

Application documentation

content

1	Release Notes	1
2	Background and aim of the application.....	2

1 Release Notes

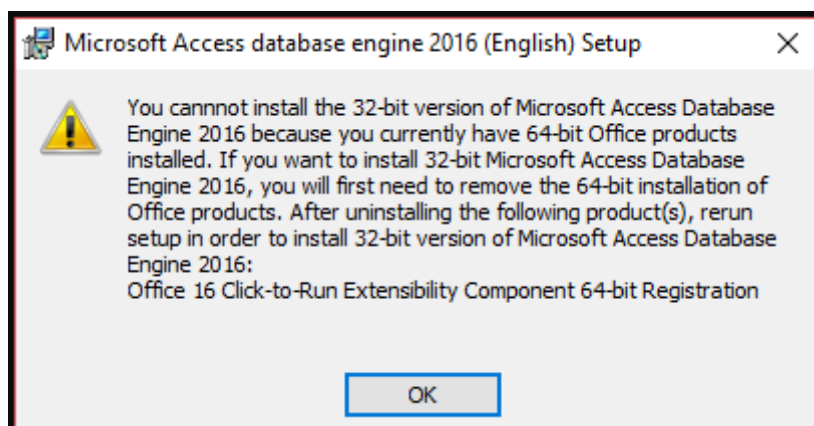
Version	Date	Editor	Comment
V0	25.10.2019	Hachem SFAR	Initial creation

Storage location: <path>

2 Installation:

The first thing to go to Driver and to install “AccessDatabaseEngine_X64.exe”

Sometime, when you double click the file, you get this error message.



In that case, the alternative way for installation is to Open the “Command prompt” in the file folder:



- After that, to launch the program, you should open “dist” then converter.exe

Nom	Modifié le	Type	Taille
converter	30/10/2019 16:15	Application	27 983 Ko
Params	30/10/2019 16:21	Document texte	1 Ko

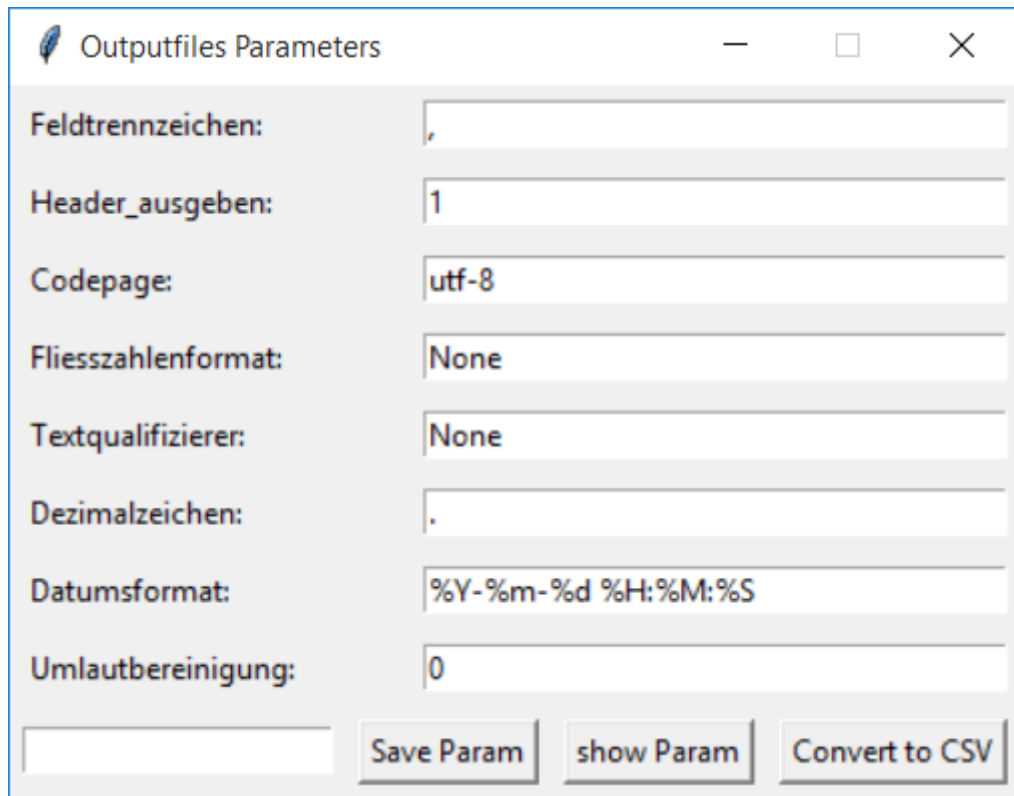
3 Background and aim of the application

The application will enable you to convert the company datasets in different format(.xlsx/.parquet/.accdb) to .CSV format

