**Algorithm: ImportExport Handler**

def function 🡪 getCSVFile (type)

FileName 🡨 User\_Input

IF (FileName 🡪 Empty)

THEN Return 🡨 False

END IF

OPEN File in Read mode 🡨 FileName

WITHIN File:

FileContent 🡨 Content of File

Content 🡨 [ ]

FOR (Each index and row in FileContent):

IF (index 🡪 Empty and row[0] != type)

THEN Return 🡨 False

END IF

END FOR

IF (no rows of FileContent iterated)

THEN Return 🡨 False

END IF

END File

Return 🡨 Content

def function 🡪 saveAs()

FileName 🡨 User\_Input

IF (FileName 🡪 Empty)

THEN Return 🡨 False

END IF

OPEN File in Write+ mode🡨 FileName

WITHIN File:

Conn 🡨 **getConnection()**

FOR line in Conn.iterdump():

File.write('{}\n'.format(line))

END FOR

Conn 🡨 CLOSE

END File

def function 🡨 load()

FileName 🡨 User\_Input

IF (FileName 🡪 Empty)

THEN Return 🡨 False

END IF

OPEN File in Read mode 🡨 FileName

WITHIN File:

Conn 🡨 **getConnection()**

Cursor 🡨 Conn.Cursor()

tables 🡨 list( Cursor.execute ("SELECT name FROM sqlite\_master WHERE type IS 'table'"))

Cursor.executescript (';'.join(['DROP TABLE IF EXISTS {}'.format(table[0]) for table in tables]))

Cursor.executescript (File.read())

Conn 🡨 CLOSE

END File

def functions 🡨 removeTables()

Conn 🡨 **getConnection()**

Cursor 🡨 Conn.Cursor()

tables 🡨 list( Cursor.execute ("SELECT name FROM sqlite\_master WHERE type IS 'table'"))

Cursor.executescript(';'.join(['DROP TABLE IF EXISTS {}'.format(table[0]) for table in tables]))

Conn 🡨 CLOSE