



NANOMAGNETICS
INSTRUMENTS

NMIXStudio API MANUAL

Version: 1.0.0

Table of Contents

INSTALLATION	8
PREPARATION	9
IP Address Detection.....	9
Creating the Python File.....	10
Enabling the API in HpSPM+ Software	11
ENDPOINTS	12
OptionsController.....	12
Get Commands.....	12
Get SPM Type List.....	13
Get AFM Mode List	14
Get SPM Type	15
Get AFM Mode.....	16
Get Scale List	17
Get XY Scale.....	18
Get Z Scale.....	19
Set SPM Type.....	20
Set AFM Mode	21
Set XY Scale	22
Set Z Scale	23
StatusController.....	24
Get Commands.....	24
Get Dashboard Status	25
Get Status.....	27
ScanController	28
Get Commands.....	28
Get Is Scanning.....	29
Get Scan Error	30
Get Scan Line Index.....	31
Get Scan Index.....	32

Get X Offset	33
Get Y Offset	34
Get Scan Width Pixel	35
Get Scan Height Pixel	36
Get Image Width	37
Get Image Height	38
Get Is Image Square	39
Get Scan Angle	40
Get Scan Speed	41
Get Scan Number of Averages	42
Get Number of Scans	43
Get Offset Position	44
Get Scan Direction.....	45
Get Is Roundtrip Scan.....	46
Get Is Save Scanned Images	47
Get Capture Pixel.....	48
Get Capture Pixel Unit Text	49
Set Use Default Scan Options.....	50
Set X Offset.....	51
Set Y Offset.....	52
Set Scan Width Pixel.....	53
Set Scan Height Pixel	54
Set Image Width.....	55
Set Image Height	56
Set Is Image Square	57
Set Scan Angle	58
Set Scan Speed	59
Set Scan Number of Averages	60
Set Number Of Scans	61
Set Offset Position	62

Set Scan Direction	63
Set Is Roundtrip Scan	64
Set Is Saved Scanned Images.....	65
Start Scan	66
Stop Scan.....	67
System Readings Controller	68
Get Commands.....	68
Get System Readings.....	69
Get System Readings Unit Text.....	71
ScannedImagesController	73
Get Commands.....	73
Get Nmi Containers.....	74
Get Selected Container Image List	75
Get Selected Container Image	76
Get Selected Container Forward Image List.....	78
Get Selected Container Backward Image List	79
XYOffsetController:.....	80
Get Commands.....	80
Get Is Offset Update Available	81
Get X Offset	82
Get Y Offset	83
Get XY Offset	84
Get X Offset Limit	85
Get Y Offset Limit	86
Get XY Offset Limit	87
Set X Offset Limit.....	89
Set Y Offset Limit.....	90
Set XY Offset.....	91
PIDController	92
Get Commands.....	92

Get Pid Types.....	93
Get PID	94
Set Pid P Value.....	95
Set Pid I Value.....	96
Set Pid D Value	97
Set Pid Set Value.....	98
Set Pid Enable.....	99
Set Pid Negative Polarity Enable	100
Set Pid Invert Vz Enable.....	101
Set Pid Disable.....	102
Set Pid Negative Polarity Disable	103
Set Pid Invert Vz Disable.....	104
AutoTune Controller	105
Get Commands.....	105
Get Is Init.....	106
Get Excitation	107
Get Excitation Percent.....	108
Get Frequency Start In Hertz.....	109
Get Frequency End In Hertz	110
Get Frequency Increment In Hertz.....	111
Get Frequency Slope Types.....	112
Get Frequency Slope Type	113
Get Is Center Span.....	114
Get Start Delay	115
Get Delay.....	116
Get Is Tunning	117
Get Max Rms.....	118
Get Max Rms Frequency	119
Get Min Slope Frequency.....	120
Get Min Slope Rms.....	121

