PRIMARY KEY (ID_SUPPLIER, ID_DRUGS),  
    FOREIGN KEY (ID_SUPPLIER) REFERENCES SUPPLIERS(ID_SUPPLIER) ON  
DELETE CASCADE,  
    FOREIGN KEY (ID_DRUGS) REFERENCES DRUGS(ID_DRUGS) ON DELETE  
CASCADE  
);
```

DROP TABLE IF EXISTS PHARMACY ;

```
CREATE TABLE PHARMACY(  
    ID_PHARMACY INT NOT NULL PRIMARY KEY,  
    Name_of_Pharmacy VARCHAR(100)  
);
```

DROP TABLE IF EXISTS CONTAINED ;

```
CREATE TABLE CONTAINED (  
    ID_PHARMACY INT NOT NULL,  
    ID_DRUGS INT NOT NULL,  
    PRIMARY KEY (ID_PHARMACY, ID_DRUGS),
```

**FOREIGN KEY (ID_PHARMACY) REFERENCES
PHARMACY(ID_PHARMACY) ON DELETE CASCADE,**

**FOREIGN KEY (ID_DRUGS) REFERENCES DRUGS(ID_DRUGS) ON DELETE
CASCADE**

);

DROP TABLE IF EXISTS ROOMS ;

CREATE TABLE ROOMS (

ID_ROOMS INT NOT NULL PRIMARY KEY,

Number_of_beds INT,

Status VARCHAR(255));

DROP TABLE IF EXISTS BEDS ;

CREATE TABLE BEDS (

ID_BEDS INT NOT NULL PRIMARY KEY,

Availability VARCHAR(255),

ID_ROOMS INT,

**FOREIGN KEY (ID_ROOMS) REFERENCES ROOMS(ID_ROOMS) ON DELETE
CASCADE);**

DROP TABLE IF EXISTS PATIENTS ;

CREATE TABLE PATIENTS (

ID_PATIENTS INT NOT NULL PRIMARY KEY,

Name VARCHAR(255),

Sex CHAR(1),

Age **INT**,
 Address **VARCHAR(255)**,
 Medical_History **TEXT**,
 File_Status **VARCHAR(255)**,
 Diagnosis **VARCHAR(255)**,
 ID_ROOMS **INT**,
 ID_EMERGENCY **INT**,
FOREIGN KEY (ID_ROOMS) REFERENCES ROOMS(ID_ROOMS) ON DELETE CASCADE,
FOREIGN KEY (ID_EMERGENCY) REFERENCES EMERGENCY_MANAGEMENT(ID_EMERGENCY) ON DELETE CASCADE);

DROP TABLE IF EXISTS CONSULT ;

CREATE TABLE CONSULT (
 ID_DOCTOR **INT NOT NULL**,
 ID_PATIENTS **INT NOT NULL**,
 Date_consult **DATE**,
PRIMARY KEY (ID_DOCTOR, ID_PATIENTS),
FOREIGN KEY (ID_DOCTOR) REFERENCES DOCTORS(ID_DOCTOR) ON DELETE CASCADE,
FOREIGN KEY (ID_PATIENTS) REFERENCES PATIENTS(ID_PATIENTS) ON DELETE CASCADE);

DROP TABLE IF EXISTS SUFFER ;

CREATE TABLE SUFFER (
 ID_PATIENTS **INT NOT NULL**,

```
DISEASE_NAME VARCHAR(100),  
PRIMARY KEY (ID_PATIENTS, DISEASE_NAME),  
FOREIGN KEY (ID_PATIENTS) REFERENCES PATIENTS(ID_PATIENTS),  
FOREIGN KEY (DISEASE_NAME) REFERENCES DISEASE(DISEASE_NAME));
```

```
DROP TABLE IF EXISTS FOLLOW ;
```

```
CREATE TABLE FOLLOW (  
    ID_PATIENTS INT NOT NULL,  
    ID_NURSES INT NOT NULL,  
    PRIMARY KEY (ID_PATIENTS, ID_NURSES),  
    FOREIGN KEY (ID_PATIENTS) REFERENCES PATIENTS(ID_PATIENTS) ON  
DELETE CASCADE,  
    FOREIGN KEY (ID_NURSES) REFERENCES NURSES(ID_NURSES) ON  
DELETE CASCADE);
```

```
DROP TABLE IF EXISTS PAY ;
```

```
CREATE TABLE PAY (  
    ID_PATIENTS INT NOT NULL,  
    ID_INVOICE INT NOT NULL,  
    PRIMARY KEY (ID_PATIENTS, ID_INVOICE),  
    FOREIGN KEY (ID_PATIENTS) REFERENCES PATIENTS(ID_PATIENTS),  
    FOREIGN KEY (ID_INVOICE) REFERENCES BILLING(ID_INVOICE));
```

```
DROP TABLE IF EXISTS APPOINTEMENT;
```

```
CREATE TABLE APPOINTEMENT (
```

```

ID_PATIENTS INT NOT NULL,

ID_AD_STAFF INT NOT NULL,

Date_Appointment DATE,

PRIMARY KEY (ID_PATIENTS, ID_AD_STAFF),

FOREIGN KEY (ID_PATIENTS) REFERENCES PATIENTS(ID_PATIENTS),

FOREIGN KEY (ID_AD_STAFF) REFERENCES
ADMINISTRATIVE_STAFF(ID_AD_STAFF));

```

1.5 Insertion Script

```

USE Hospital_CICB;

INSERT INTO SERVICES (ID_SERVICE, Type) VALUES

(21, 'Emergency department'),

(22, 'General medicine'),

(23, 'Surgery'),

(24, 'Pediatrics'),

(25, 'Obstetrics and gynecology'),

(26, 'Cardiology'),

(27, 'Neurology'),

(28, 'Radiology'),

(29, 'Rehabilitation service'),

(30, 'Psychiatry');

```

```

INSERT INTO DOCTORS (ID_Doctor, Name, Gender, Salary, Rotation, Status,
ID_SERVICE) VALUES

