

NEUE MÖGLICHKEITEN DER AUTOMATISIERUNG

OPUS & Python

Dr. Christian Scheibe
20.06.2023

Agenda

Was gab es schon?

Externe Kommunikation via http

Was ist Python?

OPUS API

Anwendungsbeispiele

OPUS Plugin

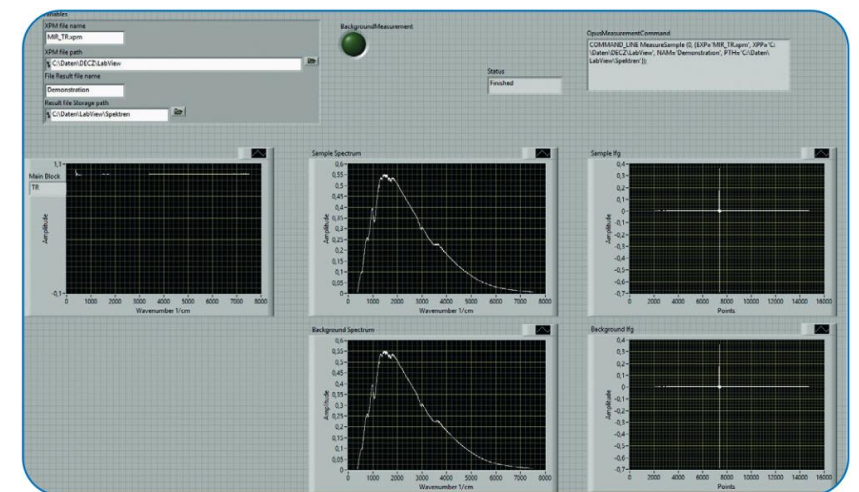
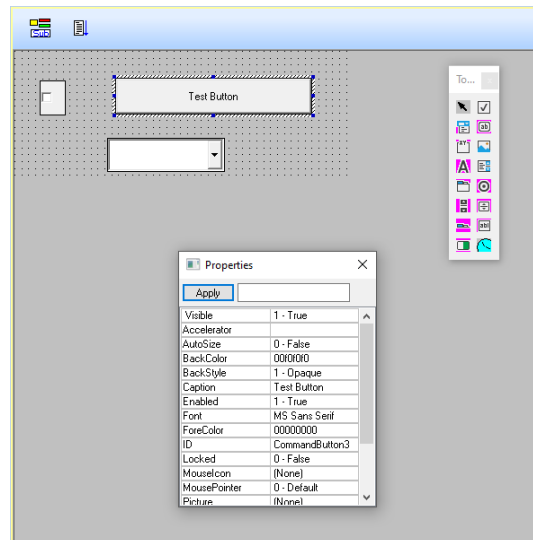
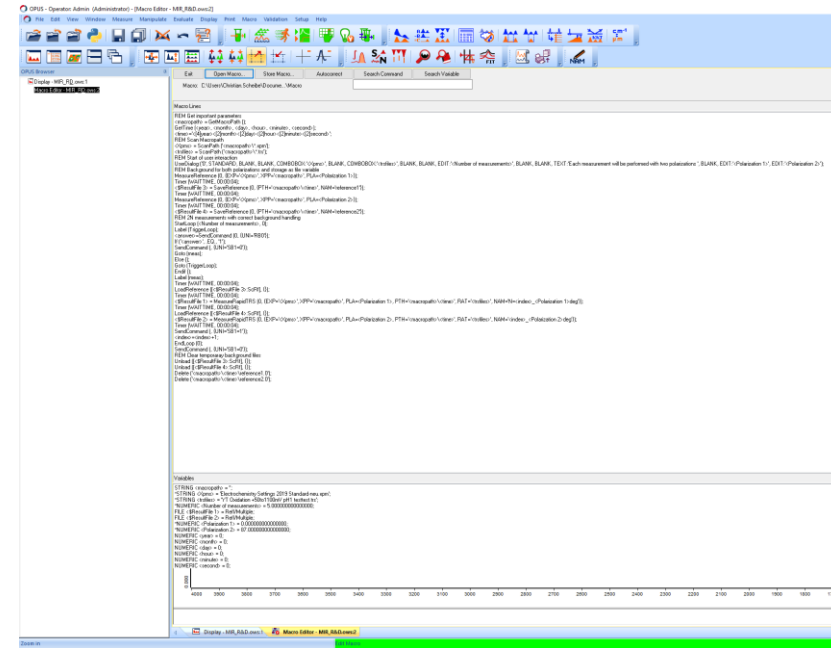
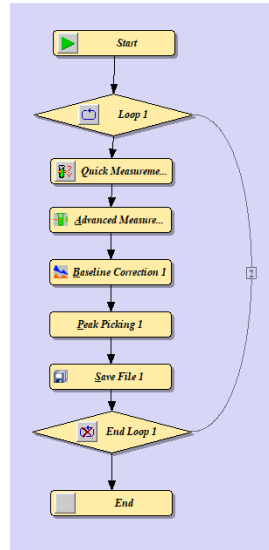
OPUS Python-Funktion