In addition, you can choose conversion parameters:

- The Field delimiter for the output file (Feldtrennzeichen: String)
- If the file has a header or not (Header_ausgeben: Boolean)
- The encoding of the output file (Codepage: String)
- The string format for floating point numbers (Fließzahlenformat: String)
- Character used to quote fields (Textqualifizierer: String of length 1)
- Character recognized as decimal separator (Dezimalzeichen: String)
- Format string for datetime objects (Datumsformat: String)

- If we want to transform the german letter or not (Umlautbereinigung: Boolean)



Outputfiles Parameters

Feldtrennzeichen: ,

Header_ausgeben: 1

Codepage: utf-8

Fließzahlenformat: None

Textqualifizierer: None

Dezimalzeichen: .

Datumsformat: %Y-%m-%d %H:%M:%S

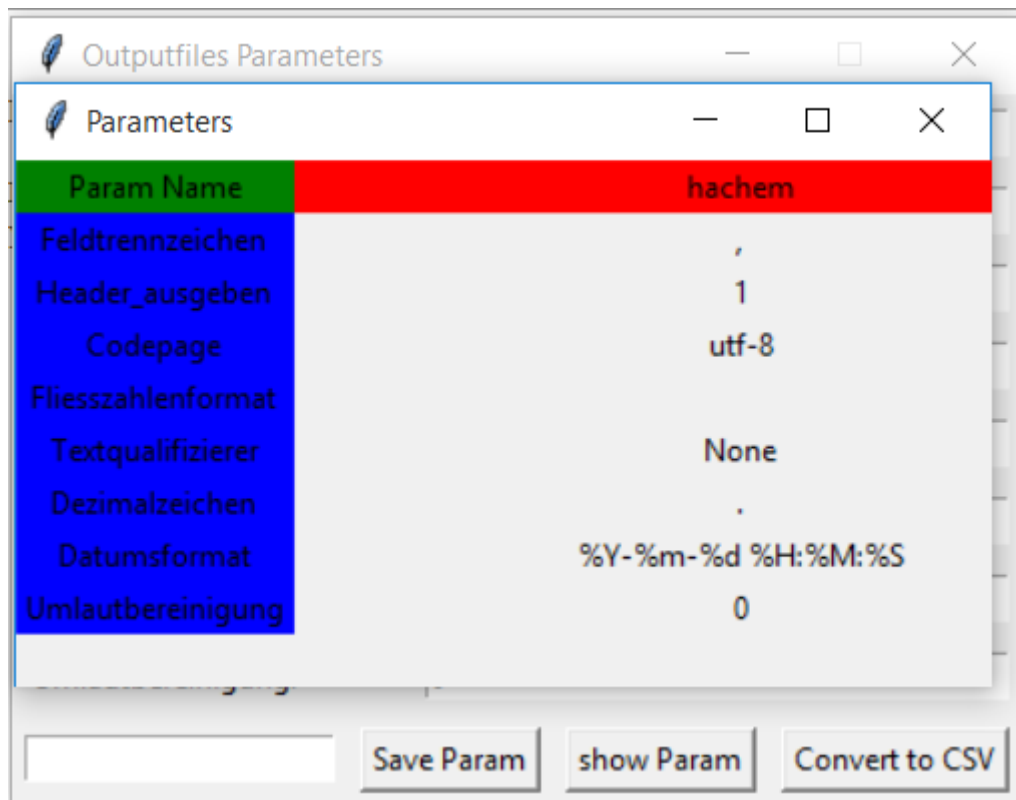
Umlautbereinigung: 0

Save Param show Param Convert to CSV

After that, you can save these parameters for later use by giving it a name and then clicking save Param (it's be saved in .JSON format in Params.txt)

If the param name contain some upcase letters. They will be transformed on lowercase and also space at the beginning and at the end will be deleted

Then, you can see all the Parameters saved by clicking “show Param”.