Get Max Slope Rms	122
Get Max Slope Frequency	123
Get Coarse Rms Series	124
Get Coarse Phase Series.....	125
Get Fine Rms Series.....	126
Get Fine Phase Series.....	127
Set Initialize	128
Set Excitation Percent	129
Set Excitation.....	130
Set Frequency Start In Hertz	131
Set Frequency End In Hertz.....	132
Set Frequency Increment In Hertz	133
Set Delay	134
Set Start Delay.....	135
Set Frequency Slope.....	136
Set Center Span Type	137
Set Start End Type	138
Start Tune	139
Stop Tune	140
PhotoDiodeController	141
Get Commands.....	141
Get Is Laser Enabled	142
Get Laser Power	143
Get Laser RF Frequency.....	144
Get FN10	145
Get FN	146
Get FT	147
Set Laser Power	148
Set Laser RF Frequency	149
Null FL.....	150

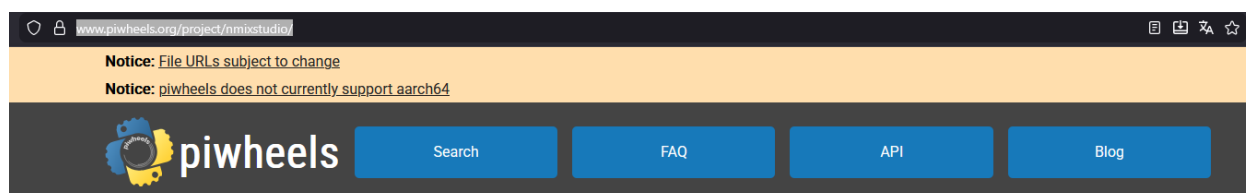
Null 10FN.....	151
Photo Diode Reset.....	152
Set Laser Enable	153
Set Laser Disable	154
Bias Controller	155
Get Commands.....	155
Get DC Offset.....	156
Get AC Amplitude.....	157
Get AC Frequency.....	158
Get Min DC Offset	159
Get Max DC Offset.....	160
Set DC Offset	161
Set AC Amplitude	162
Set AC Frequency	163
WindowController	164
Get Commands.....	164
Get Opened Windows	165
Get Is Opened	166
Set Minimize Window	167
Set Maximize Window	168
Set Close Window	169
Set Minimize All.....	170
Set Normalize All	171

INSTALLATION

Now we will look at the installation process. Go to the web page

<https://www.piwheels.org/project/nmixstudio/> Copy the command provided on the page for the Python library and install the NMIXStudio library by running the command in CMD or any other command-line interface.

Additionally, the command may need to be updated based on the selected version.



nmixstudio

Installation

In a virtualenv (see [these instructions](#) if you need to create one):

```
pip3 install nmixstudio
```

Dependencies

▪ None

Releases

Version	Released	Bullseye Python 3.9	Bookworm Python 3.11	Trixie Python 3.13	Files
1.0.0	2025-12-10	

PyPI page	pypi.org/project/nmixstudio
Project JSON	piwheels.org/project/nmixstudio/json
Versions	1
Files	0
Downloads (all time)	failed to load
Downloads (last 30 days)	failed to load

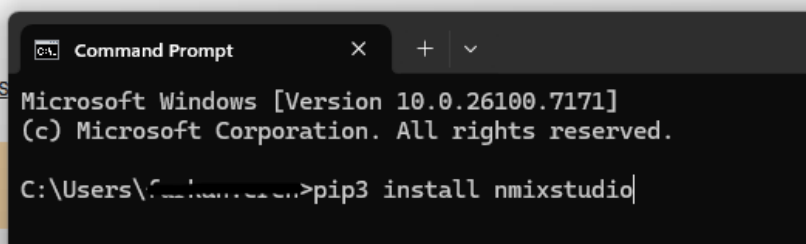
Key

- ✓ Build succeeded
- ✗ Build failed
- ⏸ Build skipped
- ⏳ Build pending

Installation

In a virtualenv (see [these instructions](#))

```
pip3 install nmixstudio
```



PREPARATION

IP Address Detection

1. Open the **Command Prompt** (Windows).
 - Press Win + R, type **cmd**, and press Enter.
2. Run the following command to retrieve your computer's IP address:
 - Ipconfig

