DSCI D532 : HOMEWORK 1 (schekka@iu.edu)

```
CREATE DATABASE adt_hw1;

-- Creating the hospitals table by following all the constraints properly
CREATE TABLE IF NOT EXISTS hospitals (
    hospital_id INT AUTO_INCREMENT PRIMARY KEY,
    hospital_name VARCHAR(255) NOT NULL,
    state CHAR(2) NOT NULL,
    city VARCHAR(100) NOT NULL,
    doctor VARCHAR(255) NOT NULL
);


-- Creating the doctors table by following all the constraints properly
CREATE TABLE IF NOT EXISTS doctors (
    doctor_id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(255) NOT NULL,
    gender CHAR(7) CHECK (gender IN ('Male', 'Female', 'O')),
    insurance CHAR(3) CHECK (insurance IN ('Yes', 'No')),
    new_patients CHAR(3) CHECK (new_patients IN ('Yes', 'No')),
    speciality_one VARCHAR(100),
    speciality_two VARCHAR(100),
    speciality_three VARCHAR(100),
    license CHAR(3) CHECK (license IN ('MFT', 'PhD', 'MD')),
    phone CHAR(10)
);


INSERT INTO
        doctors(name, gender, insurance, new_patients, speciality_one,
                        speciality_two, speciality_three, license, phone)
VALUES
        ('Flora Martinez', 'Female', 'Yes', 'Yes', 'Diabetes', 'Cholesterol',
'immunology', 'MD', '8495776489'),
        ('Andy James', 'Male', 'Yes', 'No', 'Hypertension', 'Diabetes', 'PTSD', 'PhD',
'2340894766'),
```