(1111, 'Sarah Dupuis', 'F', 9000, 'Day', 'Contract', 21),

```

(1112, 'Maxime Leclerc', 'M', 6000, 'Night', 'Temporary', 22),
 (1113, 'Zoé Bernard', 'F', 8000, 'Night', 'Temporary', 23),
 (1114, 'Nathan Roy', 'M', 4000, 'Night', 'Interim', 24),
 (1115, 'Jade Lavoie', 'F', 3000, 'Day', 'Interim', 25),
 (1116, 'Alexandre Gagnon', 'M', 6000, 'Day', 'Consultant', 26),
 (1117, 'Maëlle Caron', 'F', 6000, 'Night', 'Temporary', 27),
 (1118, 'Gabriel Tremblay', 'M', 3000, 'Day', 'Interim', 28),
 (1119, 'Charlotte Desjardins', 'F', 3000, 'Day', 'Consultant', 29),
 (1110, 'Mathis Fortin', 'M', 3000, 'Day', 'Contract', 30);

INSERT INTO DISEASE (DISEASE_NAME, Mortality_rate) VALUES

('Cancer', 30.5),
 ('Diabetes', 5.6),
 ('HIV', 2.3),
 ('Hepatitis', 5.3),
 ('Mental illnesses', 10.5),
 ('Chronic diseases', 5.5),
 ('Obesity', 4.8),
 ('Allergies', 1.9),
 ('Sleep disorders', 1.7),
 ('Respiratory disease', 10.5);

INSERT INTO LABORATORY (ID_LABORATORY, Specialty) VALUES

(81, 'Clinical Laboratory'),
 (82, 'Microbiology Laboratory'),

(83, 'Anatomic Pathology Laboratory'),

(84, 'Medical Genetics Laboratory'),

(85, 'Cardiology Laboratory'),

(86, 'Toxicology Laboratory'),

(87, 'Hematology Laboratory'),

(88, 'Quality Control'),

(89, 'Genetic Testing'),

(90, 'Radiographie');

INSERT INTO ANALYSE (ID_Doctor, ID_LABORATORY) VALUES

(1111, 81),

(1112, 82),

(1113, 83),

(1114, 84),

(1115, 85),

(1116, 86),

(1117, 87),

(1118, 88),

(1119, 89),

(1110, 90);

**INSERT INTO NURSES (ID_NURSES, Name, Gender, Status, Rotation, Salary)
VALUES**

(11, 'Marie Dubois', 'F', 'Contractual', 'Day', 3000),

(12, 'Julien Lefevre', 'M', 'Temporary', 'Night', 5000),

(13, 'Anaïs Martin', 'F', 'Temporary', 'Night', 6000),

(14, 'Lucas Dupont', 'M', 'Interim', 'Night', 5000),
(15, 'Sarah Lambert', 'F', 'Interim', 'Day', 7000),
(16, 'Thomas Laurent', 'M', 'Consultant', 'Day', 8000),
(17, 'Camille Moreau', 'M', 'Temporary', 'Night', 5000),
(18, 'Pierre Girard', 'M', 'Interim', 'Day', 5000),
(19, 'Émilie Rousseau', 'F', 'Consultant', 'Day', 7000),
(20, 'Antoine Leroy', 'M', 'Contractual', 'Day', 6000);

INSERT INTO ASSIST (ID_DOCTOR, ID_NURSES) VALUES

(1111, 11),
(1112, 12),
(1113, 13),
(1114, 14),
(1115, 15),
(1116, 16),
(1117, 17),
(1118, 18),
(1119, 19),
(1110, 20);

**INSERT INTO MEDICAL_EQUIPMENT (ID_EQUIPMENT, Name, Status,
Type_of_Maintenance, Date_of_Maintenance, Price_of_Maintenance) VALUES**

(71, 'X-ray machine', 'functional', 'Maintenance préventive', '2022-03-15', 500),
(72, 'MRI scanner (Magnetic Resonance Imaging)', 'unfunctional', 'Adaptive
Maintenance', '2022-03-17', 200),
(73, 'Ultrasound machine', 'functional', 'Adaptive Maintenance', '2022-01-05', 400),

(74, 'ECG machine (Electrocardiogram)', 'functional', 'Adaptive Maintenance', '2022-03-15', 500),

(75, 'Blood pressure monitor', 'unfunctional', 'Corrective Maintenance', '2022-05-10', 200),

(76, 'Ventilator', 'unfunctional', 'Maintenance préventive', '2022-03-06', 100),

(77, 'Defibrillator', 'functional', 'Corrective Maintenance', '2022-03-15', 100),

(78, 'Infusion pump', 'functional', 'Corrective Maintenance', '2022-03-06', 200),

(79, 'Surgical microscope', 'functional', 'Maintenance préventive', '2022-04-02', 300),

(80, 'Electroencephalogram (EEG) machine', 'functional', 'Adaptive Maintenance', '2022-04-02', 500);

INSERT INTO EQUIP (ID_LABORATORY, ID_EQUIPMENT) VALUES

(81, 71),

(82, 72),

(83, 73),

(84, 74),

(85, 75),

(86, 76),

(87, 77),

(88, 78),

(89, 79),

(90, 80);

INSERT INTO EMERGENCY_MANAGEMENT (ID_EMERGENCY, Reason, Date_of_Arrival) VALUES

(101, 'Fever', '2022-01-01'),

(102, 'Headache', '2022-06-17'),

(103, 'Dizziness', '2022-01-01'),

(104, 'Diarrhea', '2022-07-15'),
(105, 'Headache', '2022-07-10'),
(106, 'Fatigue', '2022-01-01'),
(107, 'Dizziness', '2022-03-06'),
(108, 'Headache', '2022-07-10'),
(109, 'Fever', '2022-06-17'),
(200, 'Fatigue', '2022-04-02');

INSERT INTO BILLING (ID_INVOICE, Payment_Method, Date, Time) VALUES

(201, 'Credit Card', '2022-02-15', '03:00:52'),
(202, 'Cash', '2022-03-17', '01:06:35'),
(203, 'Credit Card', '2022-01-05', '23:12:18'),
(204, 'Cash', '2022-04-25', '21:18:01'),
(205, 'Cash', '2022-05-10', '19:23:44'),
(206, 'Credit Card', '2022-03-06', '17:29:27'),
(207, 'Cash', '2022-02-18', '15:35:10'),
(208, 'Credit Card', '2022-01-20', '13:40:53'),
(209, 'Cash', '2022-05-28', '11:46:36'),
(300, 'Credit Card', '2022-04-02', '09:52:19');

INSERT INTO administrative_staff (ID_AD_STAFF, Name, Gender, Position, Salary) VALUES

(121, 'Elodie Morel', 'F', 'Hospital Director', 9000),
(122, 'Victor Lefebvre', 'M', 'Director of Human Resources', 6000),
(123, 'Manon Dupuis', 'M', 'Director of Finance', 8000),

(124, 'Théola Girard', 'F', 'Medical Secretary', 3000),
(125, 'Camille Assani', 'M', 'Admissions Officer', 3000),
(126, 'Lucas Lambert', 'M', 'Billing Manager', 2500),
(127, 'Léa Martin', 'F', 'Medical Secretary', 2500),
(128, 'Hugo Koffi', 'M', 'Logistics Manager', 3000),
(129, 'Chloé Bernard', 'F', 'Head of Risk Management', 3000),
(130, 'Nina Leroy', 'F', 'Public Relations Manager', 3000);

INSERT INTO SUPPLIERS (ID_SUPPLIER, Name, Medicine) VALUES

(2000, 'MedEquip Solutions', 'Paracétamol'),
(2011, 'Healthcare Supply Co.', 'Ibuprofène'),
(2022, 'LifeLine Medical Devices', 'Diazépam'),
(2033, 'ProMed Instruments', 'Amoxicilline'),
(2044, 'SafeGuard MedTech', 'Doliprane'),
(2055, 'VitalCare Supplies', 'Salbutamol'),
(2066, 'Prime Medical Gear', 'Metformine'),
(2077, 'FirstAid Technologies', 'Salbutamol'),
(2088, 'Reliable MedDevices', 'Dexaméthasone'),
(2099, 'TrustMed Equipment', 'Lorazépam');

INSERT INTO DELIVRER (ID_SUPPLIER, ID_EQUIPMENT) VALUES

(2000, 71),
(2011, 72),
(2022, 73),

(2033, 74),
(2044, 75),
(2055, 76),
(2066, 77),
(2077, 78),