OPUS & PYTHON: NEUE MÖGLICHKEITEN DER AUTOMATISIERUNG

Was gab es schon?

Was gab es schon?

- Grafisches Makro
- OPUS Makro
- VB Skript
- LabView



OPUS & PYTHON: NEUE MÖGLICHKEITEN DER AUTOMATISIERUNG

Was ist Python?

Was ist Python?



- Interpretierte, objektorientierte, höhere Programmiersprache
- Ursprünglich 1991 von Guido van Rossum entwickelt
- Plattformunabhängig (Windows, Linux, MacOS)
- Lizenzfrei
- Große Standardbibliothek, viele Module
- Möglichkeit Module aufzurufen, die in anderen Programmiersprachen geschrieben wurden
- Standardmäßig in aktuellen OPUS-Versionen enthalten

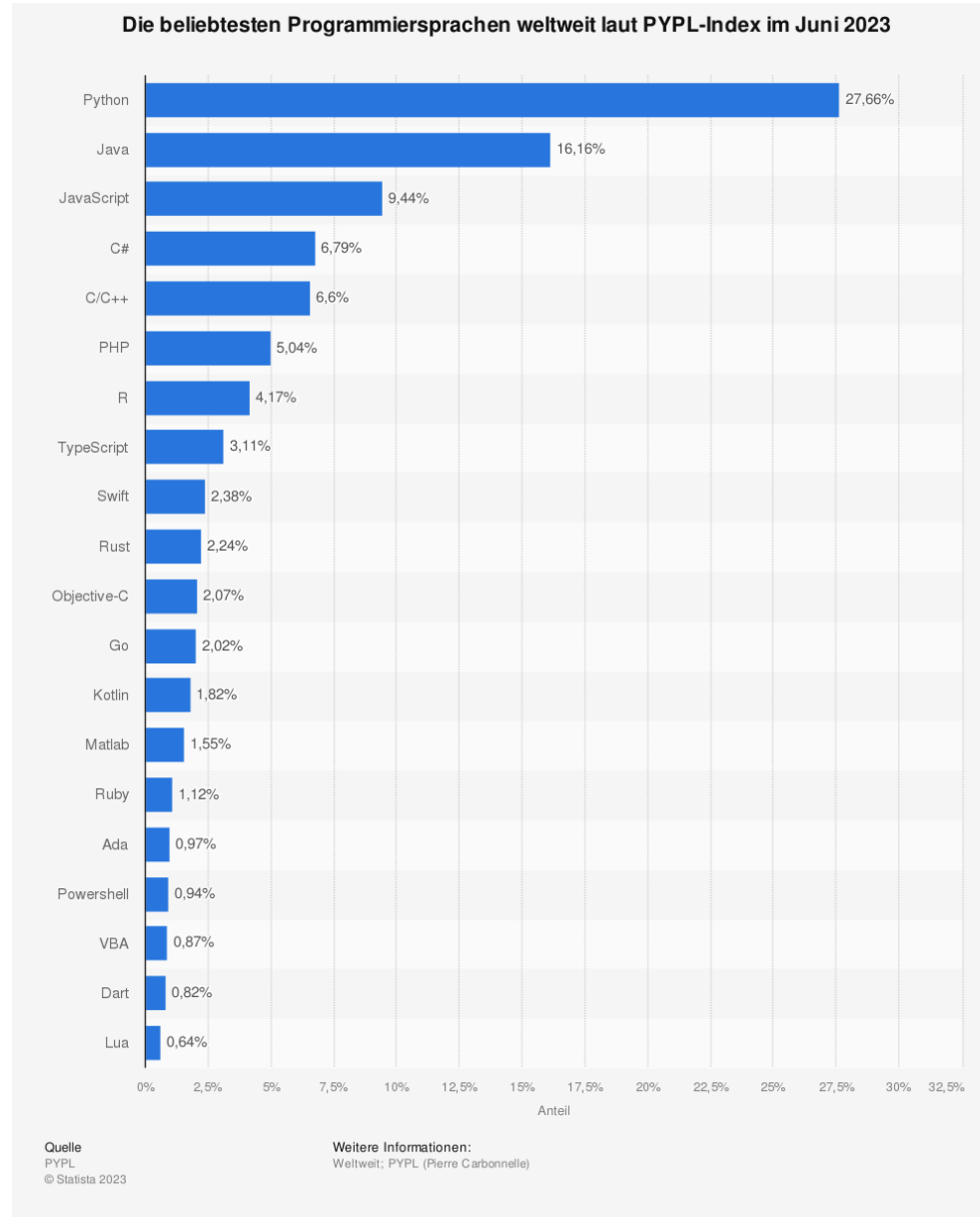
Was ist Python?