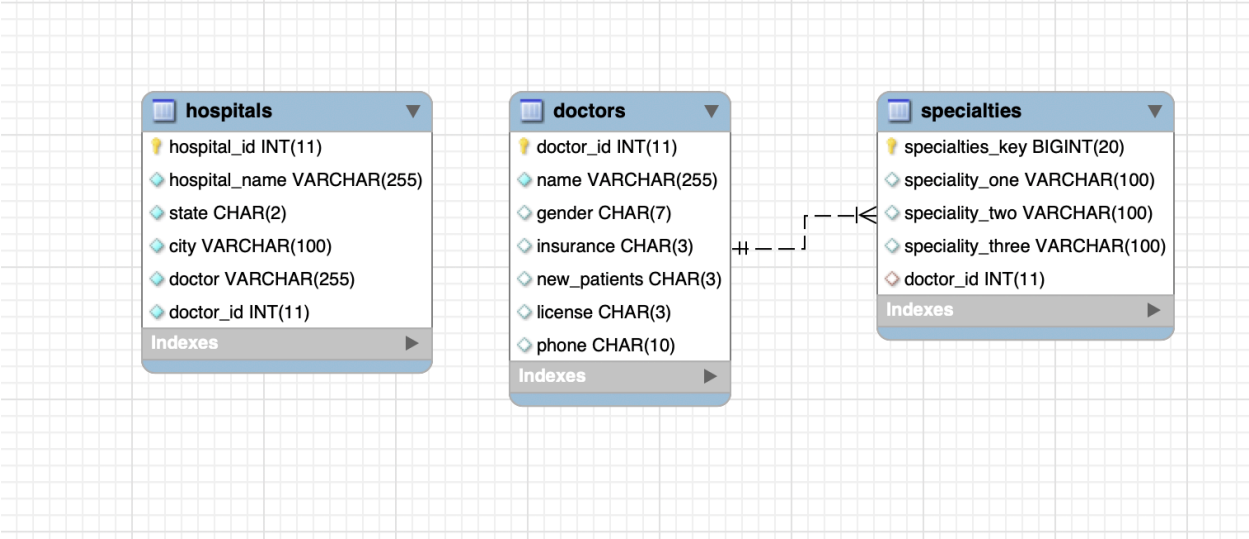
('Hannah Myers', 'Female', 'No', 'Yes', 'Diabetes', 'Hypertension', 'immunology', 'MD', '9907846574'),
('Jane Huang', 'Female', 'Yes', 'Yes', 'Dermatology', 'Hypertension', 'immunology', 'MD', '4507856797'),
('April Adams', 'Female', 'No', 'Yes', 'OCD', 'Hypertension', 'PTSD', 'MFT', '4507856797'),
('Jon Schaffer', 'Male', 'Yes', 'No', 'BPD', 'immunology', 'Dermatology', 'PhD', '9907846574'),
('Shauna West','Female', 'Yes', 'Yes', 'ADHD', 'immunology', 'OCD', 'MD', '8495776480'),
('Juan Angelo', 'Male', 'No', 'Yes', 'Diabetes', 'immunology', 'Dermatology', 'MD', '4507856797'),
('Christie Yang', 'Female', 'Yes', 'Yes', 'Autism', 'ADHD', 'OCD', 'PhD', '4507856796'),
('Annika Neusler', 'Female', 'Yes', 'No', 'Addiction', 'Dermatology', 'PTSD', 'MFT', '9907846575'),
('Simone Anderson', 'Female', 'No', 'No', 'Hypertension', 'Dermatology', 'PTSD', 'MD', '8304498765'),
('Ted Nyguen', 'Male', 'Yes', 'Yes', 'ADHD', 'Hypertension', 'Allergy', 'PhD', '4301239990'),
('Valentino Rossi', 'Male', 'Yes', 'Yes', 'Autism', 'Hypertension', 'Dermatology', 'MD', '8304498765'),
('Jessica Armer', 'Female', 'No', 'Yes', 'PTSD', 'immunology', 'Dermatology', 'MD', '3330456612'),
('Sid Michaels', 'Female', 'Yes', 'Yes', 'OCD', 'Allergy', 'Hypertension', 'MFT', '4301239997'),
('Yen Waters', 'Male', 'Yes', 'Yes', 'Hypertension', 'Dermatology', 'ADHD', 'PhD', '4507856796'),
('Ru Izaelia', 'Female', 'No', 'Yes', 'immunology', 'BPD', 'Allergy', 'MD', '4301239990'),
('Vishal Rao', 'Male', 'Yes', 'Yes', 'Dermatology', 'Diabetes', 'Hypertension', 'MD', '7305557894'),
('Lana John', 'Female', 'Yes', 'Yes', 'Hypertension', 'Allergy', 'OCD', 'MFT', '7305557894'),
('Izzie Geralt', 'Female', 'Yes', 'Yes', 'Dermatology', 'Addiction', 'Hypertension', 'MD', '4301239990');

```
INSERT INTO
	hospitals(hospital_name, state, city, doctor)
VALUES
	('Van Holsen Community Hospital', 'CA', 'San Francisco', 'Flora Martinez'),
	('Clear Water Services', 'CA', 'San Diego', 'Andy James'),
	('Imagery Health', 'CA', 'Sacramento', 'Hannah Myers'),
	('Blue Cross Clinic', 'CA', 'Los Angeles', 'Jane Huang'),
	('Blue Cross Clinic', 'CA', 'Los Angeles', 'April Adams'),
	('Imagery Health', 'CA', 'Sacramento', 'Jon Schaffer'),
	('Van Holsen Community Hospital', 'CA', 'Long Beach', 'Shauna West'),
	('Blue Cross Clinic', 'CA', 'Santa Barbara', 'Juan Angelo'),
	('Blue Cross Clinic', 'CA', 'San Francisco', 'Christie Yang'),
	('Imagery Health', 'CA', 'Auburn', 'Annika Neusler'),
	('Holistic Health Services', 'CA', 'Santa Barbara', 'Simone Anderson'),
	('Open Clinic', 'CA', 'San Jose', 'Ted Nyguen'),
	('Holistic Health Services', 'CA', 'Santa Barbara', 'Valentino Rossi'),
	('Clark Jamison Hospitals', 'CA', 'Fresno', 'Jessica Armer'),
	('Open Clinic', 'CA', 'Oakland', 'Sid Michaels'),
