

setenv('modedir', 'D:\_variants_private')	Set environmental	Setzt eine Umgebungsvariable
getenv()	Get environmental	Auslesen einer Umgebungsvariable
		Model Generierung
createRealSg('new', 'old')	R:\work\updates\21KW05_21KW13\FOP_C AN_FD	Erstellt die Delta Tabellen
F9 markierten Befehl ausführen		
TASKKILL /f /IM matlab.exe		
matlab -r -nodesktop "matlab.engine.shareEngine('CustomDSpaceAutomation Engine')"		
Witch nicht verwenden		

PEP: [PEP 0 – Index of Python Enhancement Proposals \(PEPs\) | peps.python.org](https://peps.python.org/)

Module : [Python Module Index — Python 3.10.4 documentation](https://docs.python.org/3.10.4/)

Data model: [3. Data model — Python 3.10.4 documentation](https://docs.python.org/3.10.4/)

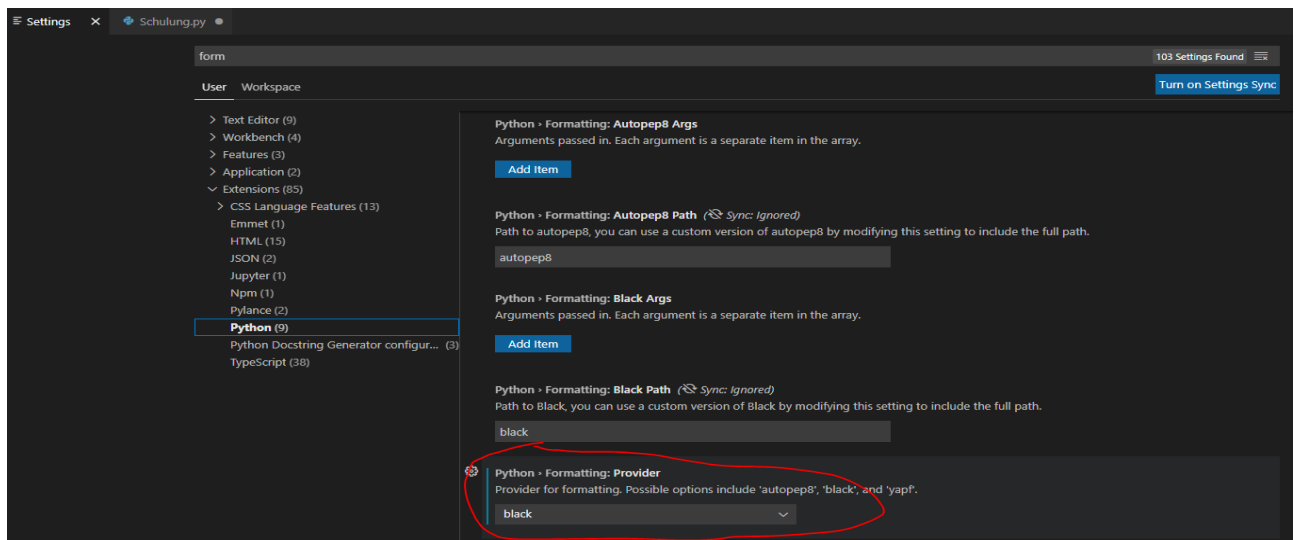
**pip**

**py -3.9 -m pip install panda**

CfdApi.MatlabInstance.bdclose('all', nargout=0)	
CfdApi.MatlabInstance.quit()	
pip3 install torch torchvision torchaudio	
C:\Users\kthlman\AppData\Local\Programs\Python\Python39\python.exe -m pip install --upgrade pip	

**Black:**

**Shift + Alt + F**



**py -3.9 -m black Schulung.py**

py -3.9 -m black --line-length 10000 LoadingBar.py

py -3.9 -m black --line-length 250 R:\CFD\_Automation\Tools\EthernetConfigChecker\EthernetConfigChecker.py

py -3.9 C:\Users\kthalman\AppData\Local\Programs\Python\Python39\Lib\site-packages\pre\_commit install

<code>DL_Tx.sort(key=lambda x: str(x.split('/')[1]))</code>	
<code>My_List.insert(0, 10)</code>	Einfügen an der Oten stelle von 10
<code>My_List.extend([4,5,6])</code>	Anhängen von mehreren Sachen
<code>List.pop(1)</code> or <code>del List[1]</code>	Löschen des 1.Elements
<code>dict_test = {"Block": "block"}</code> Wenn man nicht weiß, ob das dict einen Block Key hat, soll man <code>dict_test.get("Block1", "123")</code> verwenden. Das zweite Argument wird rausgegeben, wenn <code>Block1</code> nicht gefunden wurde	Dict -> wenn man nicht weiß, Ob ein Eintrag vorhanden ist.
<code>with open(text.txt, r+) as file:</code> <code>file.tell()</code> <code>file.seek(40)</code> <ul style="list-style-type: none"> <li>• r: Lesen</li> <li>• w: Schreiben. Der Inhalt der Datei wird gelöscht</li> <li>• a: Schreiben. Die neuen Daten werden ans Ende der Datei geschrieben.</li> <li>• x: Schreiben. Die Datei darf nicht existieren.</li> <li>• r+: Lesen und Schreiben.</li> </ul>	The tell() method returns the current file position in a file stream. Tip: You can change the current file position with the <a href="#">seek()</a> method.
<code>glob.glob(os.path.join(self.inipath, '*', '*.ini'), recursive=True)</code> <code>glob.iglob(os.path.join(self.inipath, '*.ini'), recursive=False)</code>	Suchen von Dateien mit glob und iglob