Anwendungsgebiete:

- Web Entwicklung (serverseitig)
- Wissenschaft:
 - Numerische Berechnungen
 - Grafische Darstellung
 - Machine Learning
- Lehre
- Grafische Benutzeroberflächen

Warum Python?



Region

Weltweit

Erhebungszeitraum

Juni 2023

Hinweise und Anmerkungen

Die Quelle macht folgende Anmerkung zum Index: "The PYPL PopularitY of Programming Language Index is created by analyzing how often language tutorials are searched on Google: the more a language tutorial is searched, the more popular the language is assumed to be. It is a leading indicator. The raw data comes from Google Trends".

OPUS & PYTHON: NEUE MÖGLICHKEITEN DER AUTOMATISIERUNG

Anwendungsbeispiele

Anwendungsbeispiele

A513 - Variable angle reflection accessory

File Info

A513 - Variable angle reflection accessory

Single Measurement | Measure Angle List | Measure multiple XPM

Select xpm-file

Storage Path

Mirror Control

Select angles for mirror arms: Set same angle for both mirror arms

Select angle of incidence: 45 | Select angle for reflected beam: 45

Go to Load Position | Go to Init Position | Go to selected angles

Polarizer Control

Polarizer Settings: No polarizer detected

Polarizer Pos 1: 0 | Polarizer Pos 2: 0

Set Pol 1 | Set Pol 2

Measure Reference | Measure Sample | sample name:

Current status: Idle

A500 - Automatic Sample Changer

File Info

A500 - Automatic Sample Changer

Single Measurement | Measurement Series

Select Positions to be measured

Position 1: Position 9:

Position 2: Position 10:

Position 3: Position 11:

Position 4: Position 12:

Position 5: Position 13:

Position 6: Position 14:

Position 7: Position 15:

Position 8:

Measurement parameters

Select XPM

Storage Path

Measure Background | Measure Samples | Number of repeats: 1

Current Status: Idle

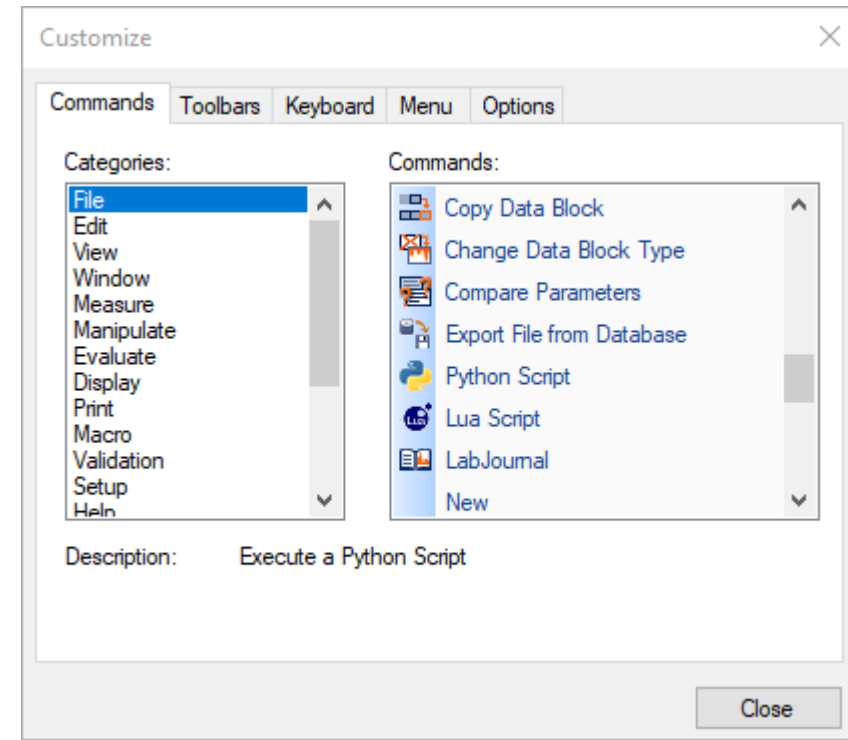
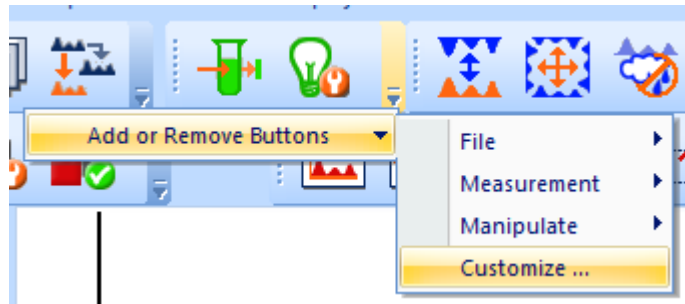
Anwendungsbeispiele

- Automatisierung von Abläufen
- Konvertieren von Daten
- Web Scraping
- Spieleentwicklung

