**Name:** Sai Priya Muvvala

**University of North Texas**

**Title of the project:** NextGen

**Project Description:**

NextGen is an electronic health record (EHR) software that delivers better healthcare outcomes for people. EHR is a digitalized form of a patient's medical record. The implementation of electronic health records is one of the greatest landmarks in medical development. It improves health care by reducing medical errors. It is a computerized physician order entry. It provides an effective means of communication between the doctor and the patient by reducing ambiguity. The collected information is also used for education and research purpose. It forms a common platform for the exchange of health information.

NextGen is a health information technology product in medical health records/documents. It replaces the outdated procedure of recording medical records on paper. The primary goals of this project are to develop a system that addresses patient demands and keeps track of treatment regimens. The database gives the user a computerized provider order entry that stores the information electronically, thereby reducing the data's inconsistency. The system also assures the quality and safety of the data. Users can schedule appointments and can also view their reports electronically through this system.

The database system streamlines the workflow and provides automatic and accurate access to information to the user. It also enhances the privacy and security of the collected data and the information collected can be precisely stored and retrieved easily. Thus, this design will serve as an efficient and better healthcare information system than other conventional systems.

**Objectives:**

* The major objective is to provide the clinical decision support
* The database system should improve the efficiency and effectiveness of care
* EHR forms a common platform to share the information required with the doctor
* It should create a database that aids the healthcare professional in making pertinent decisions and therapy suggestions with real-time diagnostics that improve patient care
* The development of a consistent database system will drastically reduce incorrect diagnosis interpretation
* The created database avoids duplicates and unnecessary tests which is the most common issue with conventional paper-based health records
* To mitigate drug interactions and addresses the potential patient issues
* To reduce the associated clinical complexities

**Scope:**

* Electronically stores and retrieves patient data to help with future hospital visits.
* Improves the quality, reduces cost, and saves the time
* Enables quick access to patient records in more coordinated, accurate, up-to-date information and patient care
* It also helps in administrative purposes and health management
* Improves patient care by making effective suggestions for potential treatments. It enhances communication and outreach
* NextGen database reduces medication errors
* It makes the ordering process more efficient and eradicates the missing information from illegible or incomplete orders

**User Requirements:**

* It should create a database that includes all the basic and required particulars such as patient demographics, vital signs, no of visits, patient medical history, medications, immunization dates, billing dates, test results, billing system
* Through a web user interface, the user should have the facility to connect to major pharmacies
* Users can claim insurance and denial management services
* Users can choose the insurance plan and can avail of discounts for the same
* Users can text and email appointment reminders
* Users can review the visit info before checking out for payment
* Users can check the amount and time charged in the Invoice electronically
* First-time users can also choose the relevant insurance plan offered by various companies from the list of insurance companies provided
* Users i.e. the patients can cross-check the entries made regarding their first and last names, phone no, email IDs, and date of birth from patent information
* Users can have the access to choose the doctors who are specialized in the particular domain from Doctor information
* Users can save time by gaining access to software that gives the documents online without any hassle

**Choice of DBMS:** MySQL

**Business rules:**

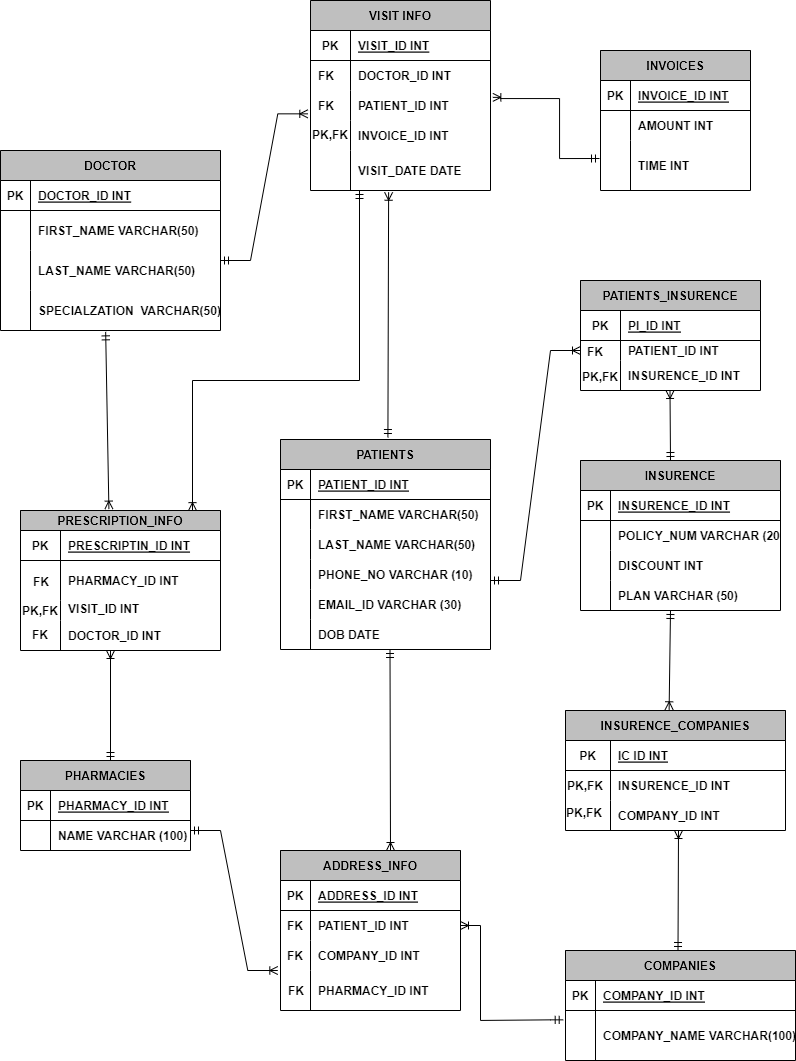
In database design, Business rules specify relation-specific rules in a database design. These comprise the preliminary steps in database design. This set of rules enhances the reliability of the database and establishes hierarchical or dependable relationships between the tables. Specifying the business rules for data plays a crucial step in normalizing the database of the system. The rules must be clear to understand and widely differentiated to interpret the data effectively.

The business rules for the NextGen database are described as follows:

* In the doctor’s table, there is a (1:M) cardinality with visit info and prescription info i.e. A doctor may attend one or more visits and one visit may associate with one doctor
* Each doctor is uniquely identified by their identification number
* The visit ID forms the unique attribute in the visit information table
* A doctor may write many prescriptions and one prescription is written by the respective doctor (1:M) with a doctor to prescription
* One visit of the patient results from one or many prescriptions, hence the relationship between the entities from visit info to prescription becomes (1:M)
* Every pharmacy has its unique ID which prevents the data duplicity
* Users may take all the medicines in the prescription from one pharmacy and one pharmacy may provide medicines from different prescriptions. The relationship between prescription info and to pharmacy thus becomes (M:1)
* The address information of the patient includes the respective pharmacy ID, patient ID, and the insurance companies identification number
* While prescribing the medicines, pharmacists charge based on the entries in the address info which depicts the patient’s insurance claims
* Different company names are represented by their unique company ID which collects the information of all the insurance companies
* The entity insurance includes the attributes such as its unique insurance\_id, policy\_number, discount\_rate, and insurance\_plans from various companies. Hence, one insurance may have many insurance plans from different companies and each individual plan represents one company’s plan.
* The cardinality between insurance to insurance companies is (1:M) whereas companies to insurance table have an (M:1) relation
* Users/ the patient’s insurance table is represented by a unique id and also has the information of the patient and insurance
* The cardinality of patients’ insurance to patients and insurance will be (M:1) respectively
* Patients may have many insurance plans from different companies and one insurance plan is taken by one patient
* The patient entity constitutes all the patient's information such as their first and last name, phone number, email id, and date of birth of the patient.
* It has the patient identification number which is unique and represents the individual’s information securely
* The details of patient information from the patient entity are entered into the visit info which also has doctor and invoice information
* Every Invoice has a unique Invoice id, the amount charged, and the time of the payment of the patient
* The cardinality from invoice to visit info is 1:M relation
* Every patient has one invoice receipt
* Visit info has many invoices from different patients
* Visit info has many items with unique Visit info id
* The cardinality from patients to visit info is 1:M relation
* Patients will have access to review the information entered and also doctors can check their entries and provide appropriate medications through pharmacies the amount charged will also be accurate as per the patient’s claimed insurance plan
* All together this system provides access and accurate information to all three users i.e. doctors, patients, and pharmacists

**Entity Relationship Diagram:**

The Crows foot representation of the system is described as follows which depicts the relevant entities and relationships among the tables:



**Data Dictionary:**

* Data dictionary in database management system plays a fundamental and vital role
* It provides detailed information about all the tables in the system and constitutes a centralized repository of the data
* A data dictionary constitutes characteristics and information of all attributes for each table in the system
* A data dictionary arranges the data in an organized way and provides a piece of precise information about the data
* It also provides detailed information about the relationship between multiple tables in a database
* It assists the database designer in taking relevant decisions about building the data structure
* Thus it aids in preventing data redundancy

The data dictionary for this project, NextGen DBMS is described as follows:



**Data entry and Update:** This project involves 11 following tables:

1. **Doctor:** This table contains basic information about the doctor including his specialization in the domain
2. **Prescription Info:** This is the table that has a unique ID and other information regarding pharmacy, visit ID, Doctor ID
3. **Pharmacies:** This table contains the details of the medicine prescribed with the ID number representing the unique bill
4. **Address info:** This constitutes the details of the patent, company, and also the details of pharmacy
5. **Companies:** This contains details of various companies with their ID
6. **Insurance Companies:** This table contains the Insurance details of all the companies
7. **Insurance:** This table contains details about the patient’s insurance plan, discount availed, and policy number
8. **Claim:** This table contains the details of respective patient-claimed insurance details and their ID.
9. **Visit info:** This is the table that contains the basic information about the Patient, doctor, Invoice, and also Visit date of the patient
10. **Invoices:** This table contains the billing amount of the patient and also the time of payment done
11. **Patients:** This table contains basic information about the patient their particulars including contact number and Email ID

* The following SQL statements are used to generate the required tables:

1. **Creation of the table PATIENTS:**

CREATE TABLE PATIENTS (PATIENT\_ID INT NOT NULL,

FIRST\_NAME VARCHAR (50) NOT NULL,

LAST\_NAME VARCHAR (50) NOT NULL,

PHONE\_NO VARCHAR (10) NOT NULL,

EMAIL\_ID VARCHAR (30) NOT NULL,

DOB\_DATE date NOT NULL,

PRIMARY KEY (PATIENT\_ID));

* The following statements are used to generate 30 records in the respective Table:

INSERT INTO PATIENTS VALUES (01, 'RITHU', 'GUPTHA', '1243212834', 'RITHUGUPTHA@GMAIL.COM', '1997-01-11');

INSERT INTO PATIENTS VALUES (02, 'PRIYA', 'REDDY', '1256212934', 'PRIYAREDDY@GMAIL.COM', '1996-02-01');

INSERT INTO PATIENTS VALUES (03, 'RIYA', 'EDDY', '1256212634', 'RIYAEDDY@GMAIL.COM', '1995-07-05');

INSERT INTO PATIENTS VALUES (04, 'SIYA', 'MARIS', '1353412634', 'SIYAMARIS@GMAIL.COM', '1985-06-04');

INSERT INTO PATIENTS VALUES (05, 'LIYA', 'THOMAS', '4103412634', 'LIYATHOMAS@GMAIL.COM', '1985-07-04');

INSERT INTO PATIENTS VALUES (06, 'OMAN', 'THOMAS', '4107812634', 'OMANTHOMAS@GMAIL.COM', '1985-12-24');

INSERT INTO PATIENTS VALUES (07, 'SOMAN', 'THOMAS', '4103412634', 'SOMANHOMAS@GMAIL.COM', '1995-07-04');

INSERT INTO PATIENTS VALUES (08, 'RAMB', 'PETER', '4106422634', 'RAMBPETER@GMAIL.COM', '1985-08-04');

INSERT INTO PATIENTS VALUES (09, 'RECK', 'PETER', '4106422634', 'RECKPETER@GMAIL.COM', '1995-08-04');

INSERT INTO PATIENTS VALUES (10, 'PHIL', 'ETHTER', '4104822634', 'RPHILETHER@GMAIL.COM', '1995-08-24');