3. Look for the **IPv4 Address** under your active network connection. It will look like this:
 - IPv4 Address. : 192.x.x.x

Komut İstemi

```

Microsoft Windows [Version 10.0.19045.5131]
(c) Microsoft Corporation. Tüm hakları saklıdır.

C:\Users\[redacted]>ipconfig

Windows IP Configuration

Ethernet adapter Ethernet:

    Connection-specific DNS Suffix  . : 
    Link-local IPv6 Address . . . . . : 
    IPv4 Address. . . . . : 
    Subnet Mask . . . . . : 
    Default Gateway . . . . . :
  
```

Creating the Python File

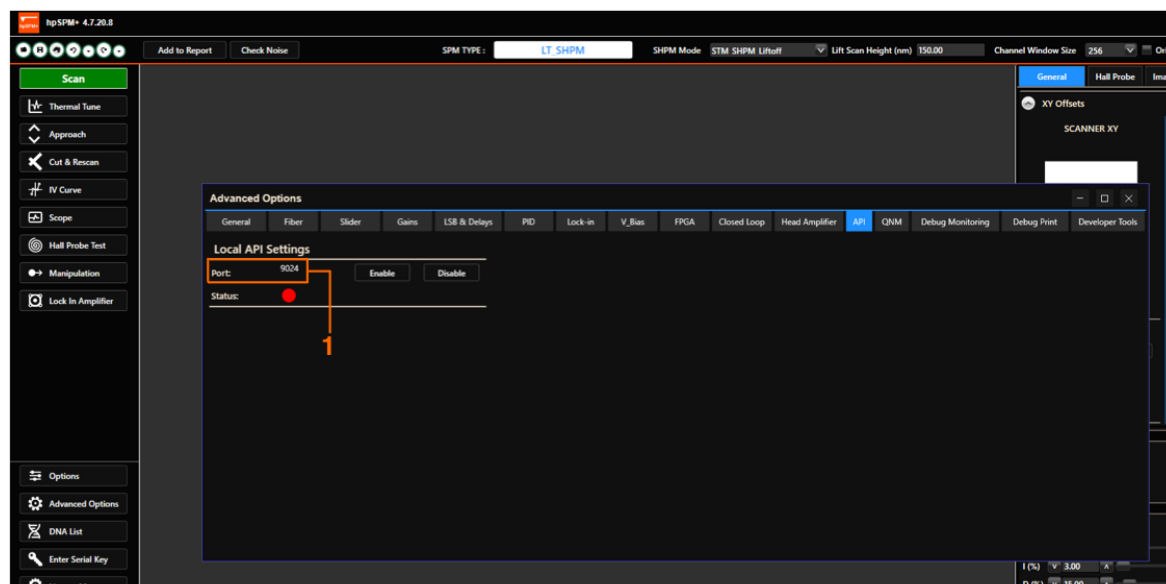
- After configuring the software, create a Python file with the **.py** extension (e.g., connect_device.py).
- Add the following code to the file:

```

from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":
    Endpoint = NMIEndpoint(IP, host)
    Device = NMIDevice(Endpoint)
  
```

- Replace the following placeholders:
 - **IP**: This is the IP address you obtained using the ipconfig command (e.g., 192.168.x.x).
 - **port**: Open the hpSPM+ software, navigate to the "Advanced Options" menu, then go to the "API" tab to check the port number.



- Save the file and ensure your Python environment is properly set up.

Enabling the API in HpSPM+ Software

- Open the **HpSPM+** software again.
- Go to the **Advanced Options** menu.
- In the **Advanced Options** section, find the **API** option and click on it.
- Enable the API by clicking on the **Enable** button.
- Once enabled, you can now use the API to interact with the device using the Python code you created.