➔ Now to convert the datasets:

- 1- Put the files in “Input directory”
- 2- Open files2convert.xlsx

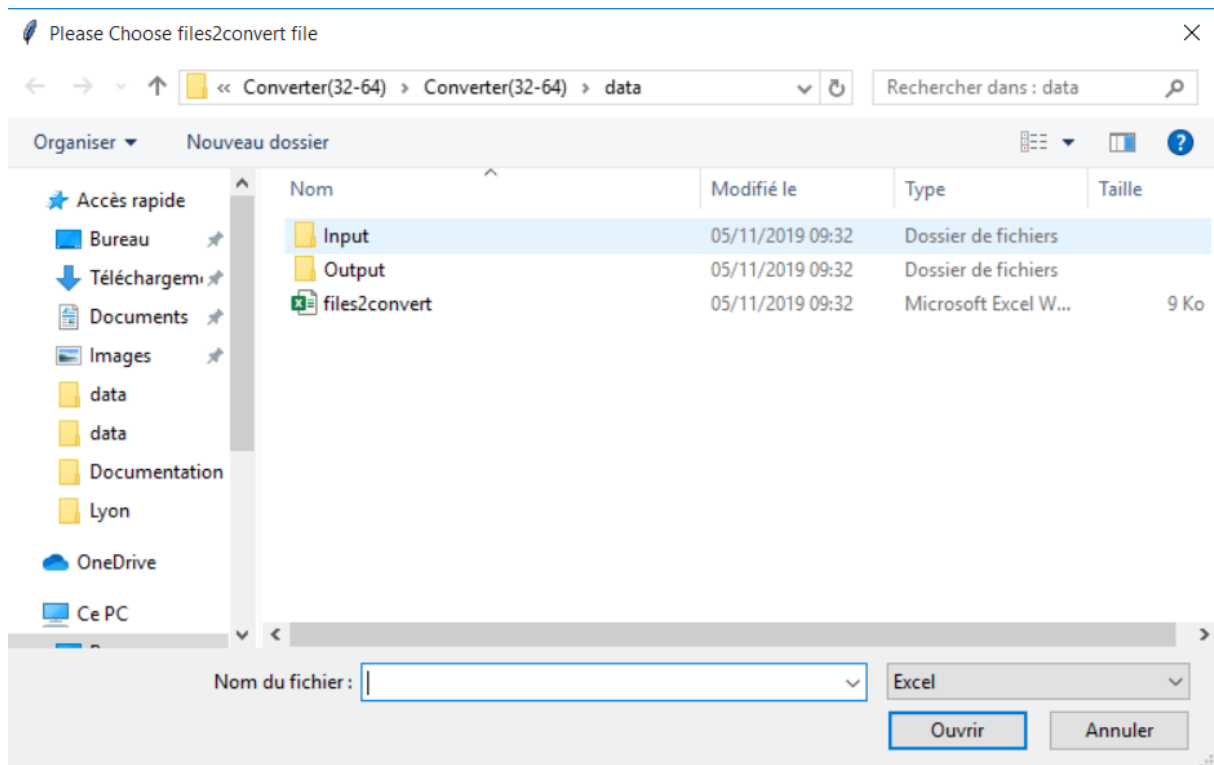
InputFilename	TableSheetName	OutputFilename	param
excel02.xlsx	Tabelle2222	output02	E.ON
excel02.xlsx	Tabelle2	output03	E.ON
excel1.xlsx	Tabelle2	output06	sfar
TESTdb.accdb	ZCC_PFL_STATUS	output10	hachem
holiday.parquet		output11	hachem

3- Fill the columns:

- InputFilename: Filename that you want to convert(.xlsx/.paquet/.accdb)
- TableSheetName: sheetname/tablename (for xlsx and accdb files)
- OutputFileName: Converted Files(.csv)
- Param: Parameter that you want to use

Note: during converting, for each row, if the program find that one of columns is wrong (inputfilename, tablesheetname, param).. It passes it and moves to the next row.