(2088, 79),
(2099, 80);

**INSERT INTO VACATION_MANAGEMENT (ID_VACATION, Type_of_Vacation,
Start_day, End_Day) VALUES**

(501, 'Sick Leave', '2022-05-02', '2022-07-05'),
(502, 'Wellness Retreat', '2022-05-03', '2022-06-16'),
(503, 'Maternity Leave', '2022-05-04', '2022-07-06'),
(504, 'Maternity Leave', '2022-05-05', '2022-08-12'),
(505, 'Wellness Retreat', '2022-05-06', '2022-06-06'),
(506, 'Maternity Leave', '2022-05-07', '2022-10-12'),
(507, 'Sick Leave', '2022-05-08', '2022-11-12'),
(508, 'Wellness Retreat', '2022-09-09', '2022-12-08'),
(509, 'Wellness Retreat', '2022-05-10', '2022-12-13'),
(600, 'Sick Leave', '2022-05-11', '2022-09-14');

INSERT INTO MANAGE (ID_AD_STAFF, ID_VACATION) VALUES

(121, 501),
(122, 502),
(123, 503),

(124, 504),
(125, 505),
(126, 506),
(127, 507),
(128, 508),
(129, 509),
(130, 600);

INSERT INTO DRUGS (ID_DRUGS, Name, Expiry_date, Price) VALUES

(301, 'Paracetamol', '2024-09-30', 2),
(302, 'Ibuprofen', '2023-12-15', 6),
(303, 'Diazepam', '2025-05-20', 6),
(304, 'Amoxicillin', '2023-08-10', 3),
(305, 'Doliprane', '2024-11-28', 3),
(306, 'Salbutamol', '2025-03-12', 10),
(307, 'Metformin', '2024-07-05', 15),
(308, 'Dexamethasone', '2023-10-18', 25),
(309, 'Lorazepam', '2025-01-25', 30),
(400, 'Vitamin', '2024-04-08', 20);

INSERT INTO PROVIDES (ID_SUPPLIER, ID_DRUGS) VALUES

(2000, 301),
(2011, 302),
(2022, 303),

(2033, 304),

(2044, 305),

(2055, 306),

(2066, 307),

(2077, 308),

(2088, 309),

(2099, 400);

INSERT INTO PHARMACY (ID_PHARMACY, Name_of_Pharmacy) VALUES

(601, 'Health Plus Pharmacy'),

(602, 'Wellness Pharmacy'),

(603, 'Care Mart Pharmacy'),

(604, 'Cure Corner Pharmacy'),

(605, 'Aid Aid Pharmacy'),

(606, 'MediShop Pharmacy'),

(607, 'Quick Relief Pharmacy'),

(608, 'LifeLine Pharmacy'),

(609, 'Easy Meds Pharmacy'),

(700, 'PillBox Pharmacy');

INSERT INTO CONTAINED (ID_PHARMACY, ID_DRUGS) VALUES

(601, 301),

(602, 302),

(603, 303),

(604, 304),

(605, 305),
(606, 306),
(607, 307),
(608, 308),
(609, 309),
(700, 400);

INSERT INTO rooms (ID_ROOMS, Number_of_beds, Status) VALUES

(31, 2, 'Busy'),
(32, 3, 'Free'),
(33, 1, 'Busy'),
(34, 2, 'Free'),
(35, 1, 'Busy'),
(36, 1, 'Free'),
(37, 2, 'Free'),
(38, 1, 'Busy'),
(39, 2, 'Free'),
(40, 3, 'Free');

INSERT INTO BEDS (ID_BEDS, Availability, ID_ROOMS) VALUES

(202112023, 'Available', 31),
(202112026, 'unavailable', 32),
(202112029, 'Available', 33),
(202112032, 'unavailable', 34),

(202112035, 'unavailable', 35),
(202112038, 'Available', 36),
(202112041, 'Available', 37),
(202112044, 'Available', 38),
(202112047, 'Available', 39),
(202112050, 'unavailable', 40);

INSERT INTO PATIENTS (ID_PATIENTS, Name, Age, Sex, Address, Medical_History, File_Status, Diagnosis, ID_ROOMS, ID_EMERGENCY) **VALUES**

(1, 'Sophie Martin', 19, 'F', '123 Rue du Soleil Levant, Ville Lumière, État de l'Aurore', 'Yes', 'Open', 'Hypertension', 31, 101),
(2, 'Lucas Dupont', 26, 'M', '456 Avenue des Cyprès, Quartier Tranquille, Ville Paisible', 'Yes', 'Close', 'Diabetes', 32, 102),
(3, 'Emma Leroy', 33, 'F', '789 Chemin des Marguerites, Quartier Fleuri, Ville Jolie', 'No', 'Close', 'Asthma', 33, 103),
(4, 'Huguette Moreau', 19, 'F', '1010 Rue des Étoiles, Cité Céleste, Ville Éclatante', 'No', 'Close', 'Depression', 34, 104),
(5, 'Camille Dubois', 29, 'M', '1313 Boulevard des Montagnes, Quartier Panoramique, Ville Vue', 'Yes', 'Close', 'Anxiety', 35, 105),
(6, 'Nathan Lefevre', 54, 'M', '1515 Allée des Chênes, Quartier Boisé, Ville Verdure', 'Yes', 'Close', 'Coronary artery disease', 36, 106),
(7, 'Léa Renault', 5, 'F', '1717 Avenue des Oiseaux, Quartier Aérien, Ville Vol', 'Yes', 'Open', 'Chronic obstructive pulmonary disease', 37, 107),
(8, 'Enzo Girard', 68, 'M', '1919 Chemin des Vagues, Quartier Maritime, Ville Océan', 'Yes', 'Open', 'Migraine', 38, 108),
(9, 'Manon Martinez', 75, 'F', '2121 Rue des Brises, Quartier Ventilé, Ville Breeze', 'No', 'Open', 'Osteoarthritis', 39, 109),

(10, 'Théo Rousseau', 82, 'M', '2323 Boulevard des Lumières, Quartier Brillant, Ville Scintillante', 'Yes', 'Open', 'Chronic kidney disease', 40, 200);

INSERT INTO consult (ID_DOCTOR, ID_PATIENTS, Date_Conconsult) VALUES

(1111, 1, '2022-04-10'),

(1112, 2, '2022-12-25'),

(1113, 3, '2022-02-02'),

(1114, 4, '2022-12-25'),

(1115, 5, '2022-03-06'),

(1115, 6, '2022-08-09'),

(1111, 7, '2022-04-10'),

(1118, 8, '2022-02-02'),

(1111, 9, '2022-04-10'),

(1110, 10, '2022-06-20');

INSERT INTO suffer (ID_PATIENTS, DISEASE_NAME) VALUES

(1, 'Cancer'),

(2, 'Diabetes'),

(3, 'HIV'),

(4, 'Hepatitis'),

(5, 'Mental illnesses'),

(6, 'Cancer'),

(7, 'HIV'),

(8, 'Diabetes'),

(9, 'Cancer'),

(10, 'Respiratory disease');

INSERT INTO follow (ID_PATIENTS, ID_NURSES) VALUES

(1, 11),

(2, 12),

(3, 13),

(4, 14),

(5, 15),

(6, 16),

(7, 17),

(8, 18),

(9, 19),

(10, 20);

INSERT INTO pay (ID_PATIENTS, ID_INVOICE) VALUES

(1, 201),

(2, 202),

(3, 203),

(4, 204),

(5, 205),

(6, 206),

(7, 207),

(8, 208),

(9, 209),

(10, 300);

INSERT INTO **rendezvous** (**ID_PATIENTS, ID_AD_STAFF, Date_rendezvous**)
VALUES

(1, 121, '2022-12-12'),

(2, 122, '2022-12-25'),

(3, 123, '2022-02-02'),

(4, 124, '2022-12-25'),

(5, 125, '2022-03-06'),

(6, 126, '2022-08-09'),

(7, 127, '2022-05-04'),

(8, 128, '2022-02-02'),

(9, 129, '2022-07-15'),

(10, 130, '2022-06-20');

INSERT INTO **Treat** (**ID_DOCTOR, DISEASE_NAME**) **VALUES**

(1111, 'Cancer'),

(1112, 'Diabetes'),

(1113, 'HIV'),

(1114, 'Hepatitis'),

(1115, 'Mental illnesses'),

(1116, 'Chronic diseases'),

(1117, 'Obesity'),

(1118, 'Allergies'),

(1119, 'Sleep disorders'),































(1110, 'Respiratory disease');

Part 2 : Queries and SQL solutions

2.1 Basic Queries

1. What are the reasons for admissions to the emergency department?