INSERT INTO PATIENTS VALUES (11, 'ALLEN', 'ETHTER', '4104822634', 'ALLENETHER@GMAIL.COM', '1995-09-24');

INSERT INTO PATIENTS VALUES (12, 'PHILY', 'ETHTER', '4104822634', 'PHILYETHER@GMAIL.COM', '1985-09-24');

INSERT INTO PATIENTS VALUES (13, 'MIKE', 'POTER', '2134822634', 'MIKEPOTER@GMAIL.COM', '1995-11-24');

INSERT INTO PATIENTS VALUES (14, 'NIKE', 'POTER', '2137652634', 'NIKEPOTER@GMAIL.COM', '1985-11-24');

INSERT INTO PATIENTS VALUES (15, 'NEHA', 'CHARLES', '2194052634', 'NEHACHARLES@GMAIL.COM', '1995-01-24');

INSERT INTO PATIENTS VALUES (16, 'SNEHA', 'CHARLES', '2184352634', 'SNEHACHARLES@GMAIL.COM', '1995-07-14');

INSERT INTO PATIENTS VALUES (17, 'LEE', 'CHARLES', '2184352634', 'LEECHARLES@GMAIL.COM', '1995-08-24');

INSERT INTO PATIENTS VALUES (18, 'SETH', 'CHRIS', '2184789634', 'SETHCHRIS@GMAIL.COM', '1995-06-14');

INSERT INTO PATIENTS VALUES (19, 'LIKTH', 'CHRIS', '1344789634', 'LIKTHCHRIS@GMAIL.COM', '1997-05-13');

INSERT INTO PATIENTS VALUES (20, 'KATH', 'CHRIS', '2184789674', 'KATHCHRIS@GMAIL.COM', '1995-09-14');

INSERT INTO PATIENTS VALUES (21, 'BETH', 'MARRY', '2184789674', 'BETHMARRY@GMAIL.COM', '1995-09-14');

INSERT INTO PATIENTS VALUES (22, 'HARIS', 'WINSON', '1346829674', 'HARISWINSON@GMAIL.COM', '1995-10-17');

INSERT INTO PATIENTS VALUES (23, 'PHILP', 'WINSON', '1345739674', 'PHILIPWINSON@GMAIL.COM', '1996-11-18');

INSERT INTO PATIENTS VALUES (24, 'CASY', 'WINSON', '1345739174', 'CASYWINSON@GMAIL.COM', '1996-05-18');

INSERT INTO PATIENTS VALUES (25, 'CASY', 'FRANKLIN', '1345739174', 'CASYFRANKLIN@GMAIL.COM', '1996-06-27');

INSERT INTO PATIENTS VALUES (26, 'BAREN', 'MAYOR', '1345739174', 'BARENMAYOR@GMAIL.COM', '1990-06-23');

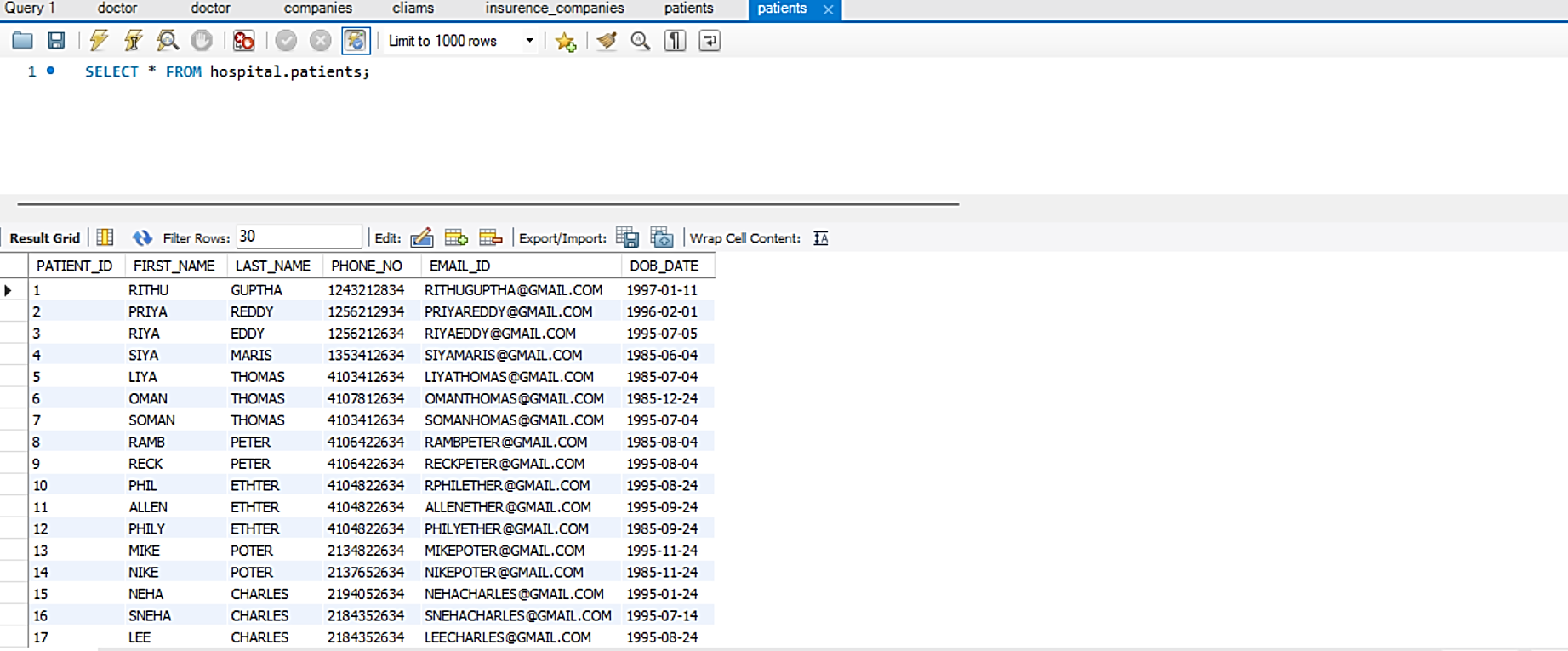
INSERT INTO PATIENTS VALUES (27, 'HARRY', 'MAYOR', '1345739174', 'HARRYMAYOR@GMAIL.COM', '1990-07-13');

INSERT INTO PATIENTS VALUES (28, 'BEHA', 'MARRIA', '1345739174', 'BEHAMARRIS@GMAIL.COM', '1995-08-15');

INSERT INTO PATIENTS VALUES (29, 'LISA', 'HALEN', '2345739174', 'LISAHALEN@GMAIL.COM', '1990-07-23');

INSERT INTO PATIENTS VALUES (30, 'ZOYA', 'LEON', '9405673910', 'ZOYALEON@GMAIL.COM', '1995-08-06');

SELECT \* FROM PATIENTS;



1. **Creation of the table DOCTORS:**

CREATE TABLE DOCTORS (DOCTOR\_ID INT NOT NULL,

FIRST\_NAME VARCHAR (50) NOT NULL,

LAST\_NAME VARCHAR (50) NOT NULL,

SPECIALIZATION VARCHAR (50) NOT NULL,

PRIMARY KEY (DOCTOR\_ID));

* The following statements are used to generate 30 records in the respective Table:

INSERT INTO DOCTOR VALUES (1, 'REK', 'HELEN', 'CARDIOLOGIST');

INSERT INTO DOCTOR VALUES (2, 'MIKE', 'HARRY', 'CARDIOLOGIST');

INSERT INTO DOCTOR VALUES (3, 'JENNY', 'NAYLOR', 'CARDIOLOGIST');

INSERT INTO DOCTOR VALUES (4, 'TERDY', 'NAYLOR', 'CARDIOLOGIST');

INSERT INTO DOCTOR VALUES (5, 'KATE', 'BITH', 'CARDIOLOGIST');

INSERT INTO DOCTOR VALUES (6, 'ZETH', 'MARRY', 'CARDIOLOGIST');

INSERT INTO DOCTOR VALUES (7, 'ACE', 'BISTH', 'CARDIOLOGIST');

INSERT INTO DOCTOR VALUES (8, 'ZEN', 'KELER', 'CARDIOLOGIST');

INSERT INTO DOCTOR VALUES (9, 'WIKI', 'LAWRENCE', 'CARDIOLOGIST');

INSERT INTO DOCTOR VALUES (10, 'NANCY', 'WILLIAMS', 'CARDIOLOGIST');

INSERT INTO DOCTOR VALUES (11, 'MARRY', 'JANSON', 'PHYSICIAN');

INSERT INTO DOCTOR VALUES (12, 'ZELEN', 'JANSON', 'PHYSICIAN');

INSERT INTO DOCTOR VALUES (13, 'SAM', 'KELLY', 'PHYSICIAN');

INSERT INTO DOCTOR VALUES (14, 'MARRY', 'JANSON', 'PHYSICIAN');

INSERT INTO DOCTOR VALUES (15, 'QUILL', 'PATRICK', 'PHYSICIAN');

INSERT INTO DOCTOR VALUES (16, 'SASSY', 'PATRICK', 'PHYSICIAN');

INSERT INTO DOCTOR VALUES (17, 'SALLY', 'JANSON', 'DERMATOLOGIST');

INSERT INTO DOCTOR VALUES (18, 'CATHY', 'WILLIAMS', 'DERMATOLOGIST');

INSERT INTO DOCTOR VALUES (19, 'SAM', 'WILLIAMS', 'DERMATOLOGIST');

INSERT INTO DOCTOR VALUES (20, 'WESY', 'HARISON', 'DERMATOLOGIST');

INSERT INTO DOCTOR VALUES (21, 'PUTHIN', 'JANSON', 'DERMATOLOGIST');

INSERT INTO DOCTOR VALUES (22, 'NITHIN', 'SHEPORD', 'DERMATOLOGIST');

INSERT INTO DOCTOR VALUES (23, 'SASSY', 'KUMAR', 'PULMONOLOGIST');

INSERT INTO DOCTOR VALUES (24, 'LALITH', 'ENAMELA', 'PULMONOLOGIST');

INSERT INTO DOCTOR VALUES (25, 'MAYKER', 'FELIS', 'PULMONOLOGIST');

INSERT INTO DOCTOR VALUES (26, 'PETER', 'HARRIS', 'PULMONOLOGIST');

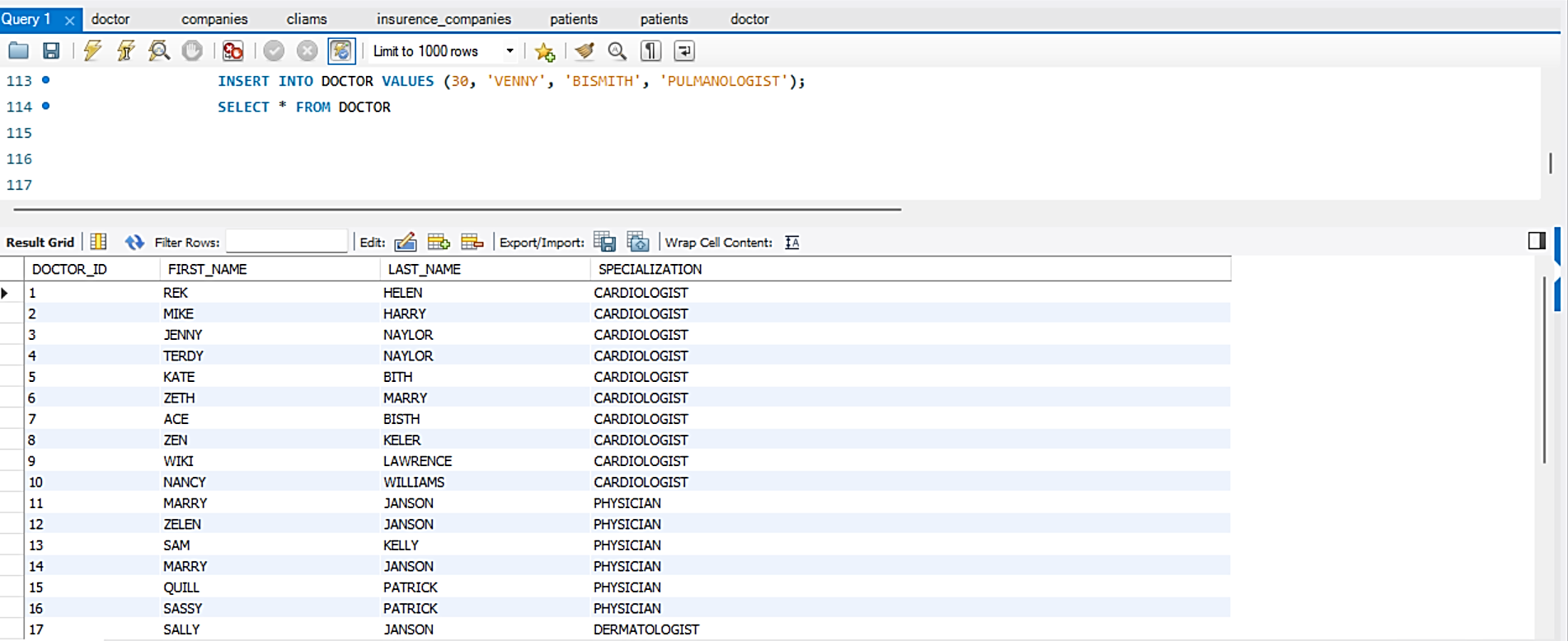
INSERT INTO DOCTOR VALUES (27, 'HENRY', 'HARRIS', 'PULMONOLOGIST');