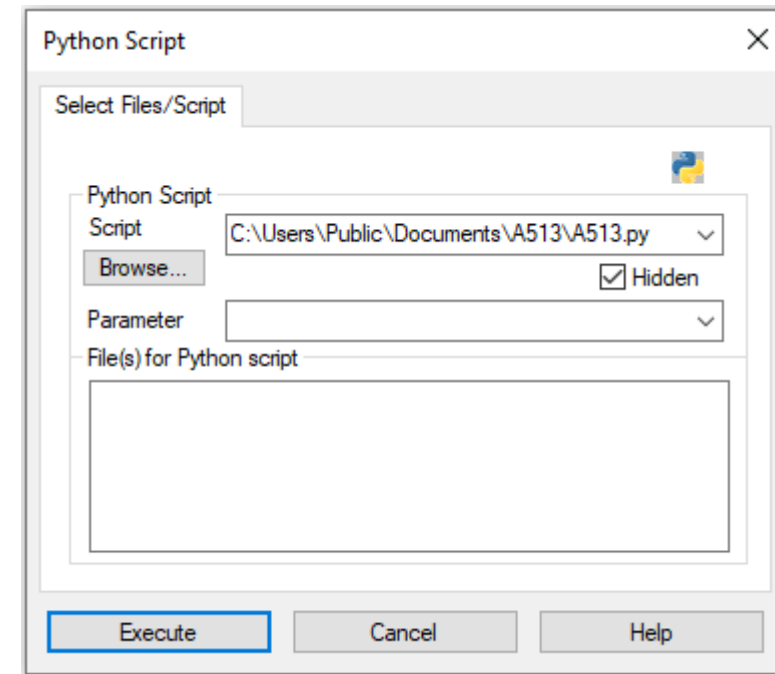
OPUS & PYTHON: NEUE MÖGLICHKEITEN DER AUTOMATISIERUNG

OPUS Python-Funktion

OPUS Python-Funktion



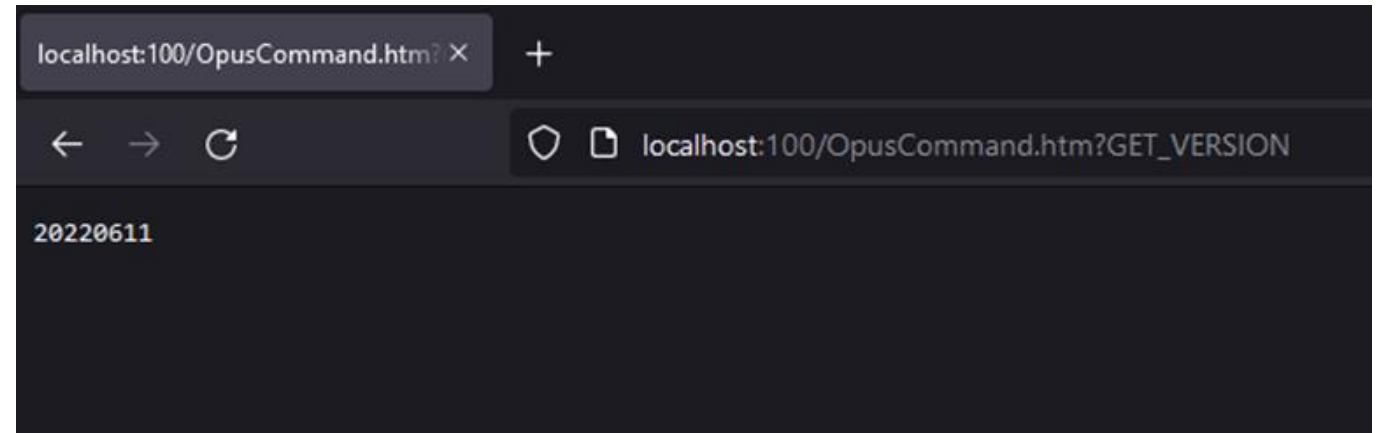
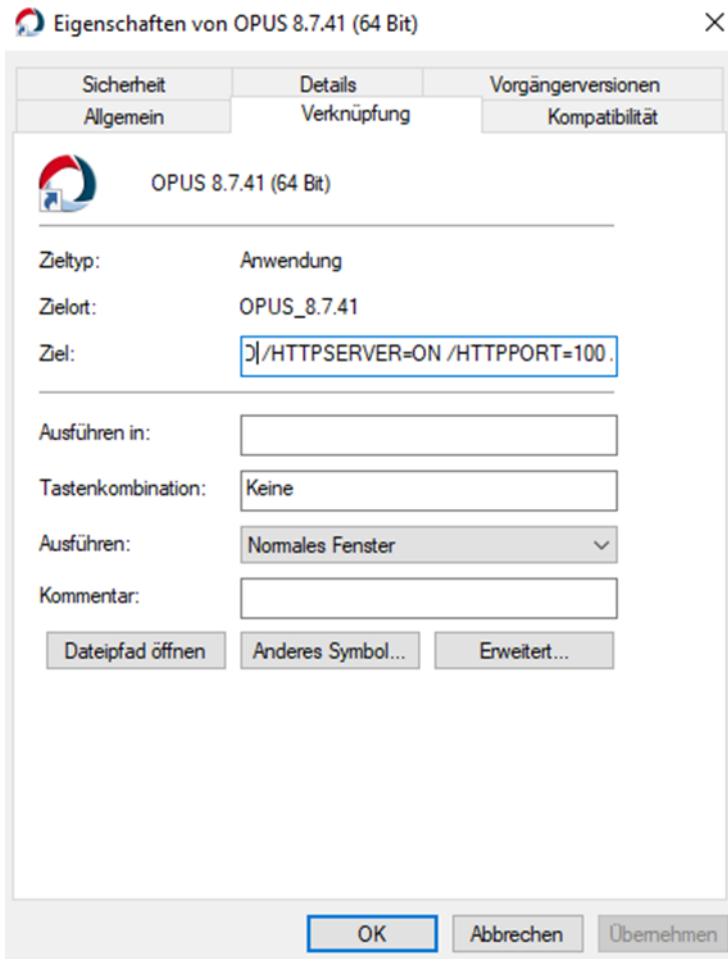
OPUS Python-Funktion



