



DATABASE SYSTEM 1

HOSPITAL SYSTEM

Supervised BY:

1. Prof. Hanafy Ismail
2. T.A: Omar Khaled
3. T.A: Maryam Hany

Submitted BY:

- Mina Midhat Labib Hakem
- Mina Yakoup Waheeb Gad
- Mina Nagy Beshoy Fathalla
- Naeem Nazih Abdelshafi Hassan Naser
- Nour El-Din Safwan Mahmoud Abdelwahab
- Nour Mohsen Samir Abd Elmonaem

TABLE OF CONTENTS

Contents

1. INTRODUCTION	3
1. Maintain Basic Data	4
2. Issue query subSystems	4
3. Perform Transactions	4
4. Entity Relationship Diagram	5
5. Mapping	6
6. Database Relationships and Tables with Sample Data	7
7. Input and Output interface	8
8. Queries	12

1. INTRODUCTION

The Hospital Management System is a comprehensive database system project that aims to streamline and optimize the management of doctors, patients, and diagnosis information within a healthcare facility. This project is designed to enhance the efficiency, accuracy, and accessibility of storing, retrieving, and analyzing data related to medical professionals, patients, and their diagnoses.

Technologies Used:

The project utilizes the following technologies to achieve its objectives:

1. Relational Database Management System (RDBMS): The system employs an RDBMS such as SQL to store and manage the hospital's data efficiently. The relational model ensures data integrity, consistency, and flexibility in managing complex relationships between doctors, patients, and diagnoses.
2. SQL: Structured Query Language (SQL) is used to define the database schema, perform data manipulation, and execute complex queries for retrieving and analyzing data. SQL provides a standardized and efficient way to interact with the database.
3. Programming Language: The project is developed using a programming language such as Python or Java to implement the user interface, business logic, and database interactions. This enables the creation of a user-friendly interface and seamless integration with the database.

Conclusion:

The Hospital Management System Database System project aims to revolutionize the way hospitals manage their operations and patient information. By leveraging a well-designed database schema and utilizing SQL, the system enhances efficiency, accuracy, and accessibility in managing doctors, patients, and diagnosis information. This ultimately improves patient care, streamlines administrative processes, and supports evidence-based decision-making within the healthcare facility.

1. Maintain Basic Data

- **Doctor** (Docid(pk) , DocName , Docgen , Experience , License)
- **Patient** (PId(pk) , PName , PAddress , PAge , PPhone , pGen , BloodGroup , MajorDisease , DocId)
- **Diagnosis** (Did(pk) , PatientId , PatientName , Symptoms , DiagnosticTest , Medicines , DocId)