```
SELECT Reason
FROM `emergency_management`;
```

	Reason
<input type="checkbox"/>  Éditer  Copier  Supprimer	Fever
<input type="checkbox"/>  Éditer  Copier  Supprimer	Headache
<input type="checkbox"/>  Éditer  Copier  Supprimer	Dizziness
<input type="checkbox"/>  Éditer  Copier  Supprimer	Diarrhea
<input type="checkbox"/>  Éditer  Copier  Supprimer	Headache
<input type="checkbox"/>  Éditer  Copier  Supprimer	Fatigue
<input type="checkbox"/>  Éditer  Copier  Supprimer	Dizziness
<input type="checkbox"/>  Éditer  Copier  Supprimer	Headache
<input type="checkbox"/>  Éditer  Copier  Supprimer	Fever
<input type="checkbox"/>  Éditer  Copier  Supprimer	Fatigue

2. What are the names of the patients?

```
SELECT Name
FROM `patients`;
```

←T→	Name
<input type="checkbox"/> Éditer Copier Supprimer	Sophie Martin
<input type="checkbox"/> Éditer Copier Supprimer	Lucas Dupont
<input type="checkbox"/> Éditer Copier Supprimer	Emma Leroy
<input type="checkbox"/> Éditer Copier Supprimer	Huguette Moreau
<input type="checkbox"/> Éditer Copier Supprimer	Camille Dubois
<input type="checkbox"/> Éditer Copier Supprimer	Nathan Lefevre
<input type="checkbox"/> Éditer Copier Supprimer	Léa Renault
<input type="checkbox"/> Éditer Copier Supprimer	Enzo Girard
<input type="checkbox"/> Éditer Copier Supprimer	Manon Martinez
<input type="checkbox"/> Éditer Copier Supprimer	Théo Rousseau

3. What are the characteristics of the patients?

```
SELECT *
FROM patients ;
```

←T→	ID_PATIENTS	Name	Sex	Age	Address	Medical_History	File_Status	Diagnosis	ID_ROOMS	ID_EMERGENCY
Éditer Copier Supprimer	1	Sophie Martin	F	19	123 Rue du Soleil Levant, Ville Lumière, État de L...	Yes	Open	Hypertension	31	101
Éditer Copier Supprimer	2	Lucas Dupont	M	26	456 Avenue des Cyprés, Quartier Tranquille, Ville ...	Yes	Close	Diabetes	32	102
Éditer Copier Supprimer	3	Emma Leroy	F	33	789 Chemin des Marguerites, Quartier Fleuri, Ville...	No	Close	Asthma	33	103
Éditer Copier Supprimer	4	Huguette Moreau	F	19	1010 Rue des Étoiles, Cité Céleste, Ville Éclatant...	No	Close	Depression	34	104
Éditer Copier Supprimer	5	Camille ...	M	29	1313 Boulevard des Montagnes,	Yes	Close	Anxiety	35	105

4. What are the names of the doctors?

```
SELECT Name
FROM `doctors`;
```

←T→	Name
<input type="checkbox"/> Éditer Copier Supprimer	Mathis Fortin
<input type="checkbox"/> Éditer Copier Supprimer	Sarah Dupuis
<input type="checkbox"/> Éditer Copier Supprimer	Maxime Leclerc
<input type="checkbox"/> Éditer Copier Supprimer	Zoé Bernard
<input type="checkbox"/> Éditer Copier Supprimer	Nathan Roy
<input type="checkbox"/> Éditer Copier Supprimer	Jade Lavoie
<input type="checkbox"/> Éditer Copier Supprimer	Alexandre Gagnon
<input type="checkbox"/> Éditer Copier Supprimer	Maëlle Caron
<input type="checkbox"/> Éditer Copier Supprimer	Gabriel Tremblay
<input type="checkbox"/> Éditer Copier Supprimer	Charlotte Desjardins

5. What is the status of the rooms?

SELECT STATUS

FROM `rooms`;

←T→	STATUS
<input type="checkbox"/> Éditer Copier Supprimer	Busy
<input type="checkbox"/> Éditer Copier Supprimer	Free
<input type="checkbox"/> Éditer Copier Supprimer	Busy
<input type="checkbox"/> Éditer Copier Supprimer	Free
<input type="checkbox"/> Éditer Copier Supprimer	Busy
<input type="checkbox"/> Éditer Copier Supprimer	Free
<input type="checkbox"/> Éditer Copier Supprimer	Free
<input type="checkbox"/> Éditer Copier Supprimer	Busy
<input type="checkbox"/> Éditer Copier Supprimer	Free
<input type="checkbox"/> Éditer Copier Supprimer	Free




2.2 WHERE Clause Queries

1. To which department does Dr. Sarah belong?

```
SELECT ID_SERVICE, Status

FROM doctors

WHERE `Name` = 'Sarah Dupuis' ;
```

←T→	ID_SERVICE	Status
<input type="checkbox"/>  Éditer  Copier  Supprimer	21	Contract

2. Which medical equipment was functional and maintained on March 15, 2022 ?

```
SELECT Name

FROM medical_equipment

WHERE Date_of_Maintenance = '2022-03-15' AND Status
='functional';
```

←T→	Name
<input type="checkbox"/>  Éditer  Copier  Supprimer	X-ray machine
<input type="checkbox"/>  Éditer  Copier  Supprimer	ECG machine (Electrocardiogram)
<input type="checkbox"/>  Éditer  Copier  Supprimer	Defibrillator

3. What are the names of the contractual nurses earning a salary of 3000 EUR or more?

```
SELECT Name

FROM nurses

WHERE Status = 'contractual' AND Salary >=3000;
```

←T→	Name
<input type="checkbox"/>  Éditer  Copier  Supprimer	Marie Dubois
<input type="checkbox"/>  Éditer  Copier  Supprimer	Antoine Leroy

4. What are the names of the female administrative staff members who work as medical secretaries?