INSERT INTO DOCTOR VALUES (28, 'AFROZ', 'ZAMER', 'ONCOLOGIST');

INSERT INTO DOCTOR VALUES (29, 'ZELLEN', 'BROTH', 'PULMANOLOGIST');

INSERT INTO DOCTOR VALUES (30, 'VENNY', 'BISMITH', 'PULMANOLOGIST');

SELECT \* FROM DOCTOR



1. **Creation of the table Pharmacies:**

CREATE TABLE PHARMACIES (PHARMACY\_ID INT NOT NULL,

NAME VARCHAR (100) NOT NULL,

PRIMARY KEY (PHARMACY\_ID));

* The following statements are used to generate 30 records in the respective Table:

INSERT INTO PHARMACIES VALUES (001, 'ATENLOL');

INSERT INTO PHARMACIES VALUES (002, 'IBUPROFEN');

INSERT INTO PHARMACIES VALUES (003, 'AMOXICILLIN');

INSERT INTO PHARMACIES VALUES (004, 'ALBUTEROL');

INSERT INTO PHARMACIES VALUES (005, 'CLONAZEPAM');

INSERT INTO PHARMACIES VALUES (006, 'CLONIDINE');

INSERT INTO PHARMACIES VALUES (007, 'BEZOCAINE');

INSERT INTO PHARMACIES VALUES (008, 'CELECOXIB');

INSERT INTO PHARMACIES VALUES (009, 'AZITHROMYCIN');

INSERT INTO PHARMACIES VALUES (010, 'DIAZEPAM');

INSERT INTO PHARMACIES VALUES (011, 'DICLOFENAC');

INSERT INTO PHARMACIES VALUES (012, 'ENLAPRIL');

INSERT INTO PHARMACIES VALUES (013, 'EZETAMIBE');

INSERT INTO PHARMACIES VALUES (014, 'FLUCANAZOLE');

INSERT INTO PHARMACIES VALUES (015, 'ITRACANAZOLE');

INSERT INTO PHARMACIES VALUES (016, 'FOLICACID');

INSERT INTO PHARMACIES VALUES (017, 'FUROSIMIDE');

INSERT INTO PHARMACIES VALUES (018, 'GABAPENTIN');

INSERT INTO PHARMACIES VALUES (019, 'GLYBURIDE');

INSERT INTO PHARMACIES VALUES (020, 'GLIPIZIDE');

INSERT INTO PHARMACIES VALUES (021, 'LANTUZ');

INSERT INTO PHARMACIES VALUES (022, 'LANZOPRAZOLE');

INSERT INTO PHARMACIES VALUES (023, 'LISINOPRIL');

INSERT INTO PHARMACIES VALUES (024, 'LOSARTAN');

INSERT INTO PHARMACIES VALUES (025, 'LEVOSTATIN');

INSERT INTO PHARMACIES VALUES (026, 'METFORMIN');

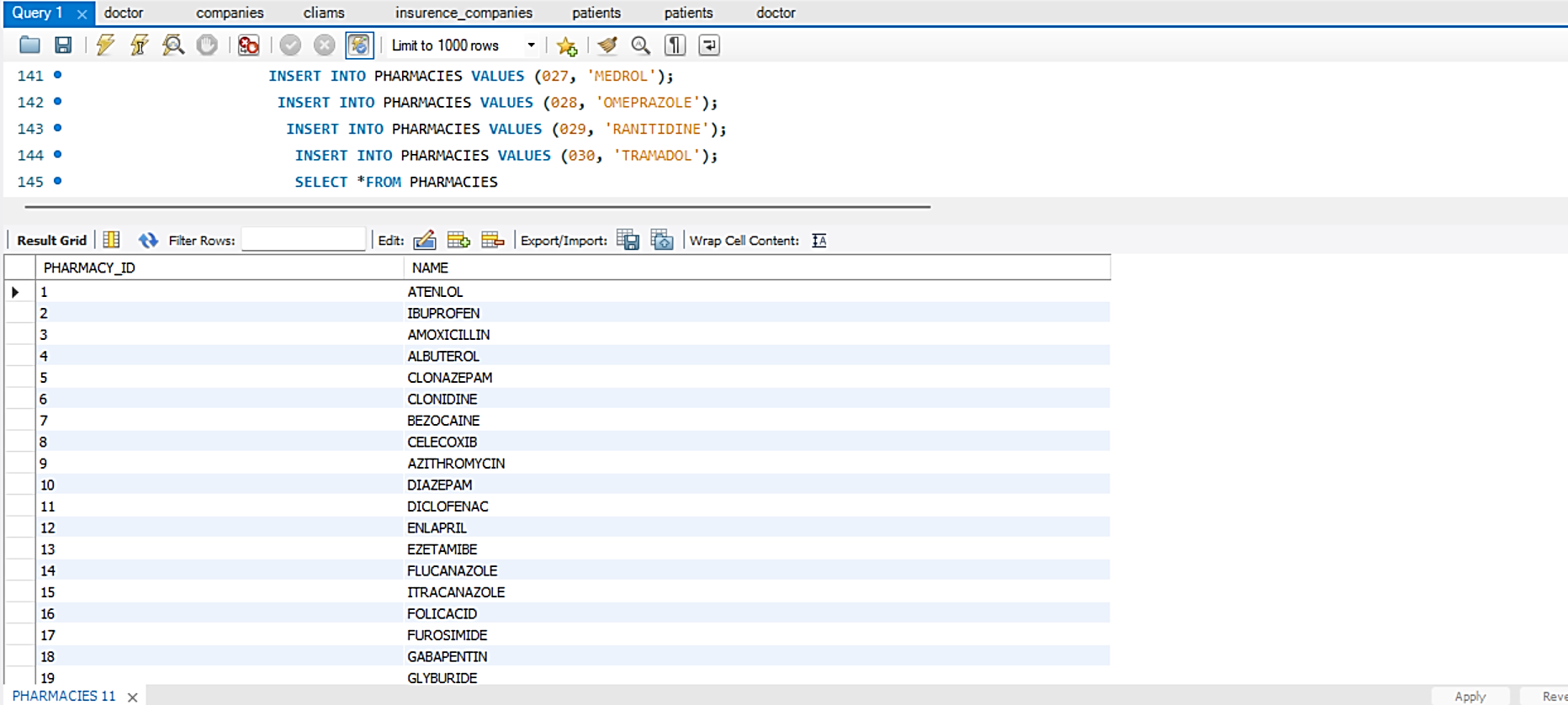
INSERT INTO PHARMACIES VALUES (027, 'MEDROL');

INSERT INTO PHARMACIES VALUES (028, 'OMEPRAZOLE');

INSERT INTO PHARMACIES VALUES (029, 'RANITIDINE');

INSERT INTO PHARMACIES VALUES (030, 'TRAMADOL');

SELECT \*FROM PHARMACIES



1. **Creation of the table INVOICES:**

CREATE TABLE INVOICES (INVOICE\_ID INT NOT NULL,

AMOUNT\_ID INT NOT NULL,

PRIMARY KEY (INVOICE\_ID));

* The following statements are used to generate 30 records in the respective Table:

INSERT INTO INVOICES VALUES (1, '534', '2022-02-11 12:10:45');

INSERT INTO INVOICES VALUES (2, '314', '2022-02-15 01:10:45');

INSERT INTO INVOICES VALUES (3, '566', '2021-01-25 10:10:45');

INSERT INTO INVOICES VALUES (4, '788', '2021-12-15 11:10:45');

INSERT INTO INVOICES VALUES (5, '794', '2022-03-15 07:10:15');

INSERT INTO INVOICES VALUES (6, '988', '2022-01-06 10:10:15');

INSERT INTO INVOICES VALUES (7, '750', '2022-03-22 12:10:35');

INSERT INTO INVOICES VALUES (8, '1080', '2021-04-07 15:07:45');

INSERT INTO INVOICES VALUES (9, '1100', '2022-12-06 12:11:25');

INSERT INTO INVOICES VALUES (10, '988', '2022-03-16 01:11:45');

INSERT INTO INVOICES VALUES (11, '898', '2022-02-07 11:10:10');

INSERT INTO INVOICES VALUES (12, '298', '2022-02-08 11:11:11');

INSERT INTO INVOICES VALUES (13, '5456', '2022-02-12 21:11:11');

INSERT INTO INVOICES VALUES (14, '198', '2022-03-23 01:17:31');

INSERT INTO INVOICES VALUES (15, '568', '2022-12-18 00:11:11');

INSERT INTO INVOICES VALUES (16, '567', '2021-03-28 00:00:00');

INSERT INTO INVOICES VALUES (17, '128', '2022-03-09 21:01:11');

INSERT INTO INVOICES VALUES (18, '1148', '2022-03-08 12:01:11');

INSERT INTO INVOICES VALUES (19, '2398', '2022-08-08 11:18:11');

INSERT INTO INVOICES VALUES (20, '3268', '2022-03-18 10:11:11');

INSERT INTO INVOICES VALUES (21, '368', '2022-04-18 10:11:11');

INSERT INTO INVOICES VALUES (22, '858', '2022-03-19 10:12:11');

INSERT INTO INVOICES VALUES (23, '3168', '2022-09-18 10:11:11');

INSERT INTO INVOICES VALUES (24, '5448', '2021-03-18 10:11:45');

INSERT INTO INVOICES VALUES (25, '3448', '2021-03-18 10:11:33');

INSERT INTO INVOICES VALUES (26, '456', '2021-07-28 10:00:11');

INSERT INTO INVOICES VALUES (27, '8998', '2021-03-13 06:21:32');

INSERT INTO INVOICES VALUES (28, '5432', '2022-06-18 00:11:11');

INSERT INTO INVOICES VALUES (29, '3221', '2021-04-23 10:34:11');

INSERT INTO INVOICES VALUES (30, '3268', '2021-10-18 00:00:00');

SELECT \* FROM INVOICES



1. **Creation of the table COMPANIES:**

CREATE TABLE COMPANIES (COMPANY\_ID INT NOT NULL,

COMPANY\_NAME VARCHAR(100) NOT NULL,

PRIMARY KEY (COMPANY\_ID));

* The following statements are used to generate 30 records in the respective Table:

INSERT INTO COMPANIES VALUES (1,'CAPITAL RX');

INSERT INTO COMPANIES VALUES (2,'TOP VENDOR');

INSERT INTO COMPANIES VALUES (3,'MEGALLEN RX');

INSERT INTO COMPANIES VALUES (4,'OPTUM');

INSERT INTO COMPANIES VALUES (5,'AMGEN');

INSERT INTO COMPANIES VALUES (6,'MAGNUS');

INSERT INTO COMPANIES VALUES (7,'MEDIMPACT');

INSERT INTO COMPANIES VALUES (8,'PHARMA RX');

INSERT INTO COMPANIES VALUES (9,'EXPRESS RX');

INSERT INTO COMPANIES VALUES (10,'EXPRESS SCRIPTS');

INSERT INTO COMPANIES VALUES (11,'CVS HEALTH');

INSERT INTO COMPANIES VALUES (12,'INGENIO');

INSERT INTO COMPANIES VALUES (13,'COMPEX');

