import sqlite3

def checkSetup():

conn = sqlite3.connect('gas.db')

cursor = conn.cursor()

cursor.execute("SELECT name FROM sqlite\_master WHERE type='table' AND name='instructors'")

result = cursor.fetchone()

conn.close()

if result is None:

return False

return True

def setup():

conn = sqlite3.connect('gas.db')

cursor = conn.cursor()

create\_instructors\_table = """

CREATE TABLE IF NOT EXISTS instructors (

id INTEGER PRIMARY KEY,

name TEXT NOT NULL,

hours INTEGER NOT NULL,

schedule TEXT NOT NULL,

active BOOLEAN NOT NULL DEFAULT 1 CHECK (

active IN (0, 1)

)

);

"""

create\_rooms\_table = """

CREATE TABLE IF NOT EXISTS rooms (

id INTEGER PRIMARY KEY,

name TEXT NOT NULL,

type TEXT NOT NULL,

schedule TEXT NOT NULL,

active BOOLEAN NOT NULL DEFAULT 1 CHECK (

active IN (0, 1)

)

);

"""

create\_subjects\_table = """

CREATE TABLE IF NOT EXISTS subjects (

id INTEGER PRIMARY KEY,

name TEXT NOT NULL,

hours REAL NOT NULL,

code TEXT NOT NULL,

description TEXT NOT NULL,

instructors TEXT NOT NULL,

divisible BOOLEAN NOT NULL DEFAULT 1 CHECK (

divisible IN (0, 1)

),

type TEXT NOT NULL

);

"""

create\_sections\_table = """

CREATE TABLE IF NOT EXISTS sections (

id INTEGER PRIMARY KEY,

name TEXT NOT NULL,

schedule TEXT NOT NULL,

subjects TEXT NOT NULL,

active BOOLEAN NOT NULL DEFAULT 1 CHECK (

active IN (0, 1)

),

stay BOOLEAN NOT NULL DEFAULT 0 CHECK (

active IN (0, 1)

)

);

"""

create\_sharing\_table = """

CREATE TABLE IF NOT EXISTS sharings (

id INTEGER PRIMARY KEY,

subjectId INTEGER NOT NULL,

sections TEXT NOT NULL,

final BOOLEAN NOT NULL DEFAULT 0 CHECK (

final IN (0, 1)

)

);

"""

create\_results\_table = """

CREATE TABLE IF NOT EXISTS results (

id INTEGER PRIMARY KEY,

content BLOB NOT NULL,

timestamp DATETIME DEFAULT CURRENT\_TIMESTAMP

);

"""

cursor.execute(create\_instructors\_table)

cursor.execute(create\_rooms\_table)

cursor.execute(create\_subjects\_table)

cursor.execute(create\_sections\_table)

cursor.execute(create\_sharing\_table)

cursor.execute(create\_results\_table)

conn.commit()

conn.close()

def getConnection():

return sqlite3.connect('gas.db')

Explanation:

The code imports sqlite3 as the author wants to create a database. The code will then create a “gas.db” which is a database file for each of the table that it creates.

The code is creating six tables, instructors, rooms, subjects, sections and sharing.

The SQLite3 code while creating each table checks if the table already exists or not. If yes then entries will be updated to the existing table otherwise a new table will be created.

The code then executed all these SQLite3 codes and stores them as a database file.isme