

Python-MySQL Documentation

Generated by Doxygen 1.8.11

Contents

1	File Index	1
1.1	File List	1
2	File Documentation	3
2.1	Python API/DataHandler.py File Reference	3
2.2	Python API/DBHandler.py File Reference	3
2.2.1	Function Documentation	3
2.2.1.1	dbConnect()	3
2.2.1.2	dbInsert(location, inOrOut)	3
2.2.1.3	fileToDB()	4
2.2.1.4	insert(dataMode, location, inOrOut)	4
2.3	Python API/FileHander.py File Reference	4
2.3.1	Function Documentation	4
2.3.1.1	fileIsEmpty()	4
2.3.1.2	insertToFile(location, inOrOut)	4
	Index	5

Chapter 1

File Index

1.1 File List

Here is a list of all documented files with brief descriptions:

Python API/ DataHandler.py	3
Python API/ DBHandler.py	3
Python API/ FileHander.py	4

Chapter 2

File Documentation

2.1 Python API/DataHandler.py File Reference

Variables

- int **DataHandler.dataMode** = 0
- **DataHandler.ser** = serial.Serial('COM3', 9600)
- string **DataHandler.location** = "lot35"
- **DataHandler.data** = ser.readline()[:-2]

2.2 Python API/DBHandler.py File Reference

Functions

- def **DBHandler.dbConnect** ()
- def **DBHandler.dbInsert** (location, inOrOut)
- def **DBHandler.fileToDB** ()
- def **DBHandler.insert** (dataMode, location, inOrOut)

2.2.1 Function Documentation

2.2.1.1 def DBHandler.dbConnect ()

This function attempts to establish a database connection using the specified parameters and throws an exception if needed.

:return: Boolean - specifies whether the connection was successful or not

2.2.1.2 def DBHandler.dbInsert (location, inOrOut)

This functions inserts the data entry that is read in from the Arduino into the MySQL database. The entry contains whether the car is entering or exiting as well as the lot number.

:param inOrOut: Boolean - whether the car entered or exited

:param location: String - specifies which lot we are recording for

:return: Boolean - specifies whether or not the insert was successful or not

2.2.1.3 def DBHandler.fileToDB ()

This function reads in the data from the data file that is used to store data if a DB connection could not be established and inserts it into the database.
:return: void

2.2.1.4 def DBHandler.insert (*dataMode*, *location*, *inOrOut*)

This function handles inserting the data read from the Arduino into either the file or the database depending on whether a network connection can be established.
:param dataMode: Boolean/integer - Whether there is a network connection or not
:param location: String - specifies which lot we are recording for
:param inOrOut: Boolean - whether the car entered or exited
:return: void

2.3 Python API/FileHandler.py File Reference

Functions

- def [FileHandler.insertToFile](#) (location, inOrOut)
- def [FileHandler.filesEmpty](#) ()

2.3.1 Function Documentation

2.3.1.1 def FileHandler.filesEmpty ()

This function checks the data file to see if it is empty.
It is used by the fileToDB function to send file contents to the DB if the file is not empty.
:return: Boolean - specifies whether or not the file has more to read from

2.3.1.2 def FileHandler.insertToFile (*location*, *inOrOut*)

This function inserts data read from the Arduino into a text file. Each data entry is a line in the text file that contains the entry time, location, and type.
:param location:
:param inOrOut:
:return: void

Index

DBHandler.py

 dbConnect, [3](#)

 dbInsert, [3](#)

 fileToDB, [3](#)

 insert, [4](#)

dbConnect

 DBHandler.py, [3](#)

dbInsert

 DBHandler.py, [3](#)

FileHandler.py

 fileIsEmpty, [4](#)

 insertToFile, [4](#)

fileIsEmpty

 FileHandler.py, [4](#)

fileToDB

 DBHandler.py, [3](#)

insert

 DBHandler.py, [4](#)

insertToFile

 FileHandler.py, [4](#)

Python API/DBHandler.py, [3](#)

Python API/DataHandler.py, [3](#)

Python API/FileHandler.py, [4](#)