INSERT INTO COMPANIES VALUES (14,'WITHME HEALTH');

INSERT INTO COMPANIES VALUES (15,'MPATH');

INSERT INTO COMPANIES VALUES (16,'COMPARE');

INSERT INTO COMPANIES VALUES (17,'NAVITUS');

INSERT INTO COMPANIES VALUES (18,'ALIGNS');

INSERT INTO COMPANIES VALUES (19,'PHARMA BENIFIT');

INSERT INTO COMPANIES VALUES (20,'EMPIRE');

INSERT INTO COMPANIES VALUES (21,'RIGTHWAY');

INSERT INTO COMPANIES VALUES (22,'MED ONE');

INSERT INTO COMPANIES VALUES (23,'KROGER');

INSERT INTO COMPANIES VALUES (24,'CLINICIAN RX');

INSERT INTO COMPANIES VALUES (25,'PHARMAPIX');

INSERT INTO COMPANIES VALUES (26,'IPM');

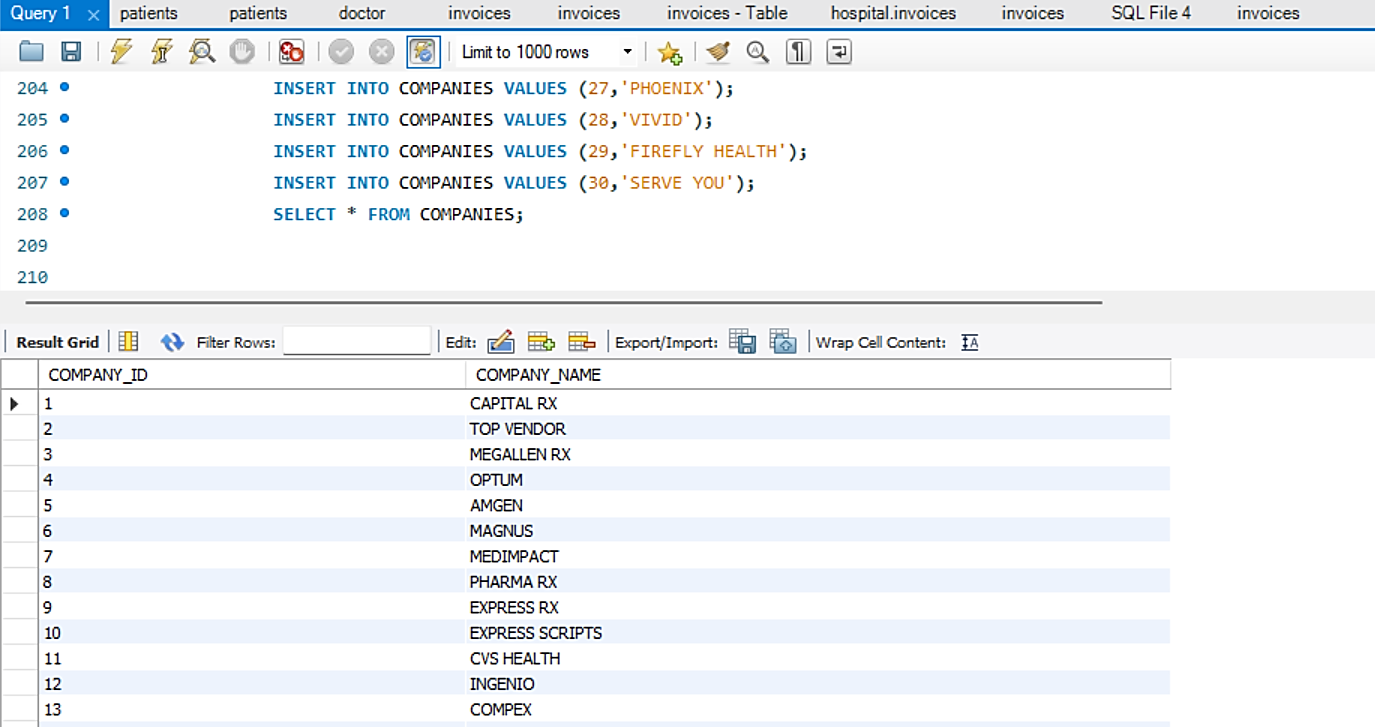
INSERT INTO COMPANIES VALUES (27,'PHOENIX');

INSERT INTO COMPANIES VALUES (28,'VIVID');

INSERT INTO COMPANIES VALUES (29,'FIREFLY HEALTH');

INSERT INTO COMPANIES VALUES (30,'SERVE YOU');

SELECT \* FROM COMPANIES;



1. **Creation of the table INSURANCE:**

CREATE TABLE INSURANCE (INSURENCE\_ID INT NOT NULL,

POLICY\_NUM VARCHAR (20) NOT NULL,

DISCOUNT INT NOT NULL,

PLAN VARCHAR (50) NOT NULL,

PRIMARY KEY (INSURENCE\_ID));

* The following statements are used to generate 30 records in the respective Table:

INSERT INTO INSURENCE VALUES (1, 21, 30, 'ANNUAL');

INSERT INTO INSURENCE VALUES (2, 33, 10, 'ANNUAL');

INSERT INTO INSURENCE VALUES (3, 6, 30, 'ANNUAL');

INSERT INTO INSURENCE VALUES (4, 2, 40, 'ANNUAL');

INSERT INTO INSURENCE VALUES (5, 12, 10, 'ANNUAL');

INSERT INTO INSURENCE VALUES (6, 31, 50, 'ANNUAL');

INSERT INTO INSURENCE VALUES (7, 8, 20, 'ANNUAL');

INSERT INTO INSURENCE VALUES (8, 8, 30, 'ANNUAL');

INSERT INTO INSURENCE VALUES (9, 3, 40, 'ANNUAL');

INSERT INTO INSURENCE VALUES (10, 41, 10, 'ANNUAL');

INSERT INTO INSURENCE VALUES (11, 12, 50, 'BINNUAL');

INSERT INTO INSURENCE VALUES (12, 4, 50, 'BINNUAL');

INSERT INTO INSURENCE VALUES (13, 5, 50, 'BINNUAL');

INSERT INTO INSURENCE VALUES (14, 6, 50, 'BINNUAL');

INSERT INTO INSURENCE VALUES (15, 7, 50, 'BINNUAL');

INSERT INTO INSURENCE VALUES (16, 8, 50, 'BINNUAL');

INSERT INTO INSURENCE VALUES (17, 2, 50, 'BINNUAL');

INSERT INTO INSURENCE VALUES (18, 5, 50, 'BINNUAL');

INSERT INTO INSURENCE VALUES (19, 7, 50, 'BINNUAL');

INSERT INTO INSURENCE VALUES (20, 8, 50, 'BINNUAL');

INSERT INTO INSURENCE VALUES (21, 9, 50, 'BINNUAL');

INSERT INTO INSURENCE VALUES (22, 7, 20, 'ANNUAL');

INSERT INTO INSURENCE VALUES (23, 6, 30, 'BINNUAL');

INSERT INTO INSURENCE VALUES (24, 5, 40, 'BINNUAL');

INSERT INTO INSURENCE VALUES (25, 4, 10, 'ANNUAL');

INSERT INTO INSURENCE VALUES (26, 3, 25, 'ANNUAL');

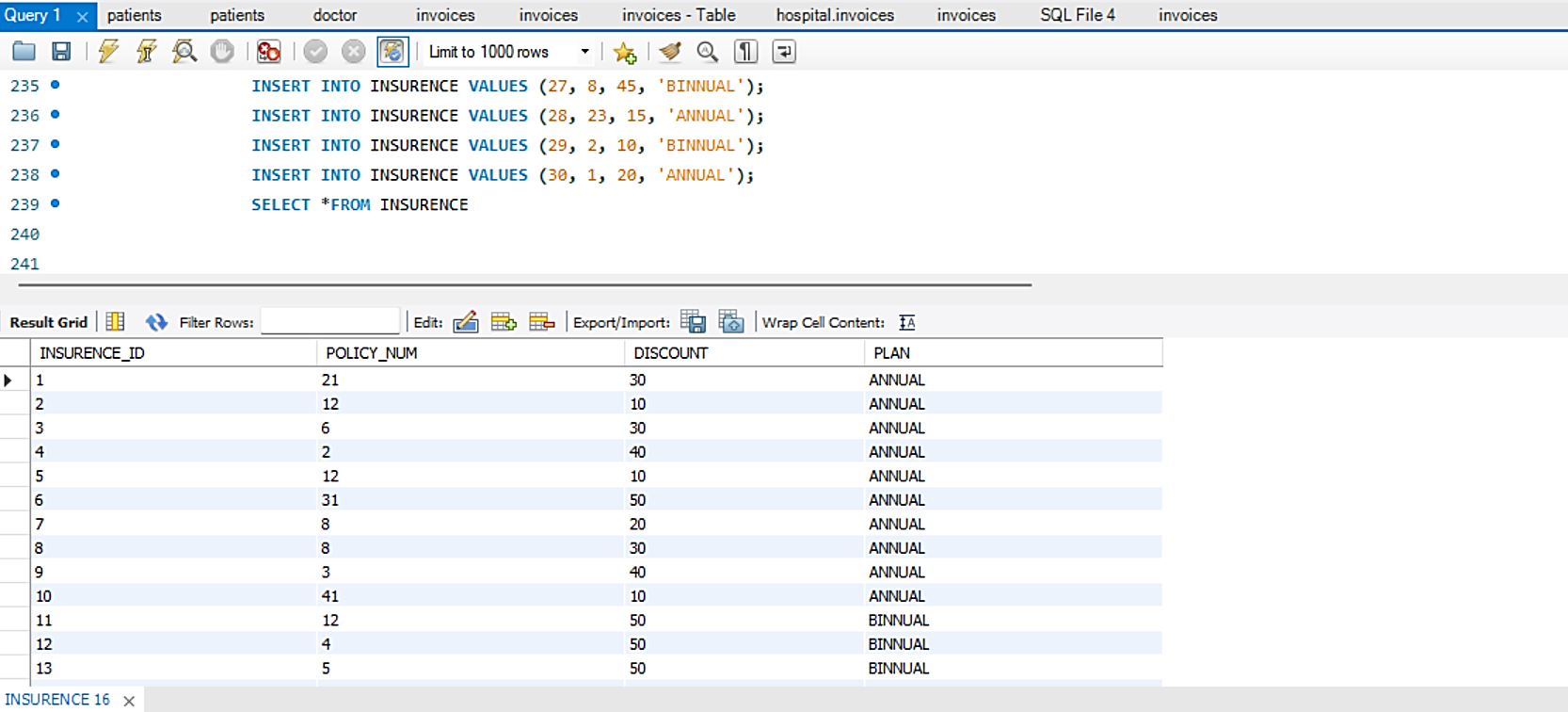
INSERT INTO INSURENCE VALUES (27, 8, 45, 'BINNUAL');

INSERT INTO INSURENCE VALUES (28, 23, 15, 'ANNUAL');

INSERT INTO INSURENCE VALUES (29, 2, 10, 'BINNUAL');

INSERT INTO INSURENCE VALUES (30, 1, 20, 'ANNUAL');

SELECT \*FROM INSURANCE



1. **Creation of the table ADDRESS\_INFO:**

CREATE TABLE ADDRESS\_INFO (ADDRESS\_ID INT NOT NULL,

PATIENT\_ID INT NOT NULL,

COMPANY\_ID INT NOT NULL,

PHARMACY\_ID INT NOT NULL,

PRIMARY KEY (ADDRESS\_ID),

FOREIGN KEY PATIENT(PATIENT\_ID) REFERENCES PATIENTS(PATIENT\_ID),

FOREIGN KEY COMAPNY(COMPANY\_ID) REFERENCES COMPANIES(COMPANY\_ID),

FOREIGN KEY PHARMACY(PHARMACY\_ID) REFERENCES PHARMACIES(PHARMACY\_ID));

* The following statements are used to generate 30 records in the respective Table:

INSERT INTO ADDRESS\_INFO VALUES (7, 01, 1, 001);

INSERT INTO ADDRESS\_INFO VALUES (2, 02, 2, 002);

INSERT INTO ADDRESS\_INFO VALUES (3, 03, 3, 003);

INSERT INTO ADDRESS\_INFO VALUES (4, 04, 4, 004);

INSERT INTO ADDRESS\_INFO VALUES (5, 05, 5, 005);