- 4- Click “Convert to CSV”
- 5- Select files2convert.xlsx



- 6- Then, you will find the converted files in “Output directory” and “snowflake_SQL.xlsx” file, you find for each converted file, the SQL request to insert the files in “snowflake”

➔ To launch the application, you just need to double click on “converter.exe” and params.txt must always in the same folder of the application.

If not, the application will not run.



converter



Params

➔ Parameters:

- Feldtrennzeichen: (Seperator)

It used to specify the separator between columns names and values in the output CSV file.

For examples:

Sep=;

ID;NAME;AGE

"23434";"Alice";"24"

Sep=,

ID,NAME,AGE

"23434","Alice","24"

Sep=

ID/NAME/AGE

"23434"/"Alice"/"24"

- Datumsformat:

	Meaning	Example
%a	Weekday as locale's abbreviated name.	Sun, Mon, ..., Sat (en_US);
%A	Weekday as locale's full name.	Sunday, Monday, ..., Saturday (en_US)
%w	Weekday as a decimal number, where 0 is Sunday and 6 is Saturday.	0, 1, ..., 6
%d	Day of the month as a zero-padded decimal number.	01, 02, ..., 31
%b	Month as locale's abbreviated name.	Jan, Feb, ..., Dec (en_US)
%B	Month as locale's full name.	January, February, ..., December
%m	Month as a zero-padded decimal number.	01, 02, ..., 12
%y	Year without century as a zero-padded decimal number.	00, 01, ..., 99
%Y	Year with century as a decimal number.	1970, 1988, 2001, 2013
%H	Hour (24-hour clock) as a zero-padded decimal number.	00, 01, ..., 23
%M	Minute as a zero-padded decimal number.	00, 01, ..., 59
%S	Second as a zero-padded decimal number.	00, 01, ..., 59
%p	Locale's equivalent of either AM or PM.	AM, PM (en_US);
%j	Day of the year as a zero-padded decimal number.	001, 002, ..., 366
%U	Week number of the year (Sunday as the first day of the week) as a zero padded decimal number. All days in a new year preceding the first Sunday are considered to be in week 0.	00, 01, ..., 53
%W	Week number of the year (Monday as the first day of the week) as a decimal number. All days in a new year preceding the first Monday are considered to be in week 0.	00, 01, ..., 53

For example:

```
import datetime

Now=datetime.datetime.now()
print(Now)
#2019-10-29 11:00:44.206279

t = Now.strftime("%H:%M:%S")
print("time:", t)
#time: 11:00:44
|
s1 = Now.strftime("%m/%d/%Y, %H:%M:%S")
print("s1:", s1)
#s1: 10/29/2019, 11:00:44

s2 = Now.strftime("%a %W %B %d-%m-%Y, %H:%M")
print("s2:", s2)
#s2: Tue 43 October 29-10-2019, 11:00
```

- Umlautbereinigung:

When it is equal to 0 or False:

There is no transformation of german characters

Else there will be a transformation.

Transformations:

'ö' → 'oe'
 'Ö' → 'Oe'
 'ü' → 'ue'
 'Ü' → 'Ue'
 'ä' → 'ae'
 'Ä' → 'Ae'
 'ß' → 'ss'

For examples:

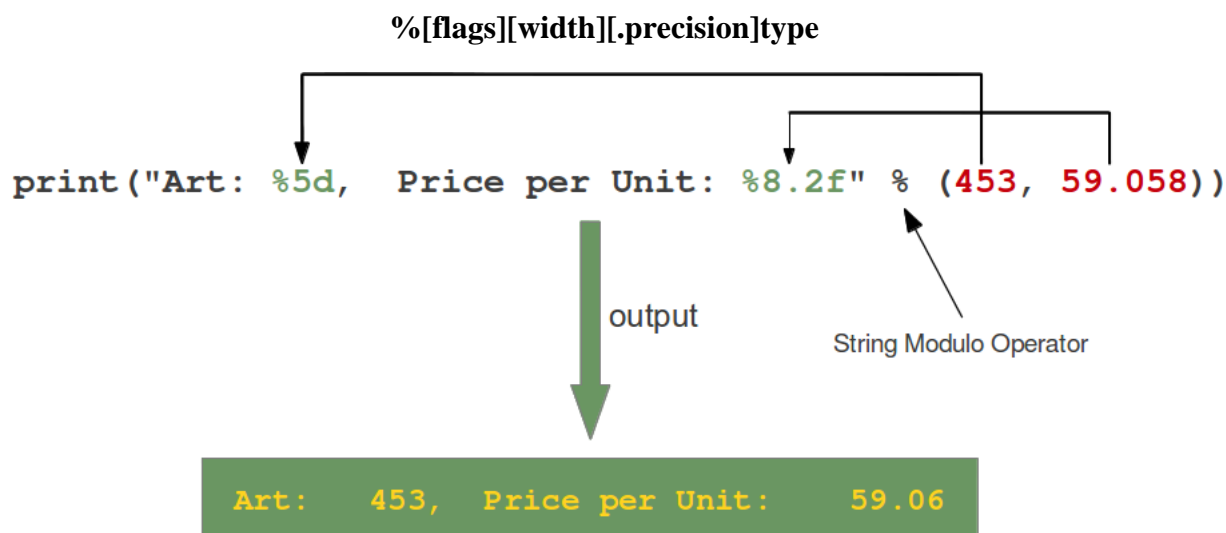
Without char transformation:

```
AB_Preise_ID,VNBG_Nr,Tarifgebiet,guelting_start,Id,Standard,Zaehler_ID,AB_ID,Einheit_ID,Ääö öüü,Hardware_ID,Dienstleistung_ID|
688,91110,9111001,2007-01-01,2,1,1000,1,1,ä,ä,0
689,91110,9111001,2007-10-01,1,1,1000,1,1,äö öüü,0,0
690,91110,9111001,2008-04-01,1,1,1000,1,1,äöüü,0,0
691,91110,9111001,2009-01-01,1,1,1000,1,1,äöüü,0,0
692,91110,9111001,2010-01-01,1,1,1000,1,1,äöüü,0,0
693,91110,9111001,2011-01-01,1,1,1000,1,1,äöüü,0,0
694,91110,9111001,2013-01-01,3,1,1000,1,1,äöüü,0,0
695,91110,9111001,2012-01-01,3,1,1000,1,1,äöüü,0,0
3858,582850,58285001,2007-10-01,1,1,1000,1,1,äöüü,0,0
3859,582850,58285001,2007-10-01,2,1,1000,1,1,äöüü,0,0
3860,582850,58285001,2009-01-01,1,1,1000,1,1,äöüü,0,0
3861,582850,58285001,2009-01-01,2,1,1000,1,1,äöüü,0,0
3862,582850,58285001,2010-01-01,1,1,1000,1,1,äöüü,0,0
3863,582850,58285001,2010-01-01,2,1,1000,1,1,äöüü,0,0
3864,582850,58285001,2011-01-01,1,1,1000,1,1,äöüü,0,0
3865,582850,58285001,2011-01-01,2,1,1000,1,1,äöüü,0,0
3866,582850,58285001,2012-01-01,1,1,1000,1,1,äöüü,0,0
3867,582850,58285001,2012-01-01,2,1,1000,1,1,äöüü,0,0
3868,582850,58285001,2012-01-01,3,1,1000,1,1,äöüü,0,0
```