```
SELECT Name
```

FROM administrative_staff

WHERE Gender = 'F' AND Position= 'Medical Secretary';


					Name	
<input type="checkbox"/>		Éditer		Copier	 Supprimer	Théola Girard
<input type="checkbox"/>		Éditer		Copier	 Supprimer	Léa Martin

5. What is the list of medications whose names start with the letter 'D'?

SELECT*

FROM drugs

WHERE Name LIKE 'D%';

<div><div>←</div><div>T</div><div>→</div></div>				ID_DRUGS	Name	Expiry_date	Price
<div><div><input type="checkbox"/></div><div></div><div>Éditer</div><div></div><div>Copier</div><div></div><div>Supprimer</div></div>	303	Diazepam	2025-05-20	6			
<div><div><input type="checkbox"/></div><div></div><div>Éditer</div><div></div><div>Copier</div><div></div><div>Supprimer</div></div>	305	Doliprane	2024-11-28	3			
<div><div><input type="checkbox"/></div><div></div><div>Éditer</div><div></div><div>Copier</div><div></div><div>Supprimer</div></div>	308	Dexamethasone	2023-10-18	25			

2.3 ORDER BY queries

1. Display the names of the patients in alphabetical order ?

SELECT Name

FROM patients

ORDER BY Name **ASC**;

							Name ▲ 1
<input type="checkbox"/>		Éditer		Copier		Supprimer	Camille Dubois
<input type="checkbox"/>		Éditer		Copier		Supprimer	Emma Leroy
<input type="checkbox"/>		Éditer		Copier		Supprimer	Enzo Girard
<input type="checkbox"/>		Éditer		Copier		Supprimer	Huguette Moreau
<input type="checkbox"/>		Éditer		Copier		Supprimer	Léa Renault
<input type="checkbox"/>		Éditer		Copier		Supprimer	Lucas Dupont
<input type="checkbox"/>		Éditer		Copier		Supprimer	Manon Martinez
<input type="checkbox"/>		Éditer		Copier		Supprimer	Nathan Lefevre
<input type="checkbox"/>		Éditer		Copier		Supprimer	Sophie Martin
<input type="checkbox"/>		Éditer		Copier		Supprimer	Théo Rousseau

2. List the medications sorted by their expiration date in ascending order



































































































































```
SELECT *
FROM DRUGS
ORDER BY Expiry_date ASC;
```

<div><div><div><div></div><div></div><div></div></div><div>←T→</div></div></div>						ID_DRUGS	Name	Expiry_date ▲ 1	Price
<input type="checkbox"/>	 Éditer	 Copier	 Supprimer	304 Amoxicillin	2023-08-10	3			
<input type="checkbox"/>	 Éditer	 Copier	 Supprimer	308 Dexamethasone	2023-10-18	25			
<input type="checkbox"/>	 Éditer	 Copier	 Supprimer	302 Ibuprofen	2023-12-15	6			
<input type="checkbox"/>	 Éditer	 Copier	 Supprimer	400 Vitamin	2024-04-08	20			
<input type="checkbox"/>	 Éditer	 Copier	 Supprimer	307 Metformin	2024-07-05	15			
<input type="checkbox"/>	 Éditer	 Copier	 Supprimer	301 Paracetamol	2024-09-30	2			
<input type="checkbox"/>	 Éditer	 Copier	 Supprimer	305 Doliprane	2024-11-28	3			
<input type="checkbox"/>	 Éditer	 Copier	 Supprimer	309 Lorazepam	2025-01-25	30			
<input type="checkbox"/>	 Éditer	 Copier	 Supprimer	306 Salbutamol	2025-03-12	10			
<input type="checkbox"/>	 Éditer	 Copier	 Supprimer	303 Diazepam	2025-05-20	6			

3. List of doctors grouped by gender with their salaries in ascending order

```
SELECT Name, Gender, Salary
FROM DOCTORS
```

ORDER BY Gender, Salary ASC;

				Name	Gender	Salary
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer






























































4. Show the payment methods in alphabetical order, along with the date and time of payment in chronological order

SELECT Payment_Method,Date,Time

FROM BILLING

ORDER BY Payment_Method ASC,Date,Time

ASC,Payment_Method ASC;

				Payment_Method	Date	Time
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer
<input type="checkbox"/>		Éditer		Copier		Supprimer

5. Display the medical equipment sorted by the date of maintenance in descending chronological order, followed by the price of maintenance in descending order

```
SELECT *
FROM MEDICAL_EQUIPMENT
ORDER BY Date_of_Maintenance DESC, Price_of_Maintenance
DESC;
```

← T →	ID_EQUIPMENT	Name	Status	Type_of_Maintenance	Date_of_Maintenance ▼ 1	Price_of_Maintenance ▼ 2
<input type="checkbox"/> Éditer Copier Supprimer	75	Blood pressure monitor	unfunctional	Corrective Maintenance	2022-05-10	200
<input type="checkbox"/> Éditer Copier Supprimer	80	Electroencephalogram (EEG) machine	functional	Adaptive Maintenance	2022-04-02	500
<input type="checkbox"/> Éditer Copier Supprimer	79	Surgical microscope	functional	Maintenance préventive	2022-04-02	300
<input type="checkbox"/> Éditer Copier Supprimer	72	MRI scanner (Magnetic Resonance Imaging)	unfunctional	Adaptive Maintenance	2022-03-17	200
<input type="checkbox"/> Éditer Copier Supprimer	71	X-ray machine	functional	Maintenance préventive	2022-03-15	500
<input type="checkbox"/> Éditer Copier Supprimer	74	ECG machine (Electrocardiogram)	functional	Adaptive Maintenance	2022-03-15	500
<input type="checkbox"/> Éditer Copier Supprimer	77	Defibrillator	functional	Corrective Maintenance	2022-03-15	100
<input type="checkbox"/> Éditer Copier Supprimer	78	Infusion pump	functional	Corrective Maintenance	2022-03-06	200
<input type="checkbox"/> Éditer Copier Supprimer	76	Ventilator	unfunctional	Maintenance préventive	2022-03-06	100
<input type="checkbox"/> Éditer Copier Supprimer	73	Ultrasound machine	functional	Adaptive Maintenance	2022-01-05	400

2.4 Queries with Numeric expressions and functions;

1. What is the total salary of all the nurses?

```
SELECT SUM(Salary) AS TotalSalary
FROM DOCTORS;
```

TotalSalary
51000

2. Display the names of the doctors, their salaries, and their reduced salaries (Salary - (Salary/2)) in ascending order

```
SELECT Name, Salary, (Salary-(Salary/2)) AS "Salary reduce"
FROM doctors
ORDER BY (Salary-(Salary/2)) ASC;
```

←T→						Name	Salary	Salary reduce	
<input type="checkbox"/>		Éditer		Copier		Supprimer	Mathis Fortin	3000	1500.0000
<input type="checkbox"/>		Éditer		Copier		Supprimer	Gabriel Tremblay	3000	1500.0000
<input type="checkbox"/>		Éditer		Copier		Supprimer	Jade Lavoie	3000	1500.0000
<input type="checkbox"/>		Éditer		Copier		Supprimer	Charlotte Desjardins	3000	1500.0000
<input type="checkbox"/>		Éditer		Copier		Supprimer	Nathan Roy	4000	2000.0000
<input type="checkbox"/>		Éditer		Copier		Supprimer	Maxime Leclerc	6000	3000.0000
<input type="checkbox"/>		Éditer		Copier		Supprimer	Alexandre Gagnon	6000	3000.0000
<input type="checkbox"/>		Éditer		Copier		Supprimer	Maëlle Caron	6000	3000.0000
<input type="checkbox"/>		Éditer		Copier		Supprimer	Zoé Bernard	8000	4000.0000
<input type="checkbox"/>		Éditer		Copier		Supprimer	Sarah Dupuis	9000	4500.0000

3. What is the age difference between the oldest and youngest patients (MAX(Age) - MIN(Age)), and what is the minimum age plus five (MIN(Age) + 5)?