	('Blue Cross Clinic', 'CA', 'San Francisco', 'Yen Waters'),
	('Open Clinic', 'CA', 'San Jose', 'Ru Izaelia'),
	('Clear Minds Community', 'CA', 'Sacramento', 'Vishal Rao'),
	('Clear Minds Community', 'CA', 'Sacramento', 'Lana John'),
	('Open Clinic', 'CA', 'San Jose', 'Izzie Geralt');


SELECT * FROM doctors;

SELECT * FROM hospitals;

-- Alter the hospitals table to add the doctor_id column
ALTER TABLE hospitals
ADD COLUMN doctor_id INT NOT NULL;

UPDATE hospitals
INNER JOIN doctors
ON hospitals.doctor = doctors.name
SET hospitals.doctor_id = doctors.doctor_id;
```

```
CREATE TABLE IF NOT EXISTS specialties(
    specialties_key serial PRIMARY KEY,
    speciality_one VARCHAR(100),
    speciality_two VARCHAR(100),
    speciality_three VARCHAR(100),
    doctor_id INTEGER,
    CONSTRAINT fk_doctor
        FOREIGN KEY(doctor_id)
        REFERENCES doctors(doctor_id));

INSERT INTO specialties(doctor_id, speciality_one, speciality_two,
speciality_three)
SELECT doctor_id, speciality_one, speciality_two, speciality_three
FROM doctors;

ALTER TABLE doctors
DROP COLUMN speciality_one,
DROP COLUMN speciality_two,
DROP COLUMN speciality_three;

-- Exploration
-- 1.  How many doctors are currently accepting new patients?
SELECT COUNT(*) AS doctors_accepting_new_patients
FROM doctors
WHERE new_patients = 'Yes';

-- 2. What is the distribution of doctors across different cities in California?
SELECT city, COUNT(doctor_id) AS number_of_doctors
FROM hospitals
GROUP BY city;

-- 3. How many male and female doctors have each type of license?
SELECT
  license,
  SUM(CASE WHEN gender = 'Male' THEN 1 ELSE 0 END) AS male_doctors,
```

```
  SUM(CASE WHEN gender = 'Female' THEN 1 ELSE 0 END) AS
female_doctors,
  SUM(CASE WHEN gender = 'O' THEN 1 ELSE 0 END) AS
other_gender_doctors
FROM doctors
GROUP BY license;


-- 4. How many doctors have a license in 'MD' and are treating 'Diabetes'?
SELECT COUNT(*) AS md_diabetes_doctors
FROM doctors
WHERE license = 'MD' AND doctor_id IN (
    SELECT doctor_id
    FROM specialties
    WHERE speciality_one = 'Diabetes' OR speciality_two = 'Diabetes' OR
speciality_three = 'Diabetes'
);

-- 5. What is the average number of doctors per hospital in the database?
SELECT AVG(doctor_count) as average_doctors_per_hospital
FROM (
    SELECT hospital_id, COUNT(doctor_id) as doctor_count
    FROM hospitals
    GROUP BY hospital_id
) AS doctor_counts;

-- Visualization
-- Visualization 1: Distribution of Doctors by Specialty
SELECT speciality_one, COUNT(*) AS number_of_doctors
FROM specialties
GROUP BY speciality_one
ORDER BY number_of_doctors DESC;

-- Visualization 2: Acceptance of New Patients by Gender
SELECT gender, new_patients, COUNT(*) AS number_of_doctors
FROM doctors
GROUP BY gender, new_patients;
```