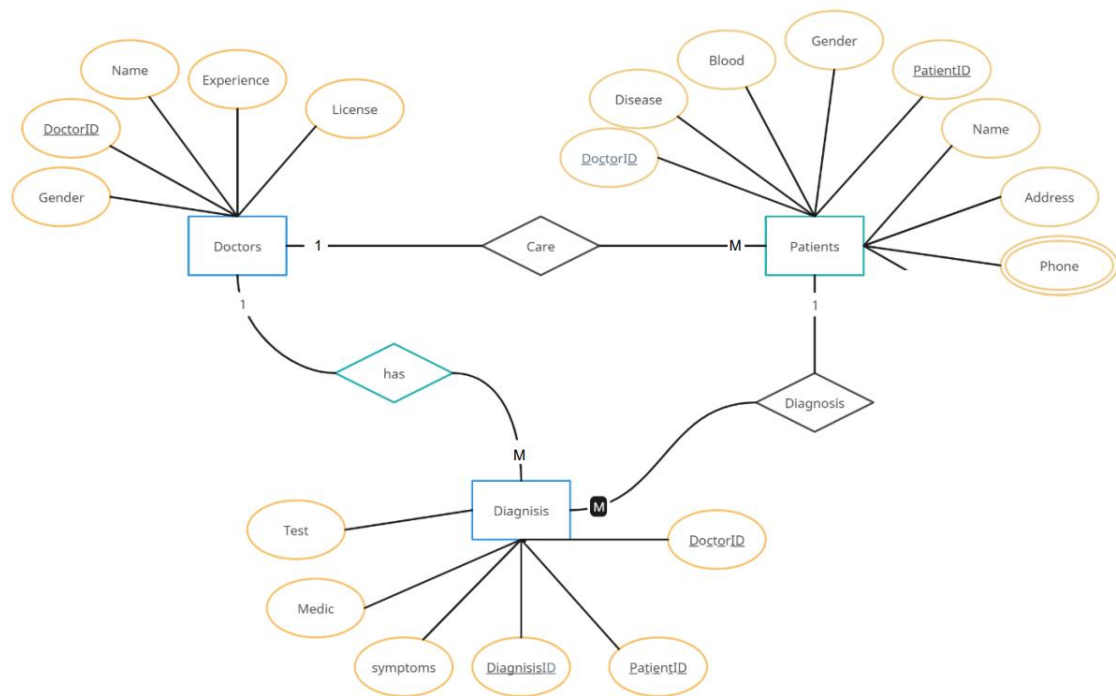
2. Issue query subSystems

- Search by Patient ID
- Search by Doctor ID
- Search by Diagnosis ID

3. Perform Transactions

- Add, delete, and update (all entities).

4. Entity Relationship Diagram



5. Mapping

Doctor:

DocId	DocName	License	Experience
-------	---------	---------	------------

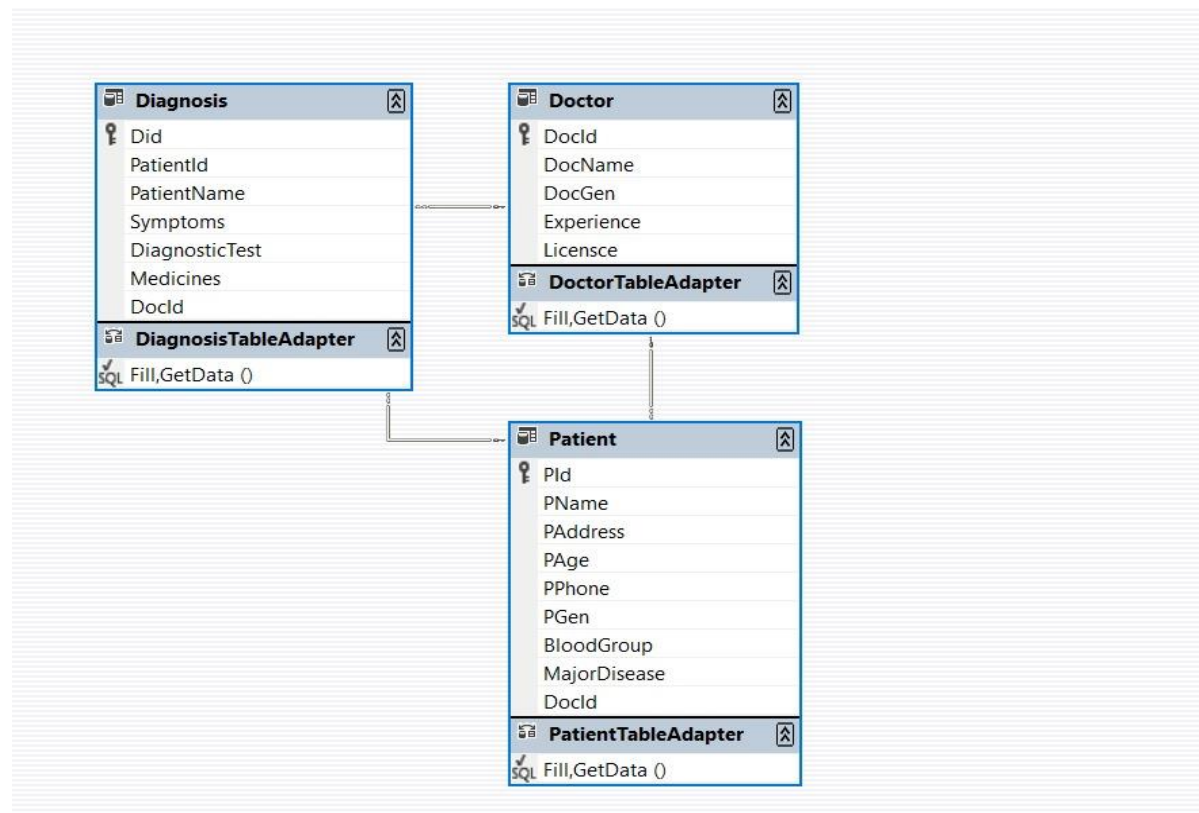
Patient:

PatientId	PName	Age	Phone	DocId	Address	Gender	Blood	Disease
-----------	-------	-----	-------	-------	---------	--------	-------	---------

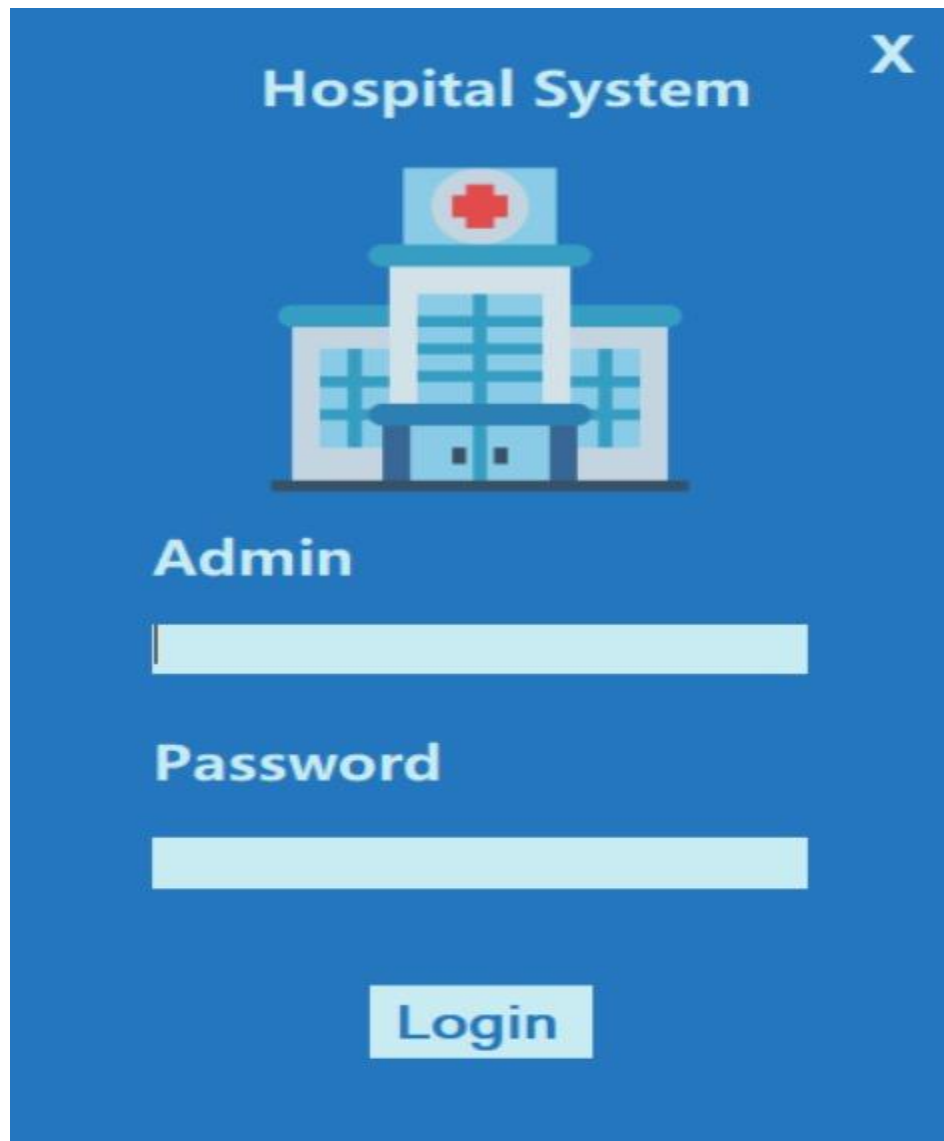
Diagnosis:

DiagnosisId	DocId	PatientId	Symptoms	Test	Medic
-------------	-------	-----------	----------	------	-------

6. Database Relationships and Tables with Sample Data

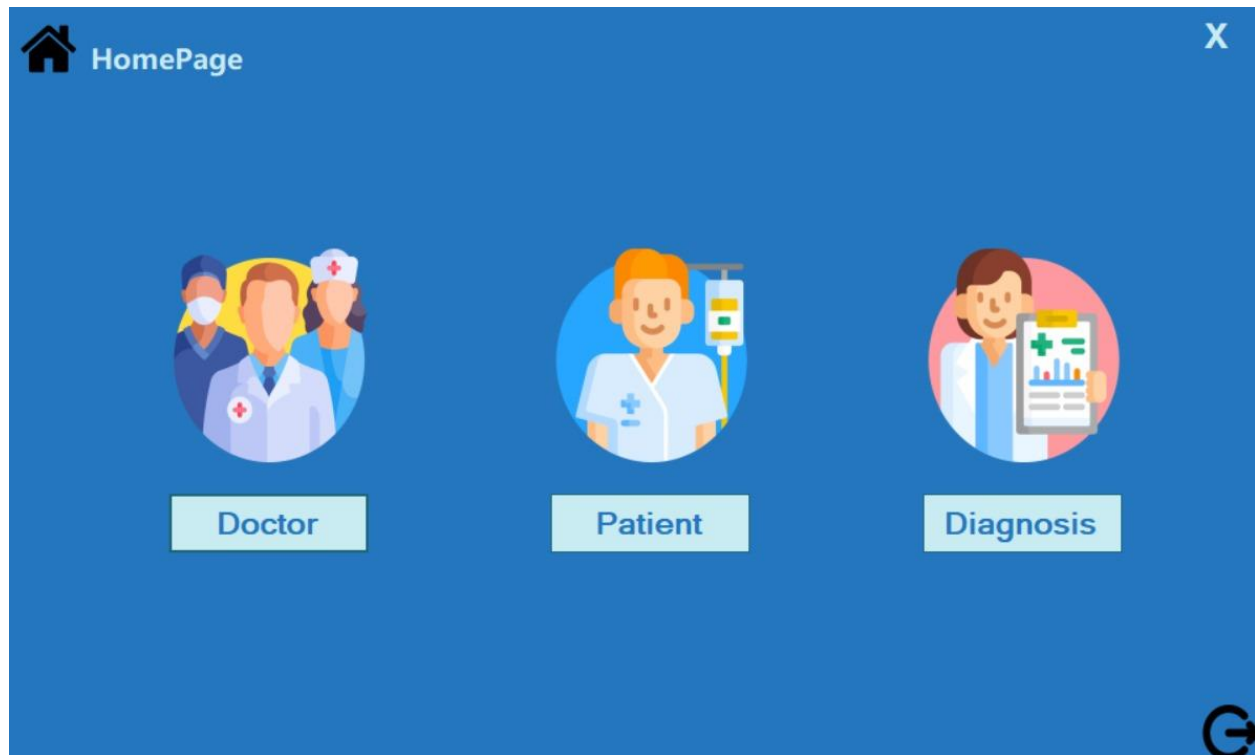


7. Input and Output interface




The image shows a login page for a 'Hospital System'. The page has a solid blue background. At the top, the text 'Hospital System' is written in white, with a white 'X' icon in the top right corner. Below the title is a stylized illustration of a hospital building with a red cross on its roof. Underneath the illustration, the word 'Admin' is written in white. Below 'Admin' is a light blue rectangular input field. Further down, the word 'Password' is written in white, followed by another light blue rectangular input field. At the bottom center, there is a white rectangular button with the word 'Login' written in blue.