```
SELECT MAX(Age) - MIN(Age) AS
"Age_difference", (MIN(Age)*2) AS "Age_min"

FROM patients

WHERE Age > 5;
```




Age_difference	Age_min
63	38

4. What is the total sum of salaries for contractual nurses?

```
SELECT Status, SUM(Salary) AS total_Salary

FROM nurses

WHERE Status="Contractual";
```

<div>←T→</div>					Status	total_Salary		
<input type="checkbox"/>		Éditer		Copier		Supprimer	Contractual	9000

5. How many patients are diagnosed with cancer?

```

SELECT COUNT(*) AS nombre_patients_Cancer

FROM SUFFER

WHERE DISEASE_NAME = 'Cancer';

```

nombre_patients_Cancer
3

2.5 GROUP BY Queries

1. Show the room number, the count of patients, and their names grouped by room number

```

SELECT r.ID_ROOMS, COUNT(p.ID_PATIENTS) AS nombre_patients, Name
FROM ROOMS r, PATIENTS p
WHERE r.ID_ROOMS = p.ID_ROOMS
GROUP BY r.ID_ROOMS;

```

ID_ROOMS	nombre_patients	Name
31	1	Sophie Martin
32	1	Lucas Dupont
33	1	Emma Leroy
34	1	Huguette Moreau
35	1	Camille Dubois
36	1	Nathan Lefevre
37	1	Léa Renault
38	1	Enzo Girard
39	1	Manon Martinez
40	1	Théo Rousseau

2. Display the number of consultations per month

```

SELECT MONTH(c.Date_consult) AS mois, COUNT(*) AS
nombre_consultations
FROM CONSULT c
GROUP BY MONTH(c.Date_consult);

```































mois	nombre_consultations
2	2
3	1
4	3
6	1
8	1
12	2

3. Present the total salary of doctors by type of service

```

SELECT s.Type, SUM(d.Salary) AS total_salaires
FROM services s, doctors d
WHERE s.ID_SERVICE = d.ID_SERVICE
GROUP BY s.Type;

```

	Type	total_salaires
<input type="checkbox"/>  Éditer  Copier  Supprimer	Cardiology	6000
<input type="checkbox"/>  Éditer  Copier  Supprimer	Emergency department	9000
<input type="checkbox"/>  Éditer  Copier  Supprimer	General medicine	6000
<input type="checkbox"/>  Éditer  Copier  Supprimer	Neurology	6000
<input type="checkbox"/>  Éditer  Copier  Supprimer	Obstetrics and gynecology	3000
<input type="checkbox"/>  Éditer  Copier  Supprimer	Pediatrics	4000
<input type="checkbox"/>  Éditer  Copier  Supprimer	Psychiatry	3000
<input type="checkbox"/>  Éditer  Copier  Supprimer	Radiology	3000
<input type="checkbox"/>  Éditer  Copier  Supprimer	Rehabilitation service	3000
<input type="checkbox"/>  Éditer  Copier  Supprimer	Surgery	8000

4. Reveal the number of disease treatments sorted by each disease

```

SELECT t.DISEASE_NAME,
COUNT(*) AS nombre_traitements
FROM treat t
GROUP BY t.DISEASE_NAME;

```

DISEASE_NAME	nombre_traitements
Allergies	1
Cancer	1
Chronic diseases	1
Diabetes	1
Hepatitis	1
HIV	1
Mental illnesses	1
Obesity	1
Respiratory disease	1
Sleep disorders	1

5. Calculate the total number of invoices per payment method

```

SELECT    Payment_Method,    COUNT(ID_INVOICE)    AS
Total_Invoices

FROM BILLING

GROUP BY Payment_Method;
```

Payment_Method	Total_Invoices
Cash	5
Credit Card	5

2.6 Multi-Table queries

1. List of patients, the doctors who treated them, and the diagnosed diseases

```

SELECT    PATIENTS.Name    AS    Patient_Name,    DOCTORS.Name    AS
Doctor_Name, DISEASE.DISEASE_NAME
FROM PATIENTS
JOIN CONSULT ON PATIENTS.ID_PATIENTS = CONSULT.ID_PATIENTS
JOIN DOCTORS ON CONSULT.ID_DOCTOR = DOCTORS.ID_Doctor
JOIN TREAT ON DOCTORS.ID_Doctor = TREAT.ID_DOCTOR
JOIN DISEASE ON TREAT.DISEASE_NAME = DISEASE.DISEASE_NAME;
```

Patient_Name	Doctor_Name	DISEASE_NAME
Théo Rousseau	Mathis Fortin	Respiratory disease
Sophie Martin	Sarah Dupuis	Cancer
Léa Renault	Sarah Dupuis	Cancer
Manon Martinez	Sarah Dupuis	Cancer
Lucas Dupont	Maxime Leclerc	Diabetes
Emma Leroy	Zoé Bernard	HIV
Huguette Moreau	Nathan Roy	Hepatitis
Camille Dubois	Jade Lavoie	Mental illnesses
Nathan Lefevre	Jade Lavoie	Mental illnesses
Enzo Girard	Gabriel Tremblay	Allergies

2. List the medical equipment, their status, and the laboratory Where they are used

```

SELECT MEDICAL_EQUIPMENT.Name AS Equipment_Name,
MEDICAL_EQUIPMENT.Status, LABORATORY.Specialty

FROM MEDICAL_EQUIPMENT

JOIN EQUIP ON MEDICAL_EQUIPMENT.ID_EQUIPMENT =
EQUIP.ID_EQUIPMENT

JOIN LABORATORY ON EQUIP.ID_LABORATORY =
LABORATORY.ID_LABORATORY;

```

Equipment_Name	Status	Specialty
X-ray machine	functional	Clinical Laboratory
MRI scanner (Magnetic Resonance Imaging)	unfunctional	Microbiology Laboratory
Ultrasound machine	functional	Anatomic Pathology Laboratory
ECG machine (Electrocardiogram)	functional	Medical Genetics Laboratory
Blood pressure monitor	unfunctional	Cardiology Laboratory
Ventilator	unfunctional	Toxicology Laboratory
Defibrillator	functional	Hematology Laboratory
Infusion pump	functional	Quality Control
Surgical microscope	functional	Genetic Testing
Electroencephalogram (EEG) machine	functional	Radiographie

3. List of nurses, the doctors they assist, and the departments they work in


```

SELECT NURSES.Name AS Nurse_Name, DOCTORS.Name AS Doctor_Name,
SERVICES.Type AS Service_Type
FROM NURSES
JOIN ASSIST ON NURSES.ID_NURSES = ASSIST.ID_NURSES
JOIN DOCTORS ON ASSIST.ID_DOCTOR = DOCTORS.ID_Doctor
JOIN SERVICES ON DOCTORS.ID_SERVICE = SERVICES.ID_SERVICE;

```

Nurse_Name	Doctor_Name	Service_Type
Antoine Leroy	Mathis Fortin	Psychiatry
Marie Dubois	Sarah Dupuis	Emergency department
Julien Lefevre	Maxime Leclerc	General medicine
Anaïs Martin	Zoé Bernard	Surgery
Lucas Dupont	Nathan Roy	Pediatrics
Sarah Lambert	Jade Lavoie	Obstetrics and gynecology
Thomas Laurent	Alexandre Gagnon	Cardiology
Camille Moreau	Maëlle Caron	Neurology
Pierre Girard	Gabriel Tremblay	Radiology
Émilie Rousseau	Charlotte Desjardins	Rehabilitation service

4. List the patients, their appointments, and the administrative staff who scheduled them