With char transformations:

```
AB_Preise_ID;VNRG_Nr;Tarifgebiet;gueltig_seit;ID;Standard;Zaehler_ID;AB_ID;Einheit_ID;AeaeOe_oeueUe;Hardware_ID;Dienstleistung_ID
688;91110;9111001;2007-01-01;2;1;1000;1;1;ae;0;0
689;91110;9111001;2007-10-01;1;1;1000;1;1;aeOe_oeueUe;0;0
690;91110;9111001;2008-04-01;1;1;1000;1;1;aeOeoeueUe;0;0
691;91110;9111001;2009-01-01;1;1;1000;1;1;aeOeoeueUe;0;0
692;91110;9111001;2010-01-01;1;1;1000;1;1;aeOeoeueUe;0;0
693;91110;9111001;2011-01-01;1;1;1000;1;1;aeOeoeueUe;0;0
694;91110;9111001;2013-01-01;3;1;1000;1;1;aeOeoeueUe;0;0
695;91110;9111001;2012-01-01;3;1;1000;1;1;aeOeoeueUe;0;0
3858;582850;58285001;2007-10-01;1;1;1000;1;1;aeOeoeueUe;0;0
3859;582850;58285001;2007-10-01;2;1;1000;1;1;aeOeoeueUe;0;0
3860;582850;58285001;2009-01-01;1;1;1000;1;1;aeOeoeueUe;0;0
3861;582850;58285001;2009-01-01;2;1;1000;1;1;aeOeoeueUe;0;0
3862;582850;58285001;2010-01-01;1;1;1000;1;1;aeOeoeueUe;0;0
3863;582850;58285001;2010-01-01;2;1;1000;1;1;aeOeoeueUe;0;0
3864;582850;58285001;2011-01-01;1;1;1000;1;1;aeOeoeueUe;0;0
3865;582850;58285001;2011-01-01;2;1;1000;1;1;aeOeoeueUe;0;0
3866;582850;58285001;2012-01-01;1;1;1000;1;1;aeOeoeueUe;0;0
3867;582850;58285001;2012-01-01;2;1;1000;1;1;aeOeoeueUe;0;0
3868;582850;58285001;2012-01-01;3;1;1000;1;1;aeOeoeueUe;0;0
```

- Fliesszahlenformat:



Let's have a look at the placeholders in our example. The second one "%8.2f" is a format description for a float number. Like other placeholders, it is introduced with the "%" character. This is followed by the total number of digits the string should contain. This number includes the decimal point and all the digits, i.e. before and after the decimal point. Our float number 59.058 has to be formatted with 8 characters. The decimal part of the number or the precision is set to 2, i.e. the number following the "." in our placeholder. Finally, the last character "f" of our placeholder stands for "float".

If you look at the output, you will notice that the 3 decimal digits have been rounded. Furthermore, the number has been preceded in the output with 3 leading blanks.

The first placeholder "%5d" is used for the first component of our tuple, i.e. the integer 453. The number will be printed with 5 characters. As 453 consists only of 3 digits, the output is padded with 2 leading blanks.

Type:

Conversion	Meaning
d	Signed integer decimal.
i	Signed integer decimal.
o	Unsigned octal.
u	Obsolete and equivalent to 'd', i.e. signed integer decimal.
x	Unsigned hexadecimal (lowercase).
X	Unsigned hexadecimal (uppercase).
e	Floating point exponential format (lowercase).
E	Floating point exponential format (uppercase).
f	Floating point decimal format.
F	Floating point decimal format.
g	Same as "e" if exponent is greater than -4 or less than precision, "f" otherwise.
G	Same as "E" if exponent is greater than -4 or less than precision, "F" otherwise.
c	Single character (accepts integer or single character string).
r	String (converts any python object using <code>repr()</code>).
s	String (converts any python object using <code>str()</code>).
%	No argument is converted, results in a "%" character in the result.

Flag:

Flag	Meaning
#	Used with o, x or X specifiers the value is preceded with 0, 0o, 0O, 0x or 0X respectively.
0	The conversion result will be zero padded for numeric values.
-	The converted value is left adjusted
	If no sign (minus sign e.g.) is going to be written, a blank space is inserted before the value.
+	A sign character ("+" or "-") will precede the conversion (overrides a "space" flag).

- Dezimalzeichen: Character recognized as decimal separator

For examples:

➔ Dezimalzeichen=.

index,column1

1,2.1

3,4.0

➔ Dezimalzeichen=,

index,column1

1,"2,1"

3,"4,0"