ENDPOINTS

OptionsController

Get Commands

Description This method retrieves the list of available commands.

URL `http://{IP}:{port}/Options/Get_Commands`

Parameters None

Response

“status”	True or False
“Commands”	List of available commands.
“msg”	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{port}/Options/Get_Commands'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Options = Device.OPTIONS()

    print(Options.Get_Commands())
```

Get SPM Type List

Description	This method retrieves the list of available SPM types.	
URL	http://{IP}:{port}/Options/Get_SPMTypeList	
Parameters	None	
Response	"status"	True or False
	"SPMTypes"	The current SPM type name.
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{port}/Options/Get_SPMTypeList'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Options = Device.OPTIONS()

    print(Options.Get_SPMTypeList())
```

Description This method retrieves the list of available AFM modes.

URL `http://{IP}:{port}/Options/Get_AFMModeList`

Parameters None

Response

"status"	True or False
"AFMModes"	A string of AFM modes.
"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{port}/Options/Get_AFMModeList'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Options = Device.OPTIONS()

    print(Options.Get_AFMModeList())
```

Get SPM Type

Description This method retrieves the current SPM type.

URL `http://{IP}:{port}/Options/Get_SPMType`

Parameters None

Response

"status"	True or False
"SPMType"	The current SPM type name.
"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{port}/Options/Get_SPMType'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Options = Device.OPTIONS()

    print(Options.Get_SPMType())
```

Description This method retrieves the current AFM mode.

URL `http://{IP}:{port}/Options/Get_AFMMode`

Parameters None

Response

"status"	True or False
"AFMMode"	The current AFM mode.
"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{port}/Options/Get_AFMMode'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Options = Device.OPTIONS()

    print(Options.Get_AFMMode())
```


Description This method retrieves the list of available scale units.

URL http://{IP}:{port}/Options/Get_ScaleList

Parameters None

Response	"status"	True or False
	"ScaleList"	A string of scale units.
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri='http://{IP}:{port}/Options/Get_ScaleList'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudioimport NMIEndpoint,NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Options = Device.OPTIONS()

    print(Options.Get_ScaleList())
```

Description	This method retrieves the current XY scale unit.	
URL	http://{IP}:{port}/Options/Get_XYScale	
Parameters	None	
Response	"status"	True or False
	"XYScale"	The current XY scale unit (e.g., "μm")
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri='http://{IP}:{port}/Options/Get_XYScale'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudioioimport NMIEndpoint,NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Options = Device.OPTIONS()

    print(Options.Get_XYScale())
```

Description	This method retrieves the current Z scale unit.	
URL	http://{IP}:{port}/Options/Get_ZScale	
Parameters	None	
Response	"status"	True or False
	"ZScale"	The current Z scale unit (e.g., "nm")
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri=' http://{IP}:{port}/Options/Get_ZScale'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Options = Device.OPTIONS()

    print(Options.Get_ZScale())
```

Description	This method sets the current SPM type.	
URL	http://{IP}:{port}/Options/Set_SPMType	
Parameters	spmType (str)	The desired SPM type to be set, such as "LT_FABRY_PEROT".
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri='http://{IP}:{port}/Options/Set_SPMType'

payload = {"reg0": "LT_FABRY_PEROT", "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Options = Device.OPTIONS()

    Options.Set_SPMType("LT_FABRY_PEROT")

    print(Options.Get_SPMType()) # control
```

Description	This method sets the current AFM mode.	
URL	http://{IP}:{port}/Options/Set_AFMMode	
Parameters	afmModel(string): The desired AFM mode to be set, such as " LFM".	
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri='http://{IP}:{port}/Options/Set_AFMMode'

payload = {"reg0": "MFM_TwoPass", "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudioimport NMIEndpoint,NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Options = Device.OPTIONS()

    Options.Set_AFMMode("LFM")

    print(Options.Get_AFMMode()) #control
```