INSERT INTO ADDRESS\_INFO VALUES (6, 06, 6, 006);

INSERT INTO ADDRESS\_INFO VALUES (33, 07, 7, 007);

INSERT INTO ADDRESS\_INFO VALUES (8, 08, 8, 008);

INSERT INTO ADDRESS\_INFO VALUES (9, 09, 9, 009);

INSERT INTO ADDRESS\_INFO VALUES (10, 10, 10, 010);

INSERT INTO ADDRESS\_INFO VALUES (11, 11, 11, 011);

INSERT INTO ADDRESS\_INFO VALUES (13, 13, 13, 013);

INSERT INTO ADDRESS\_INFO VALUES (14, 14, 14, 014);

INSERT INTO ADDRESS\_INFO VALUES (15, 15, 15, 015);

INSERT INTO ADDRESS\_INFO VALUES (16, 16, 16, 016);

INSERT INTO ADDRESS\_INFO VALUES (17, 17, 17, 017);

INSERT INTO ADDRESS\_INFO VALUES (18, 18, 18, 018);

INSERT INTO ADDRESS\_INFO VALUES (19, 19, 19, 019);

INSERT INTO ADDRESS\_INFO VALUES (20, 20, 20, 020);

INSERT INTO ADDRESS\_INFO VALUES (21, 21, 21, 021);

INSERT INTO ADDRESS\_INFO VALUES (22, 22, 22, 022);

INSERT INTO ADDRESS\_INFO VALUES (23, 23, 23, 023);

INSERT INTO ADDRESS\_INFO VALUES (24, 24, 24, 024);

INSERT INTO ADDRESS\_INFO VALUES (25, 25, 25, 025);

INSERT INTO ADDRESS\_INFO VALUES (26, 26, 26, 026);

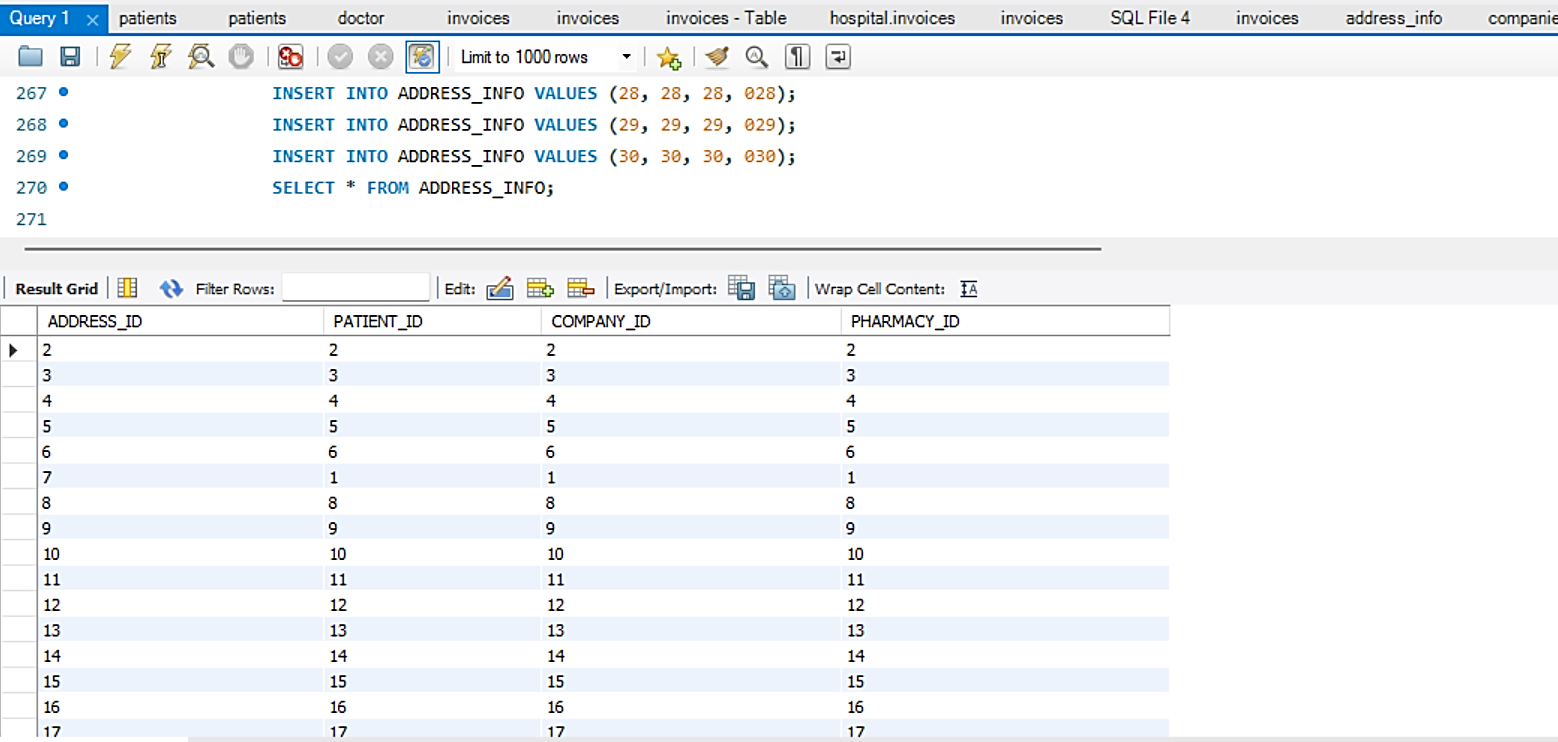
INSERT INTO ADDRESS\_INFO VALUES (27, 27, 27, 027);

INSERT INTO ADDRESS\_INFO VALUES (28, 28, 28, 028);

INSERT INTO ADDRESS\_INFO VALUES (29, 29, 29, 029);

INSERT INTO ADDRESS\_INFO VALUES (30, 30, 30, 030);

SELECT \* FROM ADDRESS\_INFO;



1. **Creation of the table VISIT\_INFO:**

CREATE TABLE VISIT\_INFO (VISIT\_ID INT NOT NULL,

DOCTOR\_ID INT NOT NULL,

PATIENT\_ID INT NOT NULL,

INVOICE\_ID INT NOT NULL,

VISIT\_DATE DATE NOT NULL,

PRIMARY KEY (VISIT\_ID),

FOREIGN KEY DOCTOR(DOCTOR\_ID) REFERENCES DOCTOR(DOCTOR\_ID),

FOREIGN KEY PATIENT(PATIENT\_ID) REFERENCES PATIENTS(PATIENT\_ID),

FOREIGN KEY INVOICE(INVOICE\_ID) REFERENCES INVOICES(INVOICE\_ID));

* The following statements are used to generate 30 records in the respective Table:

INSERT INTO VISIT\_INFO VALUES (1, 1, 01, 1, '2022-02-07');

INSERT INTO VISIT\_INFO VALUES (2, 2, 02, 2, '2022-12-08');

INSERT INTO VISIT\_INFO VALUES (3, 3, 03, 3, '2021-01-30');

INSERT INTO VISIT\_INFO VALUES (4, 4, 04, 4, '2021-12-30');

INSERT INTO VISIT\_INFO VALUES (5, 5, 05, 5, '2022-03-20');

INSERT INTO VISIT\_INFO VALUES (6, 6, 06, 6, '2022-01-07');

INSERT INTO VISIT\_INFO VALUES (7, 7, 07, 7, '2022-03-27');

INSERT INTO VISIT\_INFO VALUES (8, 8, 08, 8, '2021-01-01');

INSERT INTO VISIT\_INFO VALUES (9, 9, 09, 9, '2022-03-07');

INSERT INTO VISIT\_INFO VALUES (10, 10, 10, 10, '2022-05-17');

INSERT INTO VISIT\_INFO VALUES (11, 11, 11, 11, '2022-02-23');

INSERT INTO VISIT\_INFO VALUES (12, 12, 12, 12, '2022-03-13');

INSERT INTO VISIT\_INFO VALUES (13, 13, 13, 13, '2022-04-24');

INSERT INTO VISIT\_INFO VALUES (14, 14, 14, 14, '2022-05-25');

INSERT INTO VISIT\_INFO VALUES (15, 15, 15, 15, '2022-06-26');

INSERT INTO VISIT\_INFO VALUES (16, 16, 16, 16, '2022-12-23');

INSERT INTO VISIT\_INFO VALUES (17, 17, 17, 17, '2021-03-07');

INSERT INTO VISIT\_INFO VALUES (18, 18, 18, 18, '2021-05-13');

INSERT INTO VISIT\_INFO VALUES (19, 19, 19, 19, '2021-01-13');

INSERT INTO VISIT\_INFO VALUES (20, 20, 20, 20, '2022-02-23');

INSERT INTO VISIT\_INFO VALUES (21, 21, 21, 21, '2022-02-23');

INSERT INTO VISIT\_INFO VALUES (22, 22, 22, 22, '2021-05-13');

INSERT INTO VISIT\_INFO VALUES (23, 23, 23, 23, '2021-07-03');

INSERT INTO VISIT\_INFO VALUES (24, 24, 24, 24, '2021-08-26');

INSERT INTO VISIT\_INFO VALUES (25, 25, 25, 25, '2022-01-07');

INSERT INTO VISIT\_INFO VALUES (26, 26, 26, 26, '2022-07-18');

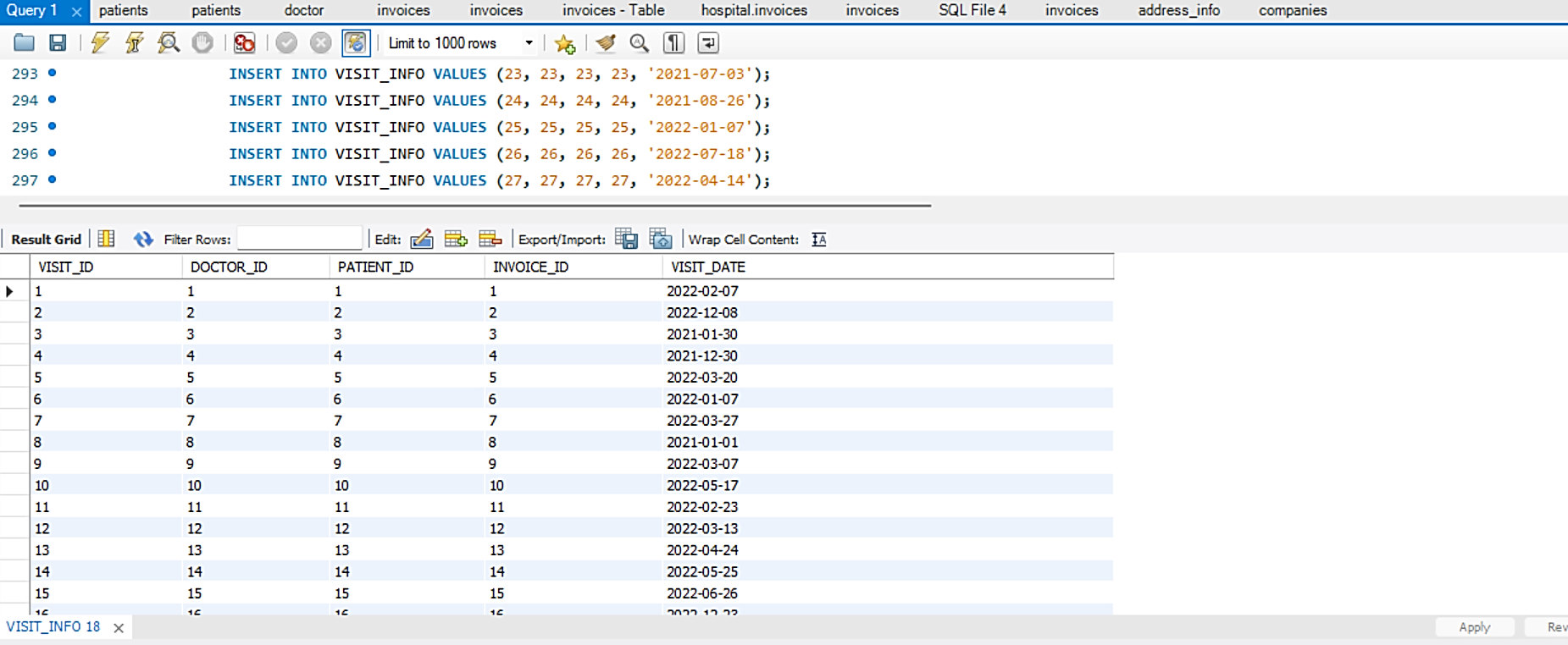
INSERT INTO VISIT\_INFO VALUES (27, 27, 27, 27, '2022-04-14');