<code>os.path.dirname(os.getcwd())</code>	Name des aktuellen Skripts
<code>Os.path.abspath(__filename__) =&gt; __filename__</code>	
<code>.removesuffix('xyz')</code> <code>.removeprefix('xyz')</code>	Ersetzen am Ende Ersetzen am Anfang
<code>for i in range(100000):</code> <code>print(f'\r{i}', end='')</code>	Ausgeben von nur einer Zeile
<code>Os.path.join(*['a', 'c']) = 'a', 'c'</code>	
<code>from pathlib import Path</code>  <code>rootpath = Path(__file__).parents[3]</code> <code># add folder "CFD_Automation" to search path</code> <code>if not str(rootpath / "CFD_Automation") in sys.path:</code> <code>sys.path.append(str(rootpath / "CFD_Automation"))</code>  <code>from CheckSimulinkPorts import CheckSimulinkPortsForEthernet</code>	Imports von anderen Modulen
Befüllen von einer list in den values mit einem gleichen key <code>com = {}</code> <code>com.setdefault(services, set()).add(item)</code>	Erstellen eines Dicts mit gleichen Keys und adden von Values
<code>import math</code> <code>[res for x in range(10) if (res := math.sin(x)) &gt;= 0]</code>	Walross-Operator
Xml Paser <code>com_services = {ele.attrib['Name'] for ele in root.findall(".*[@PduType='COM']")}</code>  <code>root.find(f"./*[@CLUSTER-ITEM='{pattern_psi.search(eth)[1]}']/PASS-THROUGH")</code>  <code>root_ethcfg.find(f'./*[@CLUSTER-ITEM="{trc_import}"]/INTERFACE-GENERATION/ACTIVATE-TRC["false"]')</code>  <code>if next(iter(ele)).find('ACTIVATE-TRC') != None:</code> <code>if ele.find('./INTERFACE-GENERATION/ACTIVATE-TRC') != None</code>	Suchen von Eigenschaften in einer XML
<code>real_path.partition("_variants_private")</code>	Wie split ohne das element zu löschen
<code>cluster_name = {'Ethernet4'}</code> <code>cluster_name.pop() = 'Ethernet4'</code>	
Schreiben einer XML <code>xml_str = minidom.parseString(ET.tostring(ethct_conf)).toprettyxml(indent=" ", encoding="UTF-8")</code>	

pathlib Path	
<code>vpp = Path('D:\_variants_private')</code>	
<code>subpathlist = [A,B]</code>  <code>arxml_database_path = vpp.joinpath("databases", *subpathlist, eth_config)</code>	

<pre>arxml_database_filepath_list = list(arxml_database_path.glob("*.arxml")) self.rootpath.joinpath("Model", "RBS", "Ethernet").glob("Ethernet*")</pre>	
--	--

<b>Flake8</b> Str+shift + p -> Python: Select Linter -> flake8	<b>Setup.cfd</b>  <pre>[flake8] extend-ignore = E501 LN001 LN002 E402</pre>
<b>Black shift+Alt + F</b>	<b>pyproject.toml</b>  <pre>[tool.black] line-length = 350</pre>
<pre>py -3.9 -m flake8 --count xxx.py py -3.9 -m flake8 --output-file=output.txt dir/</pre>	<b>Anzeigen der Fehler</b> Printer aller Fehler in eine Datei
<b>Decode Umlaute</b>	<pre>latin1.decode("iso-8859-1")</pre>

```
from win32com.client import Dispatch
from win32com.client import DispatchEx
from win32com.client import GetActiveObject

try:
    self.excelActiveInstance = GetActiveObject("Excel.Application") # try to get active excel
    instance
except:
    self.excel = Dispatch("Excel.Application") # open new excel instance
else:
    self.excel = DispatchEx("Excel.Application") # open new excel instance

if self.excelActiveInstance:
    for Workbook in self.excelActiveInstance.Workbooks:
        if Workbook.FullName == ExcelFile:
            self.CDApi.Logger.warn(
                msg := f'The Excel file {Workbook.Name} is currently opened in Excel.
Please close the file and restart the script'
            )
            raise RuntimeError(msg)
```

## Python Pathlib Cheatsheet

**f.parent**

**f.anchor** → posix `/media/chris/KINGSTON/data_analysis/Scorecard_Raw_Data/MERGED1996_97_PP.csv`

**f.name** → `MERGED1996_97_PP.csv`

**f.stem** → `MERGED1996_97_PP`

**f.suffix** → `.csv`

**f.absolute()**

Windows → `D:/data_analysis/Scorecard_Raw_Data/MERGED1996_97_PP.csv`

**f.drive** → `D`

posix `file:///media/chris/KINGSTON/data_analysis/Scorecard_Raw_Data/MERGED1996_97_PP.csv`

**f.as\_uri()**

Windows `file:///D:/data_analysis/Scorecard_Raw_Data/MERGED1996_97_PP.csv`

**list(f.parents)**

```
[PosixPath('/media/chris/KINGSTON/data_analysis/Scorecard_Raw_Data'),  
PosixPath('/media/chris/KINGSTON/data_analysis'),  
PosixPath('/media/chris/KINGSTON'),  
PosixPath('/media/chris'),  
PosixPath('/media'),  
PosixPath('/')]
```

**f.parts**

```
('/', 'media', 'chris', 'KINGSTON', 'data_analysis', 'Scorecard_Raw_Data', 'MERGED1996_97_PP.csv')
```

**Path('test.tar.gz').suffixes**

```
[ '.tar', '.gz' ]
```