Login Page for authenticate 1



Home Page for Transaction between Pages 1


Doctor Page
X

Doctor ID
Doctor Name
Gender
Years Of Exp
Licence Num

	DocId	DocName	DocGen	Experience	Licence
*					

Doctor Page for registration of doctors 1


Patient Page
X

Patient ID
P Phone
Doctor ID


P Name
Gender

P Address
bloodGroup

P Age
Disease

	PId	PName	PAddress	PAge	PPhone	PGen	BloodGroup
▶	1001	sameh	cairo	33	01345234975	Male	B+
*							

Patient Page for registration of patient 1

 **Diagnosis Page** X

Diagnosis ID

Patient Name

Add

Update

Patient ID

Diagnostic Test

Reset

Delete

Doctor ID

Medicines

Home

Symptoms

	Did	PatientId	PatientName	Symptoms	DiagnosticTest	Medicines	DocId
*							

Diagnosis Page for Diagnosis register 1

Add

Update

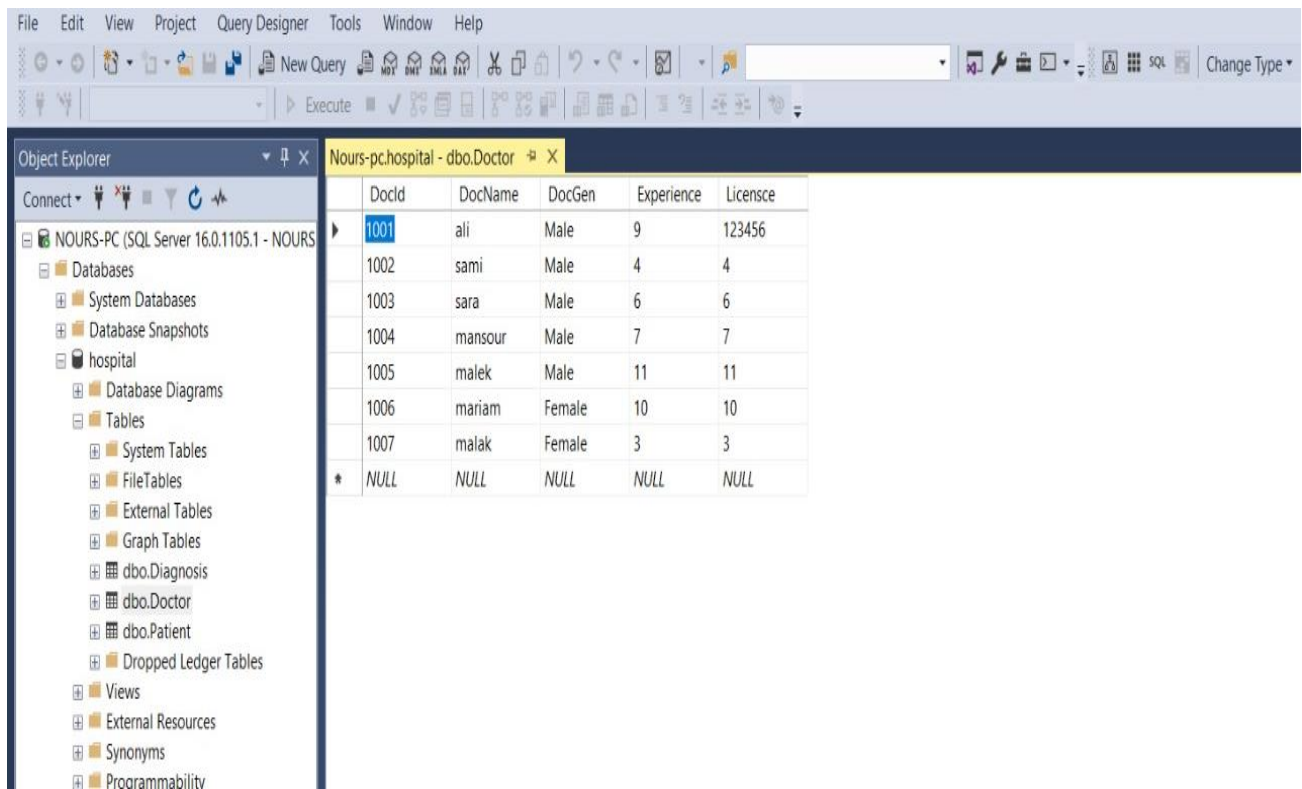
Reset

Delete

Home

Action Bottoms 1

8. Queries

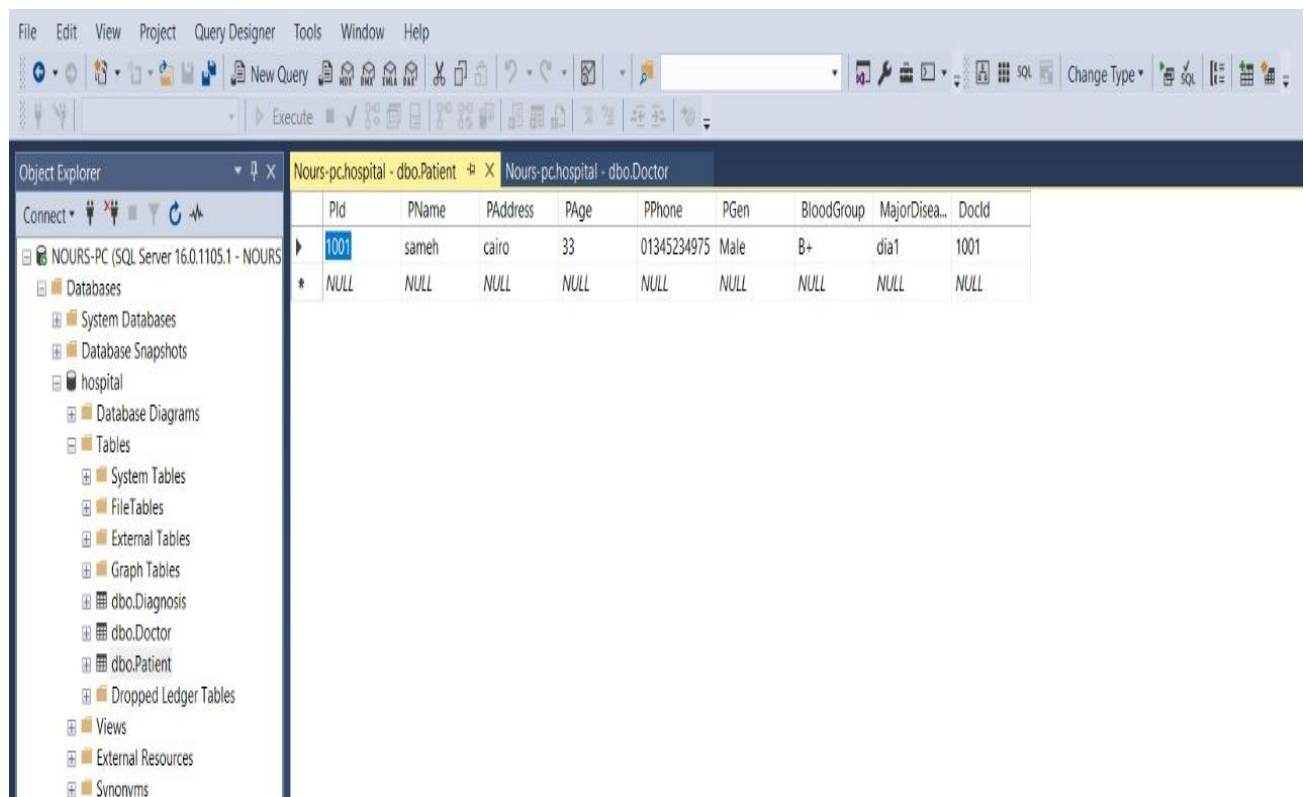


The screenshot displays the Microsoft SQL Server Enterprise Manager interface. The 'Object Explorer' on the left shows the database structure for 'NOURS-PC (SQL Server 16.0.1105.1 - NOURS)'. The 'hospital' database is expanded, showing 'dbo.Doctor' as the selected table. The main pane displays the data for 'Nours-pc.hospital - dbo.Doctor'.