Schema of the normalized database(EER Diagram):



Python notebook: Exploration

```
config = {
 'user': 'root',
 'password': 'root',
 'host': '127.0.0.1',
 'port': 8889,
 'database': 'adt_hw1',
 'raise_on_warnings': True
}
```

```
pip install mysql-connector-python
```

```
# create a MySQL server connection object
import mysql.connector
mydb = mysql.connector.connect(**config)
```

```
# cursor object
my_cursor = mydb.cursor(dictionary=True)
```

```
import mysql.connector
import pandas as pd
```

```python
# Function to run a single query and return the results as a pandas DataFrame
def run_query(query):
    my_cursor.execute(query)
    rows = my_cursor.fetchall()
    return pd.DataFrame(rows)
```

Query 1:
SELECT COUNT(*) AS doctors_accepting_new_patients
FROM doctors
WHERE new_patients = 'Yes';

Exploration

```python
# Query 1: Count of doctors accepting new patients
query1 = "SELECT COUNT(*) AS doctors_accepting_new_patients FROM doctors WHERE new_patients = 'Yes';"
result1 = run_query(query1)
print("Query 1 Results:")
print(result1)
```

[10] ✓ 0.3s

```
Query 1 Results:
   doctors_accepting_new_patients
0                              16
```

Query 2

SELECT city, COUNT(doctor_id) AS number_of_doctors
FROM hospitals
GROUP BY city;

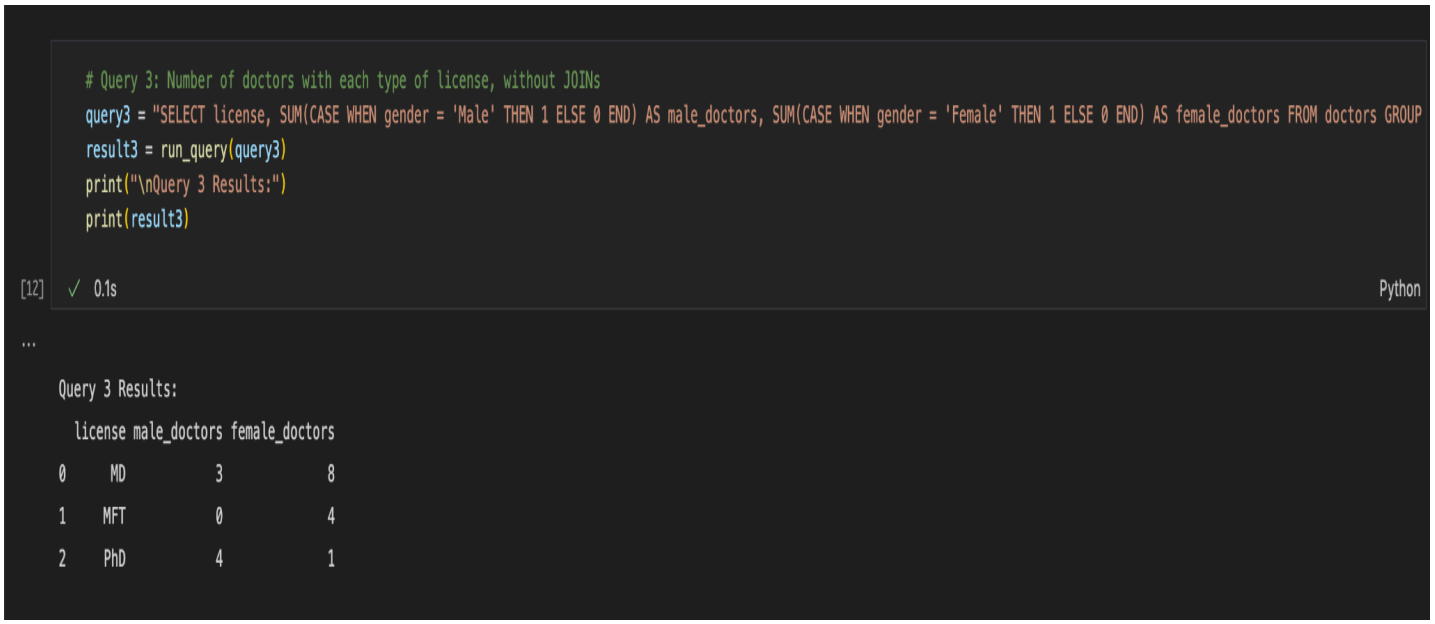
```
# Query 2: Distribution of doctors across different cities
query2 = "SELECT city, COUNT(doctor_id) AS number_of_doctors FROM hospitals GROUP BY city;"
result2 = run_query(query2)
print("\nQuery 2 Results:")
print(result2)
```

[11] ✓ 0.1s

```
Query 2 Results:
            city  number_of_doctors
0         Auburn                  1
1         Fresno                  1
2     Long Beach                  1
3    Los Angeles                  2
4        Oakland                  1
5     Sacramento                  4
6      San Diego                  1
7  San Francisco                  3
8       San Jose                  3
9  Santa Barbara                  3
```

Query 3:

```
SELECT
  license,
  SUM(CASE WHEN gender = 'Male' THEN 1 ELSE 0 END) AS male_doctors,
  SUM(CASE WHEN gender = 'Female' THEN 1 ELSE 0 END) AS
female_doctors,
  SUM(CASE WHEN gender = 'O' THEN 1 ELSE 0 END) AS
other_gender_doctors
FROM doctors
GROUP BY license;
```



```
# Query 3: Number of doctors with each type of license, without JOINs
query3 = "SELECT license, SUM(CASE WHEN gender = 'Male' THEN 1 ELSE 0 END) AS male_doctors, SUM(CASE WHEN gender = 'Female' THEN 1 ELSE 0 END) AS female_doctors FROM doctors GROUP
result3 = run_query(query3)
print("\nQuery 3 Results:")
print(result3)
```