```

SELECT PATIENTS.Name AS Patient_Name,
APPOINTEMENT.Date_Appointment, ADMINISTRATIVE_STAFF.Name AS
Staff_Name
FROM PATIENTS
JOIN APPOINTEMENT ON PATIENTS.ID_PATIENTS =
APPOINTEMENT.ID_PATIENTS
JOIN ADMINISTRATIVE_STAFF ON APPOINTEMENT.ID_AD_STAFF =
ADMINISTRATIVE_STAFF.ID_AD_STAFF;

```

Patient_Name	Date_Appointment	Staff_Name
Sophie Martin	2022-12-12	Elodie Morel
Lucas Dupont	2022-12-25	Victor Lefebvre
Emma Leroy	2022-02-02	Manon Dupuis
Huguette Moreau	2022-12-25	Théola Girard
Camille Dubois	2022-03-06	Camille Assani
Nathan Lefevre	2022-08-09	Lucas Lambert
Léa Renault	2022-05-04	Léa Martin
Enzo Girard	2022-02-02	Hugo Koffi
Manon Martinez	2022-07-15	Chloé Bernard
Théo Rousseau	2022-06-20	Nina Leroy

5. Display the names of patients diagnosed with cancer

```

SELECT PATIENTS.Name
FROM PATIENTS
JOIN SUFFER ON PATIENTS.ID_PATIENTS = SUFFER.ID_PATIENTS
WHERE SUFFER.DISEASE_NAME = 'Cancer';

```

Name
Sophie Martin
Nathan Lefevre
Manon Martinez

2.7 Nested queries

1. Identify the total number of beds available in rooms that are currently unoccupied:

```

SELECT SUM(R.Number_of_beds) AS Total_Available_Beds
FROM ROOMS R
WHERE R.Status = 'free' AND NOT EXISTS (
SELECT 1
FROM BEDS B

```

**WHERE B.Availability = 'Unavailable' AND B.ID_ROOMS =
R.ID_ROOMS);**

Total_Available_Beds

5

2. What are the medical equipment that are functional and have undergone 'Maintenance préventive', along with the names of laboratories where they are used ?

**SELECT ME.Name, L.Specialty
FROM MEDICAL_EQUIPMENT ME
JOIN EQUIPE E ON ME.ID_EQUIPMENT = E.ID_EQUIPMENT
JOIN LABORATORY L ON E.ID_LABORATORY = L.ID_LABORATORY
WHERE ME.Status = 'functional' AND ME.Type_of_Maintenance =
'Maintenance préventive';**

Name	Specialty
X-ray machine	Clinical Laboratory
Surgical microscope	Genetic Testing

3. Retrieve the total price of maintenance for all functional medical equipment delivered by suppliers named 'MedEquip Solutions':

**SELECT SUM(ME.Price_of_Maintenance) AS Total_Maintenance_Cost
FROM MEDICAL_EQUIPMENT ME
JOIN DELIVRER D ON ME.ID_EQUIPMENT = D.ID_EQUIPMENT
JOIN SUPPLIERS S ON D.ID_SUPPLIER = S.ID_SUPPLIER
WHERE ME.Status = 'functional' AND S.Name = 'MedEquip Solutions';**

Total_Maintenance_Cost

500

4. What is the total number of drugs contained in each pharmacy?

```

SELECT P.Name_of_Pharmacy, COUNT(*) AS Drug_Count
FROM PHARMACY P
JOIN CONTAINED C ON P.ID_PHARMACY = C.ID_PHARMACY
GROUP BY P.Name_of_Pharmacy;

```

Name_of_Pharmacy	Drug_Count
Aid Aid Pharmacy	1
Care Mart Pharmacy	1
Cure Corner Pharmacy	1
Easy Meds Pharmacy	1
Health Plus Pharmacy	1
LifeLine Pharmacy	1
MediShop Pharmacy	1
PillBox Pharmacy	1
Quick Relief Pharmacy	1
Wellness Pharmacy	1

5. List the medical equipment delivered by 'Healthcare Supply Co.' and their maintenance dates

```

SELECT ME.Name, ME.Date_of_Maintenance
FROM MEDICAL_EQUIPMENT ME
JOIN DELIVRER D ON ME.ID_EQUIPMENT = D.ID_EQUIPMENT
JOIN SUPPLIERS S ON D.ID_SUPPLIER = S.ID_SUPPLIER
WHERE S.Name = 'Healthcare Supply Co.';

```

Name	Date_of_Maintenance
MRI scanner (Magnetic Resonance Imaging)	2022-03-17

Part3 : MONGODB

From our model, we have chosen the following entities:

1. PATIENTS
2. DOCTORS
3. DISEASES

3.1 The 3 .json files containing all your data (1 file per entity)

1. Disease.json
2. Patients.json
3. Doctors.json

3.2 The insert script (for this, you shall use the mongoimport command to import the 3 .json files)

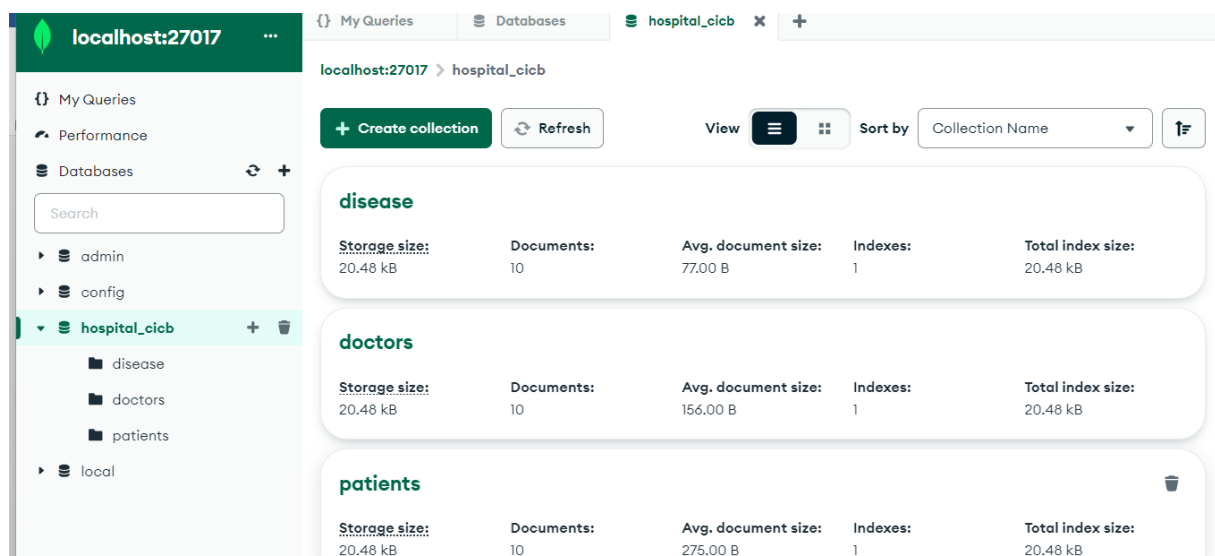
Since mongoimport does not exist in the original bin folder of MongoDB, we download it from the official MongoDB page and thus add the mongoimport command to our machine's directory.