INSERT INTO VISIT\_INFO VALUES (28, 28, 28, 28, '2021-08-12');

INSERT INTO VISIT\_INFO VALUES (29, 29, 29, 29, '2022-06-13');

INSERT INTO VISIT\_INFO VALUES (30, 30, 30, 30, '2022-06-15');

SELECT \* FROM VISIT\_INFO;



1. **Creation of the table PRESCRIPTION\_INFO:**

CREATE TABLE PRESCRIPTION\_INFO (PRESCRIPTION\_ID INT NOT NULL,

PHARMACY\_ID INT NOT NULL,

VISIT\_ID INT NOT NULL,

DOCTOR\_ID INT NOT NULL,

PRIMARY KEY (PRESCRIPTION\_ID),

FOREIGN KEY PHARMACY(PHARMACY\_ID) REFERENCES PHARMACIES(PHARMACY\_ID),

FOREIGN KEY VISIT(VISIT\_ID) REFERENCES VISIT\_INFO(VISIT\_ID),

FOREIGN KEY DOCTOR(DOCTOR\_ID) REFERENCES DOCTOR(DOCTOR\_ID));

* The following statements are used to generate 30 records in the respective Table:

INSERT INTO PRESCRIPTION\_INFO VALUES (1, 001, 1, 1);

INSERT INTO PRESCRIPTION\_INFO VALUES (2, 002, 2, 2);

INSERT INTO PRESCRIPTION\_INFO VALUES (3, 003, 3, 3);

INSERT INTO PRESCRIPTION\_INFO VALUES (4, 004, 4, 4);

INSERT INTO PRESCRIPTION\_INFO VALUES (5, 005, 5, 5);

INSERT INTO PRESCRIPTION\_INFO VALUES (6, 006, 6, 6);

INSERT INTO PRESCRIPTION\_INFO VALUES (7, 007, 7, 7);

INSERT INTO PRESCRIPTION\_INFO VALUES (8, 008, 8, 8);

INSERT INTO PRESCRIPTION\_INFO VALUES (9, 009, 9, 9);

INSERT INTO PRESCRIPTION\_INFO VALUES (10, 010, 10, 10);

INSERT INTO PRESCRIPTION\_INFO VALUES (11, 011, 11, 11);

INSERT INTO PRESCRIPTION\_INFO VALUES (12, 012, 12, 12);

INSERT INTO PRESCRIPTION\_INFO VALUES (13, 013, 13, 13);

INSERT INTO PRESCRIPTION\_INFO VALUES (14, 014, 14, 14);

INSERT INTO PRESCRIPTION\_INFO VALUES (15, 015, 15, 15);

INSERT INTO PRESCRIPTION\_INFO VALUES (16, 016, 16, 16);

INSERT INTO PRESCRIPTION\_INFO VALUES (17, 017, 17, 17);

INSERT INTO PRESCRIPTION\_INFO VALUES (18, 018, 18, 18);

INSERT INTO PRESCRIPTION\_INFO VALUES (19, 019, 19, 19);

INSERT INTO PRESCRIPTION\_INFO VALUES (20, 020, 20, 20);

INSERT INTO PRESCRIPTION\_INFO VALUES (21, 021, 21, 21);

INSERT INTO PRESCRIPTION\_INFO VALUES (22, 022, 22, 22);

INSERT INTO PRESCRIPTION\_INFO VALUES (23, 023, 23, 23);

INSERT INTO PRESCRIPTION\_INFO VALUES (24, 024, 24, 24);

INSERT INTO PRESCRIPTION\_INFO VALUES (25, 025, 25, 25);

INSERT INTO PRESCRIPTION\_INFO VALUES (26, 026, 26, 26);

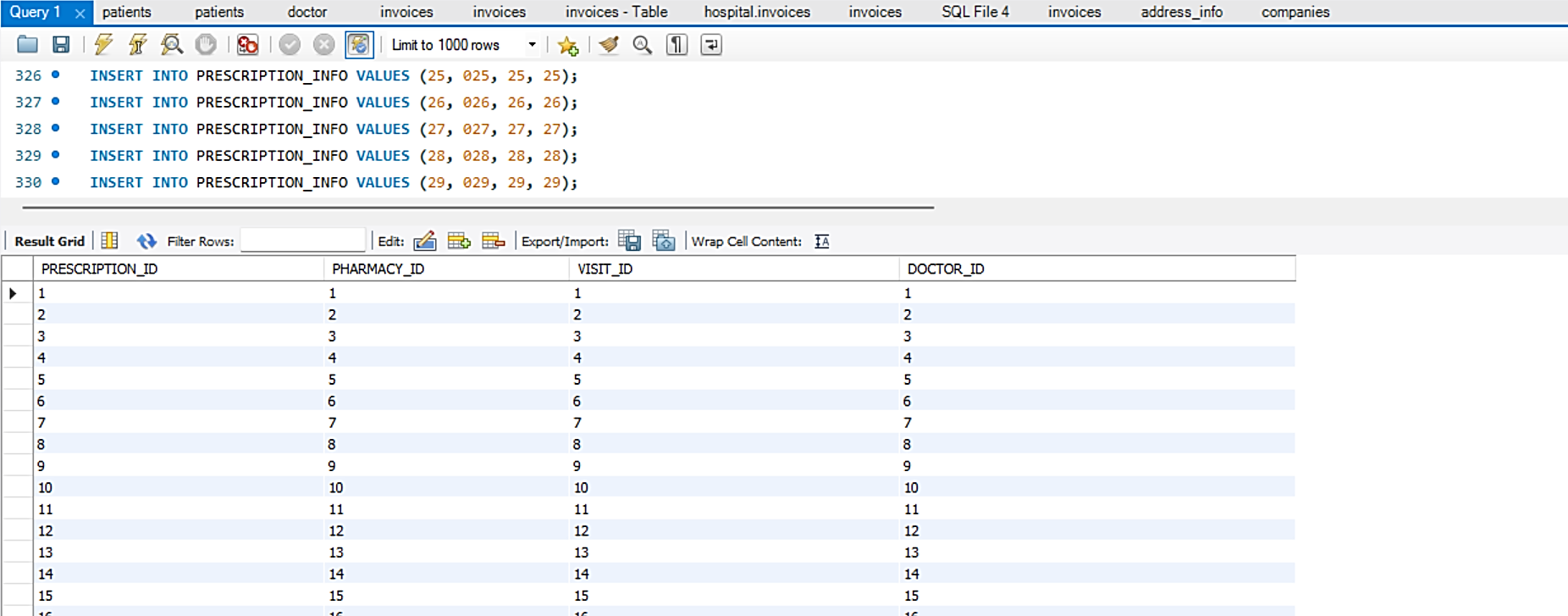
INSERT INTO PRESCRIPTION\_INFO VALUES (27, 027, 27, 27);

INSERT INTO PRESCRIPTION\_INFO VALUES (28, 028, 28, 28);

INSERT INTO PRESCRIPTION\_INFO VALUES (29, 029, 29, 29);

INSERT INTO PRESCRIPTION\_INFO VALUES (30, 030, 30, 30);

SELECT \* FROM PRESCRIPTION\_INFO;



1. **Creation of the table CLAIM:**

CREATE TABLE CLIAMS (PATIENT\_ID INT NOT NULL,

INSURENCE\_ID INT NOT NULL,

PRIMARY KEY(PATIENT\_ID, INSURENCE\_ID));

* The following statements are used to generate 30 records in the respective Table:

INSERT INTO CLIAMS VALUES (01, 1);

INSERT INTO CLIAMS VALUES (02, 2);

INSERT INTO CLIAMS VALUES (03, 3);

INSERT INTO CLIAMS VALUES (04, 4);

INSERT INTO CLIAMS VALUES (05, 5);

INSERT INTO CLIAMS VALUES (06, 6);

INSERT INTO CLIAMS VALUES (07, 7);

INSERT INTO CLIAMS VALUES (08, 8);

INSERT INTO CLIAMS VALUES (09, 9);

INSERT INTO CLIAMS VALUES (10, 10);

INSERT INTO CLIAMS VALUES (11, 11);

INSERT INTO CLIAMS VALUES (12, 12);

INSERT INTO CLIAMS VALUES (13, 13);

INSERT INTO CLIAMS VALUES (14, 14);

INSERT INTO CLIAMS VALUES (15, 15);

INSERT INTO CLIAMS VALUES (16, 16);

INSERT INTO CLIAMS VALUES (17, 17);

INSERT INTO CLIAMS VALUES (18, 18);

INSERT INTO CLIAMS VALUES (19, 19);

INSERT INTO CLIAMS VALUES (20, 20);

INSERT INTO CLIAMS VALUES (21, 21);

INSERT INTO CLIAMS VALUES (22, 22);

INSERT INTO CLIAMS VALUES (23, 23);

INSERT INTO CLIAMS VALUES (24, 24);

INSERT INTO CLIAMS VALUES (25, 25);

INSERT INTO CLIAMS VALUES (26, 26);

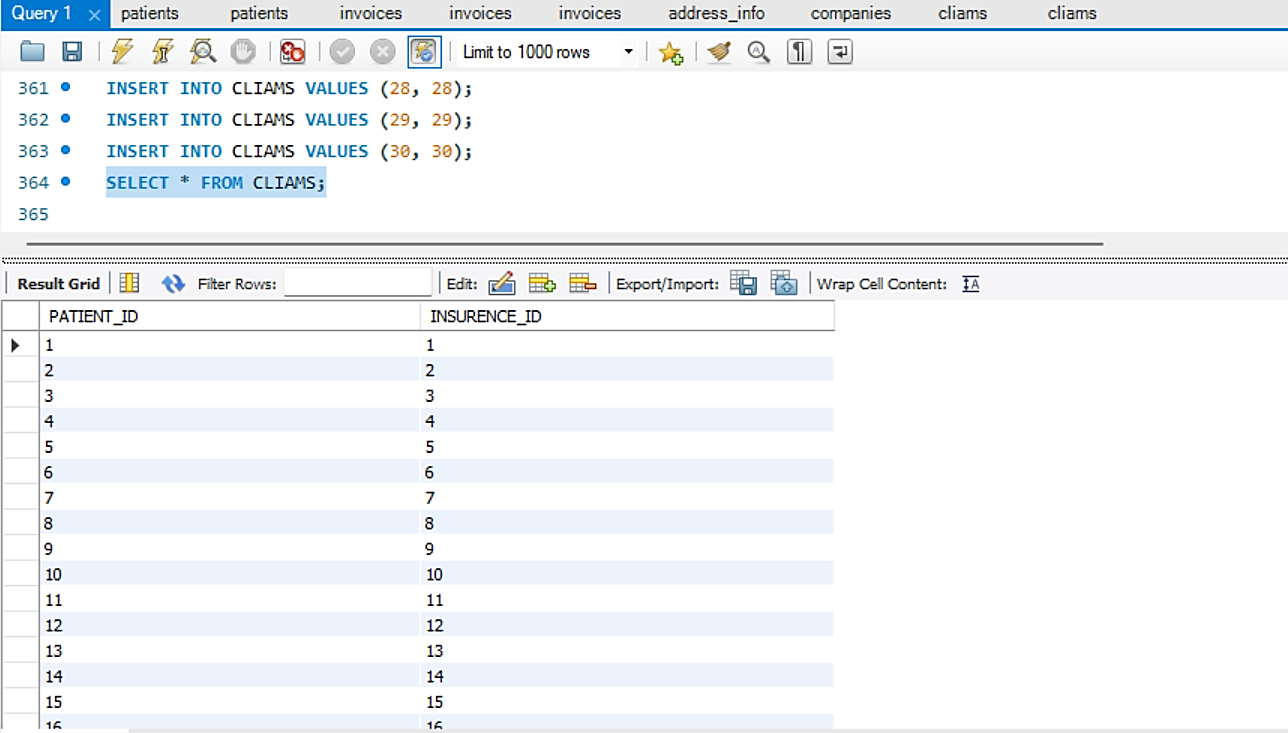
INSERT INTO CLIAMS VALUES (27, 27);