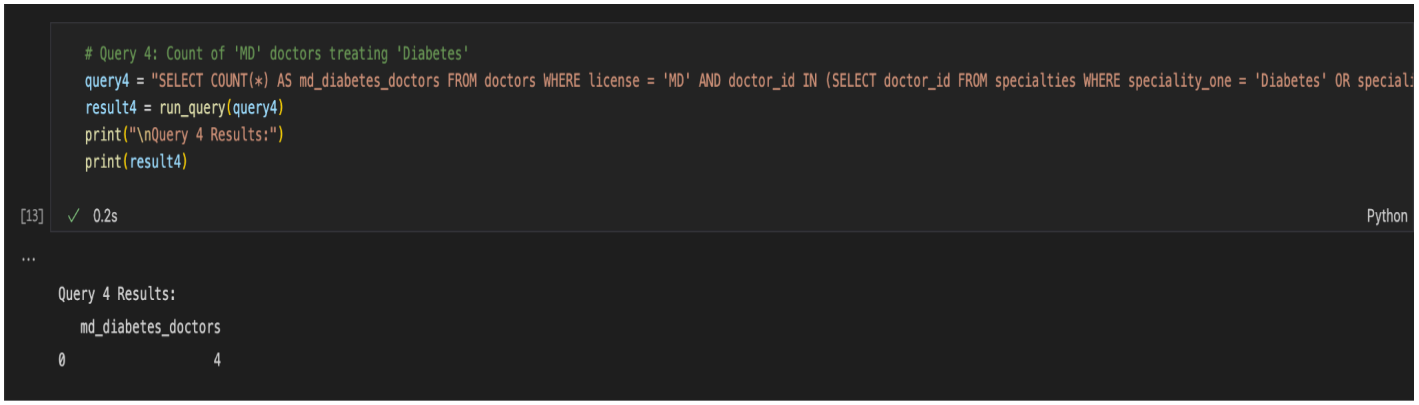
[12] ✓ 0.1s Python

...

```
Query 3 Results:
  license male_doctors female_doctors
0      MD            3              8
1     MFT            0              4
2     PhD            4              1
```

Query 4 :

```
SELECT COUNT(*) AS md_diabetes_doctors
FROM doctors
WHERE license = 'MD' AND doctor_id IN (
   SELECT doctor_id
   FROM specialties
   WHERE speciality_one = 'Diabetes' OR speciality_two = 'Diabetes' OR
speciality_three = 'Diabetes'
);
```



Query 5

```
SELECT AVG(doctor_count) as average_doctors_per_hospital
FROM (
   SELECT hospital_id, COUNT(doctor_id) as doctor_count
   FROM hospitals
   GROUP BY hospital_id
) AS doctor_counts;
```

```
# Query 5: Average number of doctors per hospital
query5 = "SELECT AVG(doctor_count) as average_doctors_per_hospital FROM (SELECT hospital_id, COUNT(doctor_id) as doctor_count FROM hospitals GROUP BY hospital_id) AS doctor_counts;
result5 = run_query(query5)
print("\nQuery 5 Results:")
print(result5)
```