```
C:\Users\ASUS>mongoimport --db hospital_cicb --collection patients --file "C:\Users\ASUS\Desktop\Final\patients.json" --jsonArray
2024-05-10T20:21:32.903+0200    connected to: mongodb://localhost/
2024-05-10T20:21:33.157+0200    10 document(s) imported successfully. 0 document(s) failed to import.

C:\Users\ASUS>mongoimport --db hospital_cicb --collection doctors --file "C:\Users\ASUS\Desktop\Final\doctors.json" --jsonArray
2024-05-10T20:21:46.189+0200    connected to: mongodb://localhost/
2024-05-10T20:21:46.470+0200    10 document(s) imported successfully. 0 document(s) failed to import.

C:\Users\ASUS>mongoimport --db hospital_cicb --collection disease --file "C:\Users\ASUS\Desktop\Final\disease.json" --jsonArray
2024-05-10T20:21:59.354+0200    connected to: mongodb://localhost/
2024-05-10T20:21:59.637+0200    10 document(s) imported successfully. 0 document(s) failed to import.

C:\Users\ASUS>
```



Collection	Storage size	Documents	Avg. document size	Indexes	Total index size
disease	20.48 kB	10	77.00 B	1	20.48 kB
doctors	20.48 kB	10	156.00 B	1	20.48 kB
patients	20.48 kB	10	275.00 B	1	20.48 kB

3.3 Python script

Connection to the MongoDB database:

```
from pymongo import MongoClient

client = MongoClient('localhost', 27017)
db = client.hospital_cicb
patients_collection = db.patients
```

Query 1: Who are the female patients with medical records?

```
patients_F = patients_collection.find({"Sex": "F", "Medical_History": "Yes"})
print("Female patients with medical records are:")
for patient in patients_F:
    print(patient)
```

Female patients with medical records are:

```
{'_id': ObjectId('663e65ac4dcf4428bf37dd2c'), 'ID_PATIENTS': '7', 'Name': 'Léa Renault', 'Sex': 'F', 'Age': '5', 'Address': '1717 Avenue des Oiseaux, Quartier Aérien, Ville Vol', 'Medical_History': 'Yes', 'File_Status': 'Open', 'Diagnosis': 'Chronic obstructive pulmonary disease', 'ID_ROOMS': '37', 'ID_EMERGENCY': '107'}
{'_id': ObjectId('663e65ac4dcf4428bf37dd31'), 'ID_PATIENTS': '1', 'Name': 'Sophie Martin', 'Sex': 'F', 'Age': '19', 'Address': '123 Rue du Soleil Levant, Ville Lumière, État de l'Aurore', 'Medical_History': 'Yes', 'File_Status': 'Open', 'Diagnosis': 'Hypertension', 'ID_ROOMS': '31', 'ID_EMERGENCY': '101'}
```

Query 2: Who are the Patient Diagnosed with Diabetes?

```
patients_diabetes = patients_collection.find({"Diagnosis": "Diabetes"})
print("\nPatients with a Diabetes Diagnosis:")
for patient in patients_diabetes:
    print(patient)
```

Patients with a Diabetes Diagnosis:

```
{'_id': ObjectId('663e65ac4dcf4428bf37dd2f'), 'ID_PATIENTS': '2', 'Name': 'Lucas Dupont', 'Sex': 'M', 'Age': '26', 'Address': '456 Avenue des Cyprès, Quartier Tranquille, Ville Paisible', 'Medical_History': 'Yes', 'File_Status': 'Close', 'Diagnosis': 'Diabetes', 'ID_ROOMS': '32', 'ID_EMERGENCY': '102'}
```

Query 3: Who are the Patients over 60 years old?

```
patients_age = patients_collection.find({"Age": {"$gt": "60"}})
print("\nPatients over 60 years of age:")
for patient in patients_age:
    print(patient)
```

Patients over 60 years of age:

```
{'_id': ObjectId('663e65ac4dcf4428bf37dd2d'), 'ID_PATIENTS': '8', 'Name': 'Enzo Girard', 'Sex': 'M', 'Age': '68', 'Address': '1919 Chemin des Vagues, Quartier Maritime, Ville Océan', 'Medical_History': 'Yes', 'File_Status': 'Open', 'Diagnosis': 'Migraine', 'ID_ROOMS': '38', 'ID_EMERGENCY': '108'}
{'_id': ObjectId('663e65ac4dcf4428bf37dd2e'), 'ID_PATIENTS': '9', 'Name': 'Manon Martinez', 'Sex': 'F', 'Age': '75', 'Address': '2121 Rue des Brises, Quartier Ventilé, Ville Breeze', 'Medical_History': 'No', 'File_Status': 'Open', 'Diagnosis': 'Osteoarthritis', 'ID_ROOMS': '39', 'ID_EMERGENCY': '109'}
{'_id': ObjectId('663e65ac4dcf4428bf37dd30'), 'ID_PATIENTS': '10', 'Name': 'Théo Rousseau', 'Sex': 'M', 'Age': '82', 'Address': '2323 Boulevard des Lumières, Quartier Brillant, Ville Scintillante', 'Medical_History': 'Yes', 'File_Status': 'Open', 'Diagnosis': 'Chronic kidney disease', 'ID_ROOMS': '40', 'ID_EMERGENCY': '200'}
```

Query 4: Who are the Male patients under 70 years old with an open medical record?

```

patients_M = patients_collection.find({"Sex": "M", "Age": {"$lt": "70"}, "Medical_History": "Yes", "File_Status": "Open"}
print("Male patient under the age of 70 with an open medical record is:")
for patient in patients_M:
    print("Patient Name:", patient["Name"])
    print("Patient Gender:", patient["Sex"])
    print("The age of the patient:", patient["Age"])
    print("Patient Medical_History:", patient["Medical_History"])
    print("Patient File_Status:", patient["File_Status"])

```

```

Male patient under the age of 70 with an open medical record is:
Patient Name: Enzo Girard
Patient Gender: M
The age of the patient: 68
Patient Medical_History: Yes
Patient File_Status: Open

```

Query 5: What is the name, gender and address of the patient who is in Emergency 103 in room 33?

```

patient = patients_collection.find_one({"ID_EMERGENCY": "103", "ID_ROOMS": "33"})
if patient:
    print("Patient Name:", patient["Name"])
    print("Patient Gender:", patient["Sex"])
    print("Patient address:", patient["Address"])
else:
    print("No patients were found with these identifiers.")

```

```

Patient Name: Emma Leroy
Patient Gender: F
Patient address: 789 Chemin des Marguerites, Quartier Fleuri, Ville Jolie

```

Conclusion

The 'hospital_cicb' database was constructed with the goal of being as close as possible to the structure of a hospital. Thus, a complete and complex database was established following the logical data model. Generally, it revolves around patients, the medical staff, non-medical administrative personnel, equipment, and other physical components of a hospital such as laboratories, pharmacies, cashiers, etc. The tables do not contain many attributes to detail the objects more in-depth, but this limitation does not change the quality of the model (tables and managed relationships). The potential of the database could be illustrated by the SQL and NoSQL queries that were presented in the report. The possibilities are vast, and the number of examples inserted has allowed for the development of advanced queries that demonstrate their usefulness. Although containing imaginary data, this database is potentially capable of being adapted to a real hospital for visualizing its internal and external activity. However, it is always open to improvement.