INSERT INTO CLIAMS VALUES (28, 28);

INSERT INTO CLIAMS VALUES (29, 29);

INSERT INTO CLIAMS VALUES (30, 30);

SELECT \* FROM CLIAMS;



1. **Creation of the table INSURANCE\_COMPANIES:**

CREATE TABLE INSURENCE\_COMPANIES (INSURENCE\_ID INT NOT NULL,

COMPANY\_ID INT NOT NULL,

PRIMARY KEY(INSURENCE\_ID, COMPANY\_ID));

* The following statements are used to generate 30 records in the respective Table:

INSERT INTO INSURENCE\_COMPANIES VALUES (1,1);

INSERT INTO INSURENCE\_COMPANIES VALUES (2,2);

INSERT INTO INSURENCE\_COMPANIES VALUES (3,3);

INSERT INTO INSURENCE\_COMPANIES VALUES (4,4);

INSERT INTO INSURENCE\_COMPANIES VALUES (5,5);

INSERT INTO INSURENCE\_COMPANIES VALUES (6,6);

INSERT INTO INSURENCE\_COMPANIES VALUES (7,7);

INSERT INTO INSURENCE\_COMPANIES VALUES (8,8);

INSERT INTO INSURENCE\_COMPANIES VALUES (9,9);

INSERT INTO INSURENCE\_COMPANIES VALUES (10,10);

INSERT INTO INSURENCE\_COMPANIES VALUES (11,11);

INSERT INTO INSURENCE\_COMPANIES VALUES (12,12);

INSERT INTO INSURENCE\_COMPANIES VALUES (13,13);

INSERT INTO INSURENCE\_COMPANIES VALUES (14,14);

INSERT INTO INSURENCE\_COMPANIES VALUES (15,15);

INSERT INTO INSURENCE\_COMPANIES VALUES (16,16);

INSERT INTO INSURENCE\_COMPANIES VALUES (17,17);

INSERT INTO INSURENCE\_COMPANIES VALUES (18,18);

INSERT INTO INSURENCE\_COMPANIES VALUES (19,19);

INSERT INTO INSURENCE\_COMPANIES VALUES (20,20);

INSERT INTO INSURENCE\_COMPANIES VALUES (21,21);

INSERT INTO INSURENCE\_COMPANIES VALUES (22,22);

INSERT INTO INSURENCE\_COMPANIES VALUES (23,23);

INSERT INTO INSURENCE\_COMPANIES VALUES (24,24);

INSERT INTO INSURENCE\_COMPANIES VALUES (25,25);

INSERT INTO INSURENCE\_COMPANIES VALUES (26,26);

INSERT INTO INSURENCE\_COMPANIES VALUES (27,27);

INSERT INTO INSURENCE\_COMPANIES VALUES (28,28);

INSERT INTO INSURENCE\_COMPANIES VALUES (29,29);

INSERT INTO INSURENCE\_COMPANIES VALUES (30,30);

SELECT \* FROM INSURENCE\_COMPANIES

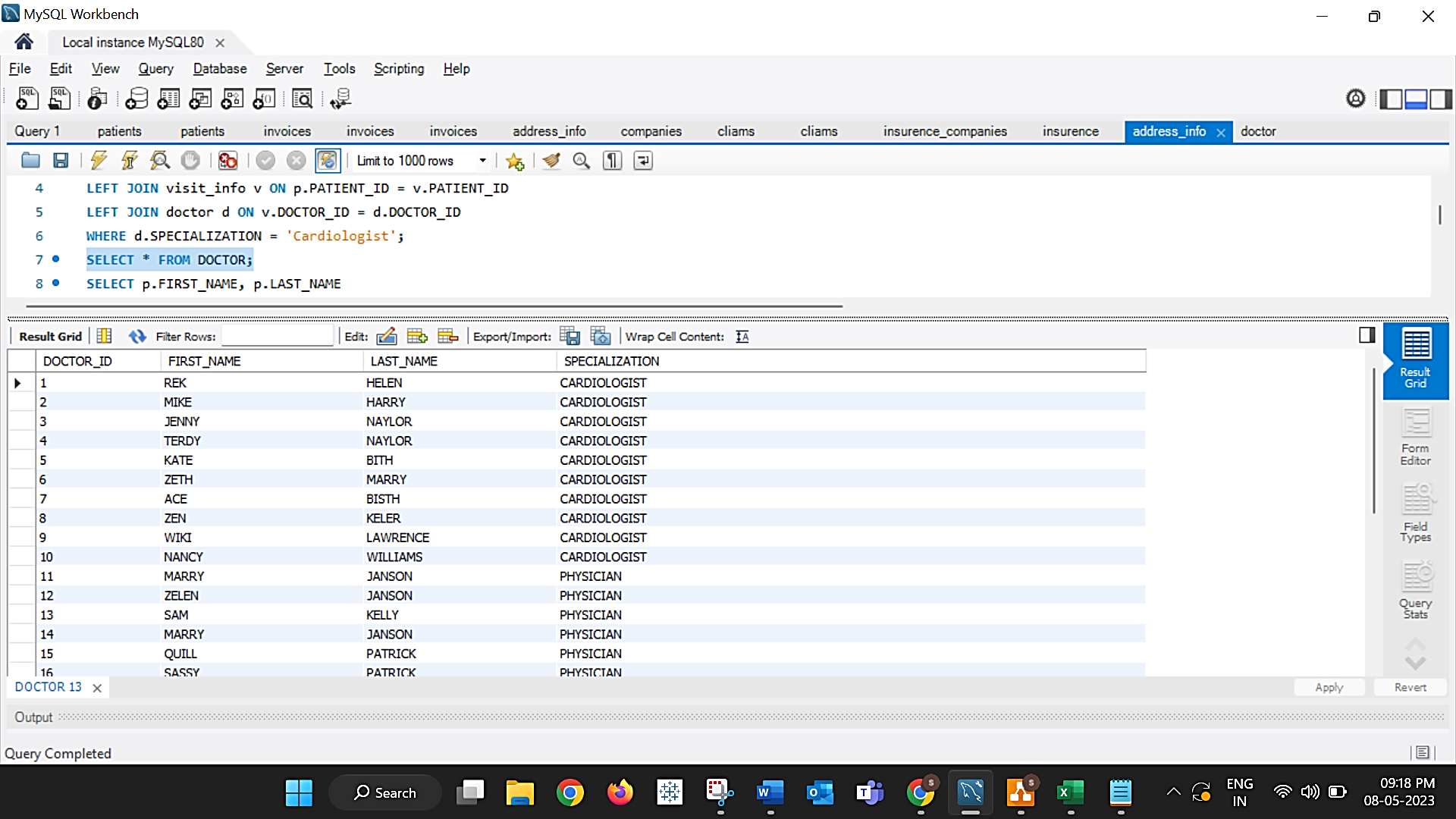


**Data Retrieval and Reports:**

1. **Give the list of all doctors with specializations**

**Query:** SELECT \* FROM DOCTOR;

**Result:** This provides the list of all doctors with their specializations along with their full names



1. **Retrieve the information of the patients who have consulted the cardiologist doctor**

**Query:**

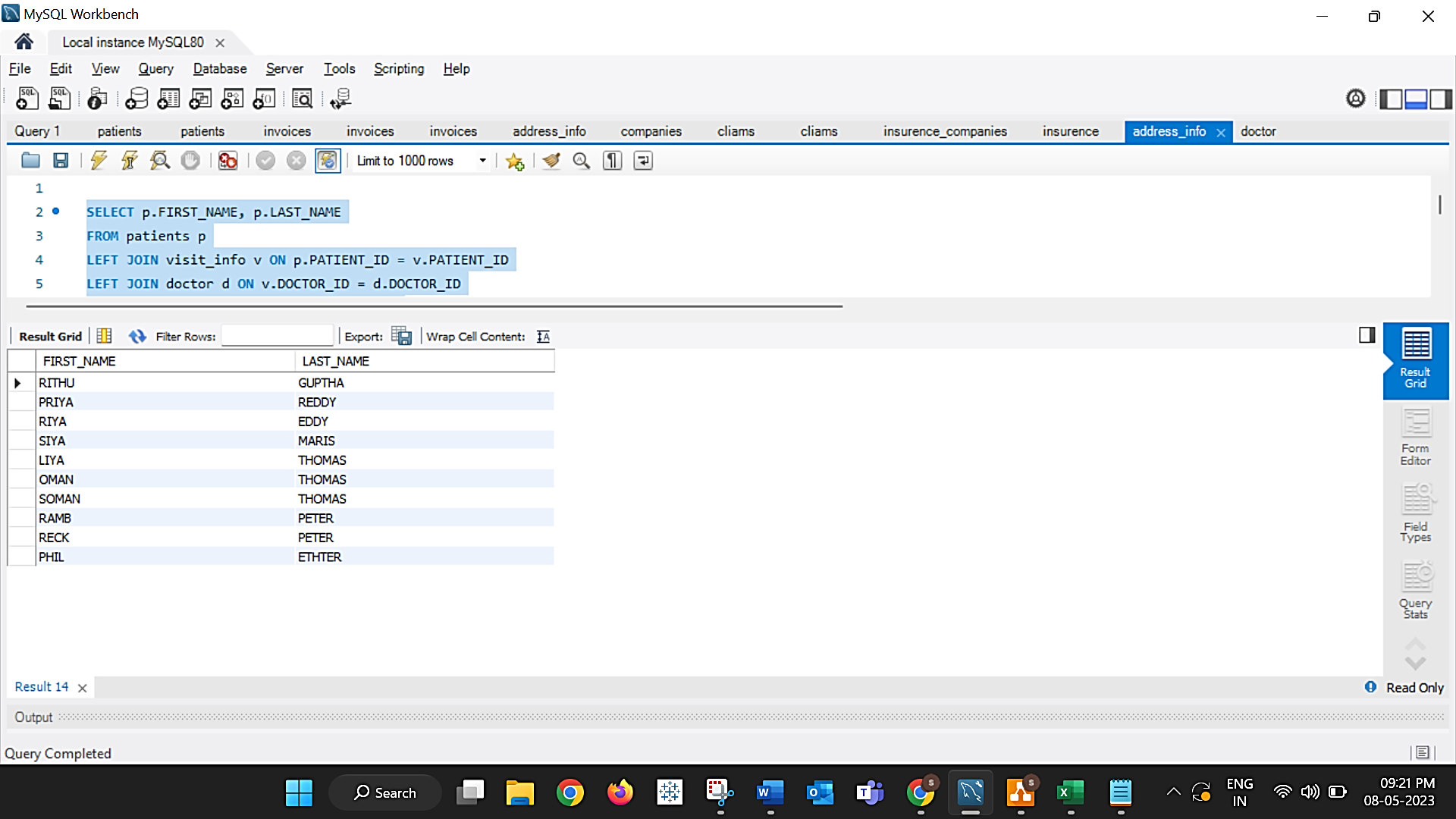
SELECT p.FIRST\_NAME, p.LAST\_NAME FROM patients p

LEFT JOIN visit\_info v ON p.PATIENT\_ID = v.PATIENT\_ID

LEFT JOIN doctor d ON v.DOCTOR\_ID = d.DOCTOR\_ID

WHERE d.SPECIALIZATION = 'Cardiologist';

**Result:** This provides the list of all patients who have visited the cardiologist

****

1. **Retrieve the names of all patients who have received a discount of 30 on their insurance policies:**

**Query:**

SELECT p.FIRST\_NAME, p.LAST\_NAME FROM patients p

LEFT JOIN cliams c ON p.PATIENT\_ID = c.PATIENT\_ID

LEFT JOIN insurence ins ON c.INSURENCE\_ID = ins.INSURENCE\_ID

WHERE ins.DISCOUNT = 30;

**Result:** This provides the list of all patients who have availed a discount of 30% on their bill

****

1. **List the name of the drug and the total amount spent by the patient with id ‘23’**

