Final Project
Using SQL and Python

Project Overview

In this project, you will work with a messy dataset containing patient records from a hospital database. Your task is to clean the data using SQL and Python, analyze it to answer specific medical-related questions, and generate a report summarizing your findings.

Project Steps

1. Data Cleaning and Preparation

You will start by cleaning the dataset using SQL and Python. The dataset contains inconsistencies such as missing values, duplicate records, inconsistent formatting, and mixed data types.

2. Data Exploration and Analysis

After cleaning the data, you will use SQL queries to explore the dataset and answer specific business questions. You will also use Python to perform additional analysis and visualization.

3. Reporting

Finally, you will generate a report summarizing your findings, including key insights, visualizations, and recommendations.

Dataset

The dataset (hospital_patient_records.csv) contains the following columns:

- **PatientID**: Unique identifier for each patient.
- **Name**: Name of the patient.
- **Age**: Patient's age.
- **Gender**: Male or Female.
- **Diagnosis**: Primary diagnosis of the patient.
- **Medication**: Prescribed medication.
- **AdmissionDate**: Date of patient admission.

- **DischargeDate**: Date of discharge.
- **Doctor**: Assigned doctor's name.
- **Department**: Hospital department (e.g., Cardiology, Orthopedics).
- **Status**: Patient status (e.g., Admitted, Discharged, Under Observation).

Dataset Issues

- **Missing Values**: Some rows have missing values in the Age and Diagnosis columns.
- **Inconsistent Formatting**: The AdmissionDate column has dates in different formats (e.g., 2025-01-10, 10/01/2025, January 10, 2025).
- **Mixed Data Types**: The Age column contains both numeric values and text (e.g., 45, forty-five).
- **Duplicate Records**: Some patient records are duplicated.
- **Inconsistent Text**: The Status column has inconsistent capitalization (e.g., Admitted, admitted, ADMITTED).

Project Tasks

Part 1: Data Cleaning (SQL and Python)

1. Load the Dataset

- Load the hospital_patient_records.csv file into a **SQL database** and a **Python environment**

2. Clean the Data

- **SQL Tasks**:
  - Remove duplicate records.
  - Standardize the AdmissionDate column to a consistent format (e.g., YYYY-MM-DD).
  - Convert the Age column to a numeric type.
  - Normalize the Status column to lowercase.

- Replace missing values in the Age and Diagnosis columns with default values (e.g., Unknown).

- **Python Tasks**:
  - Use **Pandas** to clean the dataset further (e.g., remove leading/trailing spaces, handle missing values, and standardize text).

3. Validate the Cleaning Process

- Compare the row counts before and after cleaning.
- Check for remaining missing values and duplicates.
- Verify that all columns have the correct data types.

Part 2: Data Exploration and Analysis (SQL and Python)

1. SQL Queries

Write SQL queries to answer the following questions:

- What is the **average age** of patients for each diagnosis?
- Which department has the **highest number of admitted patients**?
- How many patients have been **discharged** per month?
- What is the **most common diagnosis** among patients?
- Which doctor has treated the **most patients**?

2. Python Analysis

Use Python (Pandas, Matplotlib) to:

- **Visualize the number of patients per department** using a bar chart.
- **Create a pie chart** showing the distribution of patient statuses (Admitted, Discharged, Under Observation).
- **Generate a line chart** showing **monthly hospital admissions trends**.

Part 3: Reporting

1. Generate a Report

- Summarize your findings in a report (e.g., using Jupyter Notebook or a PDF).
- Include:

- **Key insights** from the data analysis.
- **Visualizations** (e.g., charts, graphs).
- **Recommendations** for hospital management (e.g., which departments need more resources, common patient demographics).

Theoretical Questions

1. Data Cleaning

- What are the common issues you might encounter in a messy dataset?
- How would you handle missing values in a dataset?
- What is the importance of data type consistency in data analysis?

2. SQL Queries

- What is the difference between INNER JOIN and LEFT JOIN?
- How would you use the GROUP BY clause to aggregate data?
- What is the purpose of the HAVING clause in SQL?

3. Python Analysis

- How would you use Pandas to clean a dataset with mixed data types?
- What are the benefits of using visualizations in data analysis?

Notes:

1. **Report**: Detailed summary with insights, visualizations, and recommendations.
2. **Answers to Theoretical Questions**: Written responses.