➔ Dezimalzeichen=\$

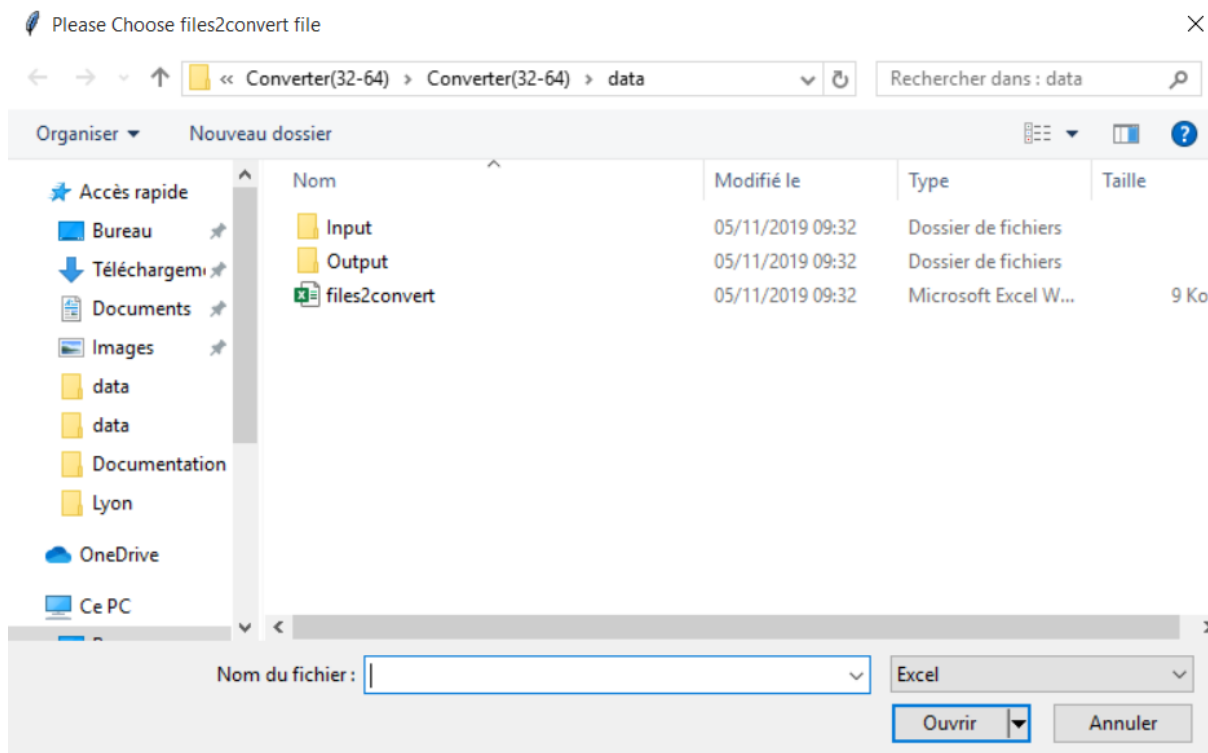
index,column1

1,"2\$1"

3,"4\$0"

➔ Console messages explanation:

- When you click “Click2Convert”, then you close the window without selecting a file.



You will get this message on the console: “please choose a the convert2csv file”

- When in the file2convert file when we have duplication of “OutputFilename” columns like in the photo:

OutputFilename
output02
output02
output04

In that case, the program abort and you get “conversion aborted” message.

```
Start Conversion: 2019-10-30 09:14:17.955211
Selected: C:/Users/Hachem/Desktop/Script/data/a.xlsx
The Output File Named output02 Exists 2 Times
Conversion Aborted
```

if the Outputfilename is blank... the Output will take the same name of the inputfilename.

For example, if we have Excel1.xlsx in the input then we find Excel1.csv

- In the TableSheetName column, when we put a sheetname that doesn’t exist in the “inputfile”.

TableSheetName
Tabelle11
ZCC_SPARTE

You get this error and the program will move to the next row.

```
No sheet named <'Tabelle11'>
```

- When we use a parameter that not saved in params.txt.

For example, EON, in that case you get this message “Parameter eon not found”.

```
C:/Users/Hachem/Desktop/Script/data/Input/TESTdb.accdk
Parameter eon not found
C:/Users/Hachem/Desktop/Script/data/Input/excell.xlsx
End Conversion: 2019-10-30 09:27:05.747147
```

And the program moves to convert the files in the next rows.

- In our program, we can put the beginning of the file name instead of writing the fullname.

For example:

- we have 3 files: Excell.xlsx EON.xlsx October.xlsx and we want to convert Excell so we can write Ex so the program will understand that you are searching for excell and program will convert it.
- When, we write E. the program will search for files which start with “E” so it returns EON and Excell. In that case it will not understand which one you want to convert and moves to the next row.

In that case, the program will show this message on the console:

“For Than One File Starts with E Was Found”

- When, we write H . the program will search for files which start with “H” so it finds nothing. In that it moves to the next row and show this message.

“File Starts with H Not Found”