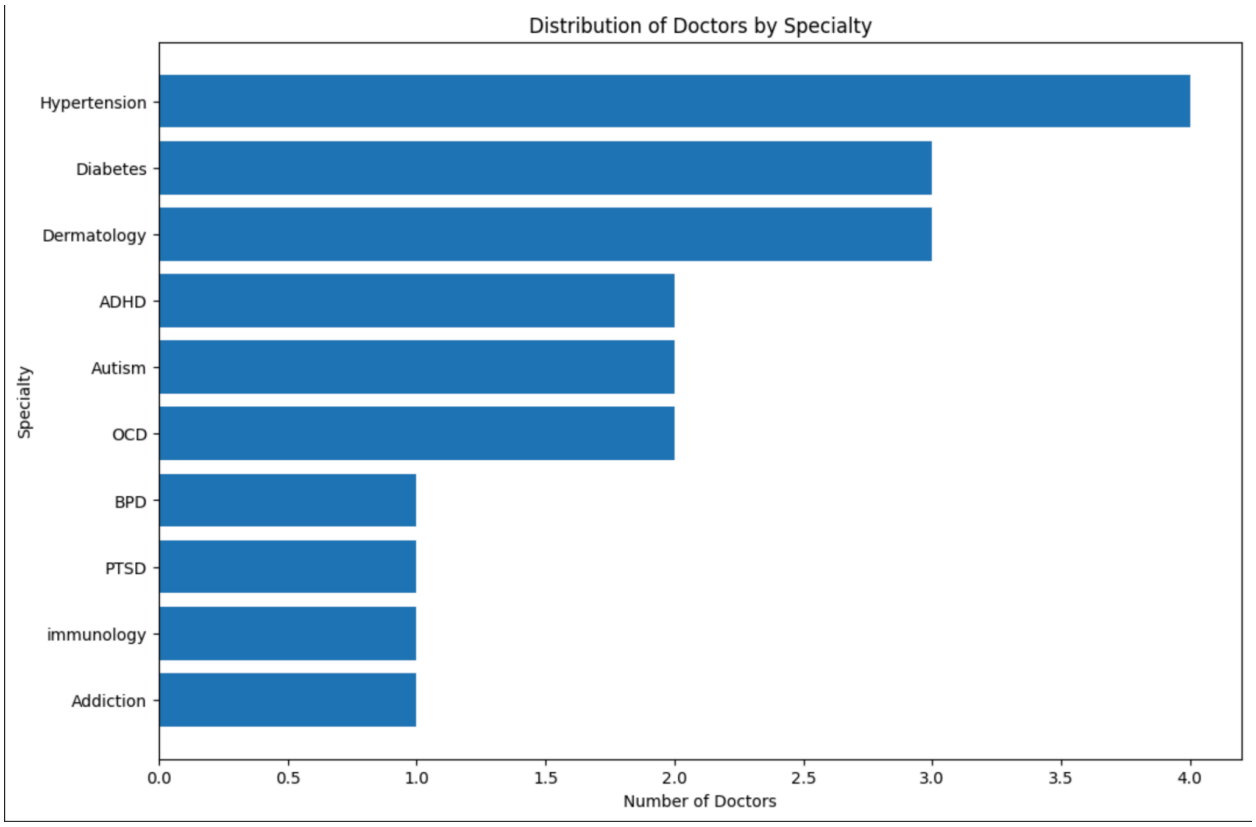
[14] ✓ 0.1s Python

```
Query 5 Results:
  average_doctors_per_hospital
0                       1.0000
```

Visualization 1 : Distribution of Doctors by Specialty

```
# Executing the query to get data
query_specialty_distribution = """
SELECT speciality_one, COUNT(*) AS number_of_doctors
FROM specialties
GROUP BY speciality_one
ORDER BY number_of_doctors DESC;
"""
specialty_distribution = run_query(query_specialty_distribution)

# Visualization with matplotlib
import matplotlib.pyplot as plt

plt.figure(figsize=(12, 8))
```

```
plt.barh(specialty_distribution['speciality_one'],
specialty_distribution['number_of_doctors'])
plt.xlabel('Number of Doctors')
plt.ylabel('Specialty')
plt.title('Distribution of Doctors by Specialty')
plt.gca().invert_yaxis()  # To display the highest count at the top
plt.show()
```



Visualization 2 : Acceptance of New Patients by Gender

```
# Executing the query to get data
query_new_patients_by_gender = """
SELECT gender, new_patients, COUNT(*) AS number_of_doctors
FROM doctors
GROUP BY gender, new_patients;
"""
new_patients_by_gender = run_query(query_new_patients_by_gender)
```

```
# Visualization with seaborn
import seaborn as sns

plt.figure(figsize=(10, 6))
sns.barplot(x='gender', y='number_of_doctors', hue='new_patients',
data=new_patients_by_gender)
plt.xlabel('Gender')
plt.ylabel('Number of Doctors')
plt.title('Acceptance of New Patients by Gender')
plt.legend(title='New Patients')
plt.show()
```