**DATABASE ALGORITHM**

Import sqlite3

def function 🡪 checkSetup()

conn 🡪 sqlite3.connect(‘Database fileName’)

cursor 🡪 conn.cursor()

cursor.execute("SELECT name FROM sqlite\_master WHERE type is 'table' AND  
 name is 'instructors'")  
   
 result 🡪 cursor.fetchone()

conn.close()

if result is None

then return False

else return True

def function 🡪 setup()

conn 🡪 sqlite3.connect(‘Database fileName’)

cursor 🡪 conn.cursor()

create\_instructions\_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 function 🡪 getConnction()

return sqlite3.connect(‘Database fileName’)