DocId	DocName	DocGen	Experience	License
1001	ali	Male	9	123456
1002	sami	Male	4	4
1003	sara	Male	6	6
1004	mansour	Male	7	7
1005	malek	Male	11	11
1006	mariam	Female	10	10
1007	malak	Female	3	3
* NULL	NULL	NULL	NULL	NULL

Doctors Tables in SQL server 1

Input / Output Interface

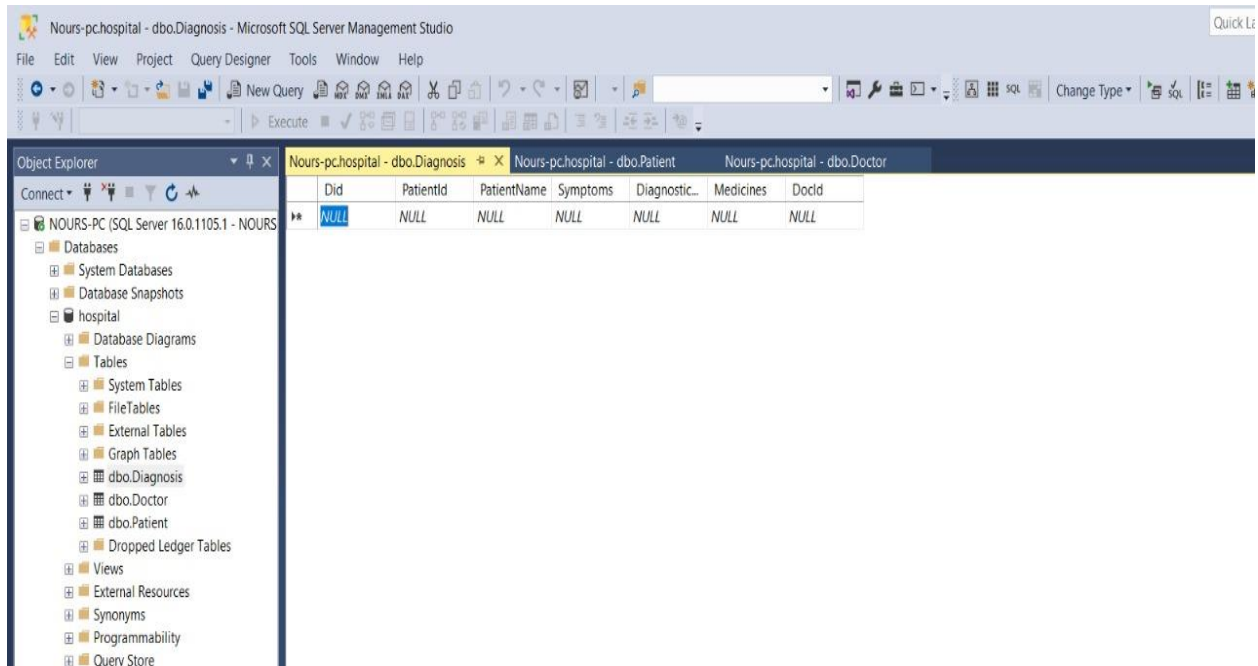


The screenshot displays the Microsoft SQL Server Enterprise Manager interface. The Object Explorer on the left shows the hierarchy: NOURS-PC (SQL Server 16.0.1105.1 - NOURS) > Databases > hospital > Tables > dbo.Patient. The main pane shows the Patient table with the following data:

PId	PName	PAddress	PAge	PPhone	PGen	BloodGroup	MajorDisea...	DocId
1001	sameh	cairo	33	01345234975	Male	B+	dia1	1001
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Patient Table in SQL server 1

Input / Output Interface



Diagnosis Table in SQL server 1 1