**Query:**

SELECT ph.name, SUM(i.amount) AS total\_amount FROM invoices i

JOIN visit\_info v ON i.invoice\_id = v.invoice\_id

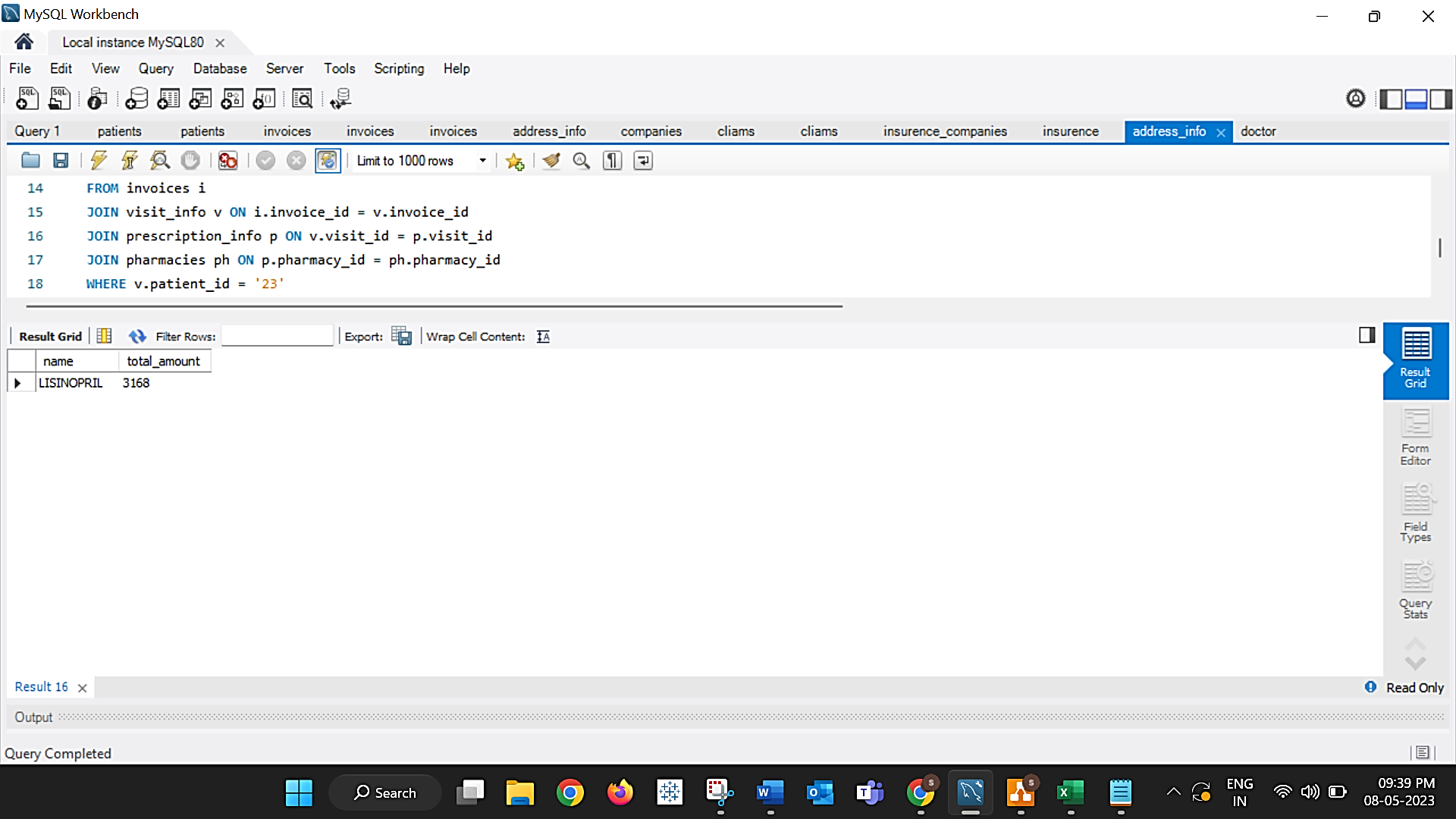
JOIN prescription\_info p ON v.visit\_id = p.visit\_id

JOIN pharmacies ph ON p.pharmacy\_id = ph.pharmacy\_id

WHERE v.patient\_id = '23'

GROUP BY ph.name;

**Result:** This provides details of specific patient’s medication and the amount spent on it.

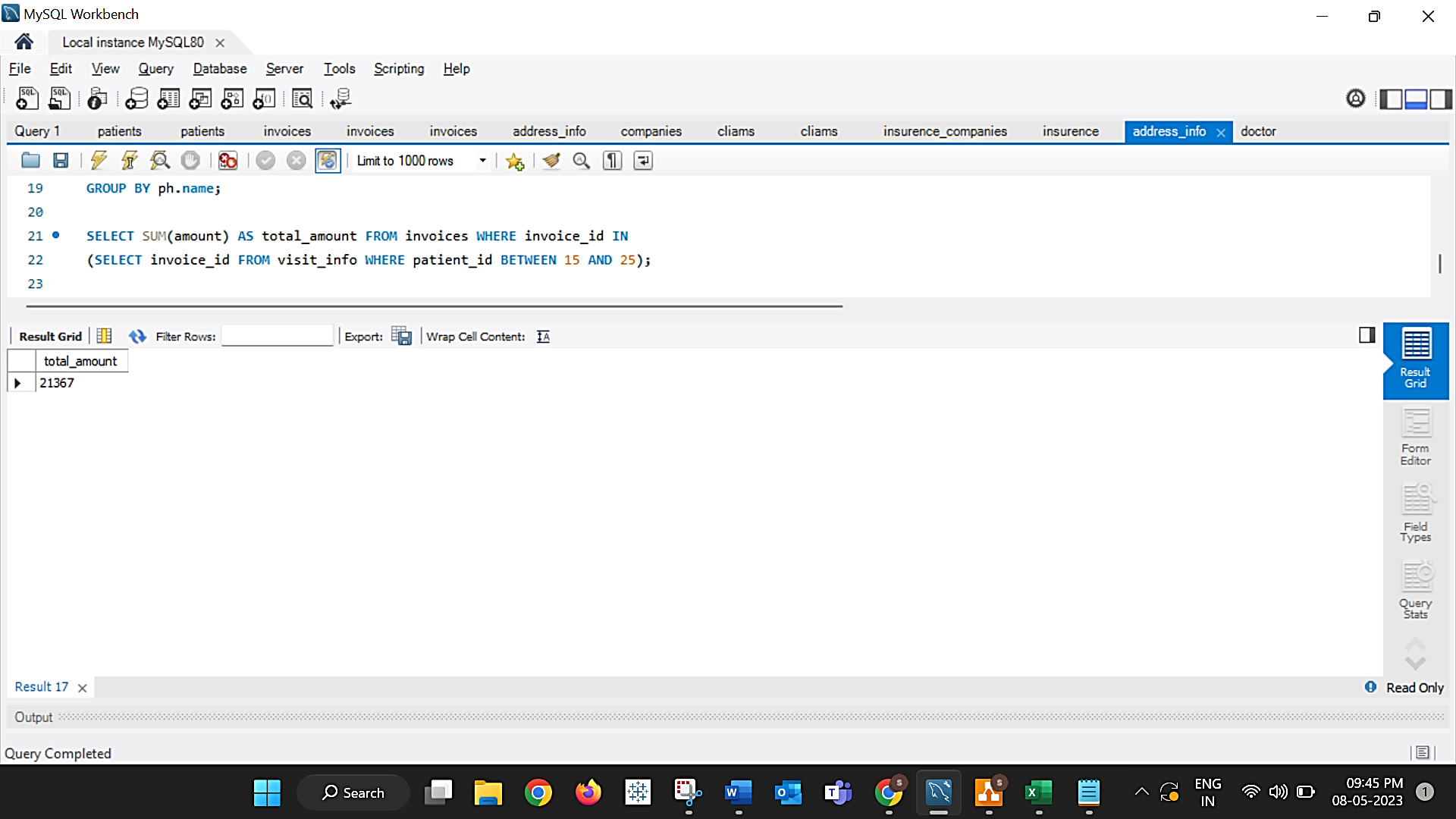
****

1. **List the total amount from the invoice bill generated from the patient ids 15 to 25**

**Query:**

SELECT SUM(amount) AS total\_amount FROM invoices WHERE invoice\_id IN (SELECT invoice\_id FROM visit\_info WHERE patient\_id BETWEEN 15 AND 25);

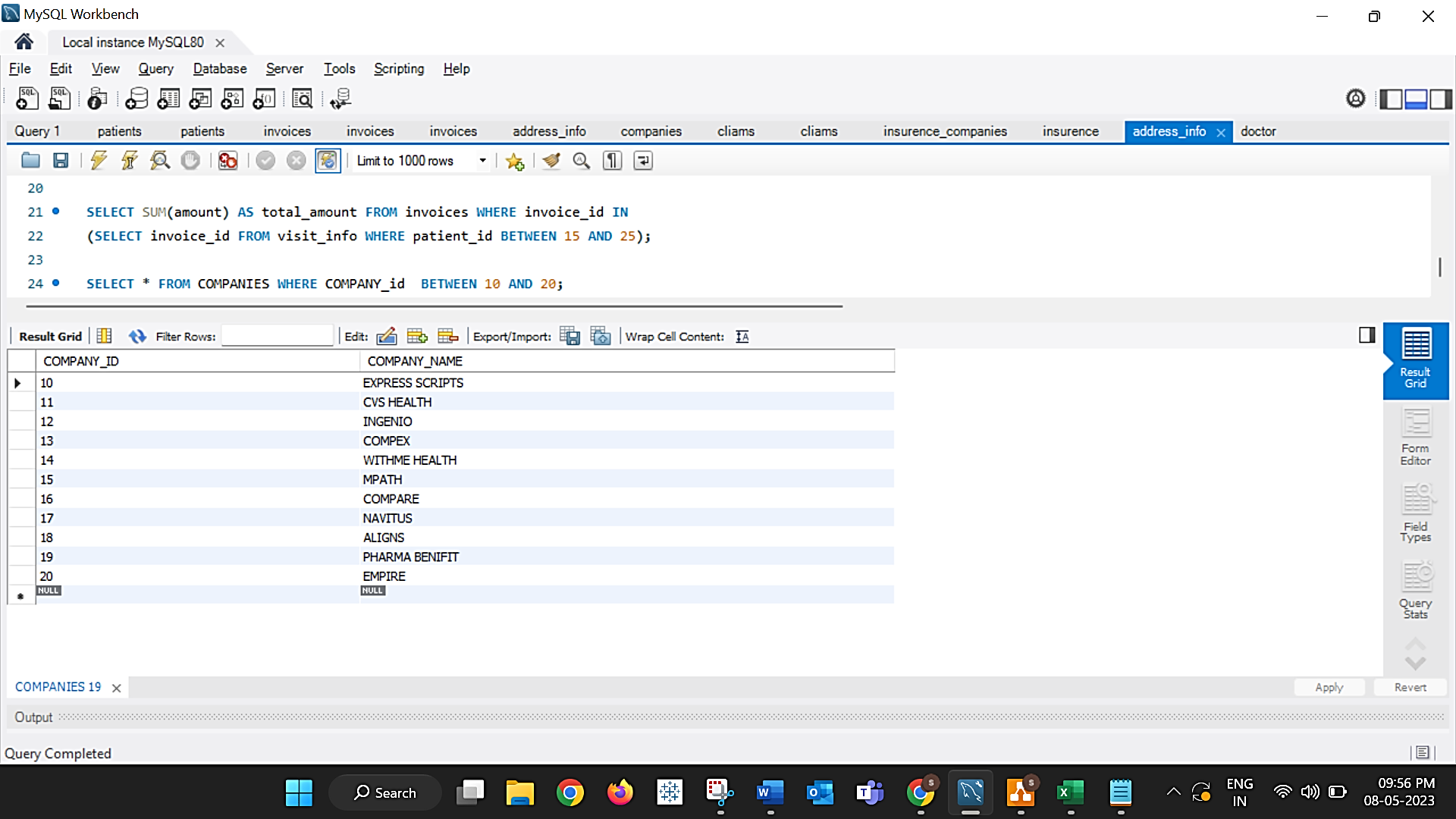
**Result:** This provides the overall bill generated by selected patients

****

1. **Retrieve the insurance companies with company ID 10 to 20**

**Query:** SELECT \* FROM COMPANIES WHERE COMPANY\_id BETWEEN 10 AND 20;

**Result:** This provides the list of insurance companies with specific to selected ID’s



**Conclusion:** The designed DBMS of NextGen, Electronic health record proves to be more advantageous in a hospital system and provides benefits to patients, pharmacists and also to doctors during the process and also in retrieving the valid information accurately and precisely in stipulated time