OPUS & PYTHON: NEUE MÖGLICHKEITEN DER AUTOMATISIERUNG

Externe Kommunikation via http

Externe Kommunikation via http



Externe Kommunikation via http

```
import socket

def opusrequest(IP,port,command):
    command=command.replace(" ", "%20")
    request="GET /OpusCommand.htm?" + command + "\r\n\r\n"
    data = ""
    part = None

    s = socket.socket()
    s.connect((IP, port))
    s.sendall(request.encode("windows-1252"))

    while(True):
        part = s.recv(1048576)
        if (part == b''):
            break
        else:
            data += part.decode("windows-1252")

    s.close()
    return(data.split("\n\r\n"))
```

```
version = opusrequest("127.0.0.1", 100, "GET_VERSION")
opusrequest("127.0.0.1", 100, "CLOSE_OPUS")
```

OPUS & PYTHON: NEUE MÖGLICHKEITEN DER AUTOMATISIERUNG

OPUS API

OPUS API

- Zukünftig in OPUS enthalten

```
1 from OpusPythonAPI import OpusAPI
2
3 if __name__ == "__main__":
4
5     opus_api = OpusAPI(opus_http_ip="127.0.0.1", opus_http_port=100)
```

```
1 file_loader = opus_api.OpusFileLoader()
2 file_id, file_path = file_loader.load_file(path_to_opus_file)
3 file_loader.unload_file(file_id)
```

```
1 block_data_reader = opus_api.BlockDataReader(file_id)
2 for block in block_data_reader.blocks:
3     print(block.block_name)
4     block_data = block_data_reader.read_data_from_block(block)
5 # or
6 block_data_list = block_data_reader.read_all_data()
```

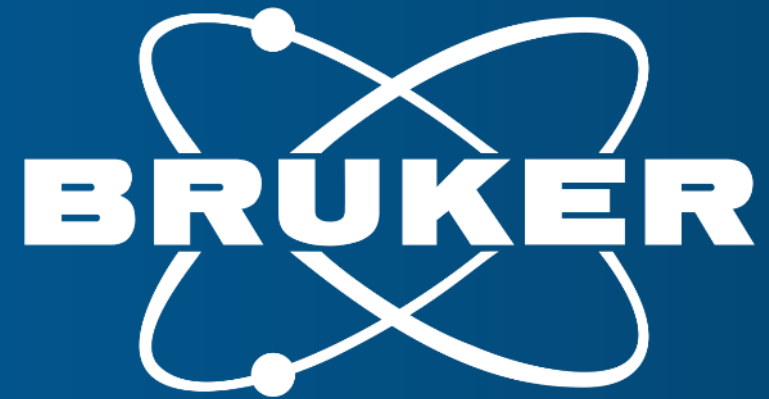
- Dokumentation wird mitgeliefert

OPUS & PYTHON: NEUE MÖGLICHKEITEN DER AUTOMATISIERUNG

OPUS Plugin

OPUS Plugin

- Kommendes Feature
- Python-Skripte können in speziellen OPUS Ordner „PythonScripts“ abgelegt werden
- Erscheinen innerhalb von OPUS wie OPUS-Funktionen



Innovation with Integrity



Dienstag, 20. Juni 2023

18:00 Gesprächstforum & Geräteausstellung

Untergeschoss Bürgersaal

Bitte alle Taschen etc mitnehmen, der Saal wird umgebaut

Ca.19:20 Bankett & Dinnertalk
Stadthalle