Description	This method sets the scale unit for the XY axes.	
URL	http://{IP}:{port}/Options/Set_XYScale	
Parameters	scale (str)	The desired scale unit to be set (e.g., "μm", "nm").
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri='http://{IP}:{port}/Options/Set_XYScale'

payload = {"reg0": "nm", "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Options = Device.OPTIONS()

    Options.Set_XYScale("nm")

    print(Options.Get_XYScale()) #control
```

Description	This method sets the scale unit for the Z axis.	
URL	http://{IP}:{port}/Options/Set_ZScale	
Parameters	scale (str)	The desired scale unit to be set (e.g., "μm", "nm").
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri='http://{IP}:{port}/Options/Set_ZScale'

payload = {"reg0": "nm", "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudioimport NMIEndpoint,NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Options = Device.OPTIONS()

    Options.Set_ZScale("nm")

    print(Options.Get_ZScale()) #control
```

StatusController

Get Commands

Description This method retrieves the list of available commands from the device.

URL http://{IP}:{port}/APP/Get_Commands

Parameters None

Response	"status"	True or False
	"Commands"	List of available commands.
	"msg"	"OK!"

Request Example

```
import requests
headers = {'accept': 'application/json',}
uri='http://{IP}:{port}/APP/Get_Commands'
response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice
if __name__ == "__main__":
    Endpoint = NMIEndpoint(IP,port)
    Device = NMIDevice(Endpoint)
    Status = Device.STATUS()
    print(Status.Get_Commands())
```


Get Dashboard Status

Description	This method retrieves the current dashboard status of the device, including connection state, device information, and scanning status.	
URL	http://{IP}:{port}/APP/Get_DashboardStatus	
Parameters	None	
Response	<div> <div>"status"</div> <div>"DeviceType"</div> <div>"SPMType"</div> <div>"AFMMode"</div> <div>"StatusMessage"</div> <div>"StatusLed"</div> <div>"XOffset"</div> <div>"XOffsetUnit"</div> <div>"YOffset"</div> <div>"YOffsetUnit"</div> <div>"ZOffset"</div> <div>"ZOffsetUnit"</div> <div>"ScannedLinesPercent"</div> <div>"ScannedScanCountPercent"</div> <div>"ScanStatusLed"</div> <div>"LandStatus"</div> <div>"msg"</div> </div>	<div> <div>True or False</div> <div>Type of the device (e.g., "hpSPM+").</div> <div>Scanning probe microscope type (e.g., "LT_SHPM").</div> <div>Current AFM mode (e.g., "STM_SHPM_Liftoff").</div> <div>Message indicating the connection status (e.g., "Not connected").</div> <div>Status LED state ("True" or "False").</div> <div>X-axis offset value (e.g., "0").</div> <div>Unit for the X-offset (e.g., "μm").</div> <div>Y-axis offset value (e.g., "0").</div> <div>Unit for the Y-offset (e.g., "μm").</div> <div>Z-axis offset value (e.g., "0").</div> <div>Unit for the Z-offset (e.g., "nm").</div> <div>Percentage of scanned lines completed (e.g., "0").</div> <div>Percentage of scans completed (e.g., "0").</div> <div>True or False</div> <div>True or False</div> <div>"OK!"</div> </div>

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri='http://{IP}:{port}/APP/Get_DashboardStatus'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Status = Device.STATUS()

    print(Status.Get_DashboardStatus())
```

Description This method retrieves the current system connection status.

URL http://{IP}:{port}/APP/Get_Status

Parameters None

Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri='http://{IP}:{port}/APP/Get_Status'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Status = Device.STATUS()

    print(Status.Get_Status())
```

ScanController

Get Commands

Description This method retrieves the list of available commands.

URL http://{IP}:{Port}/Scan/Get_Commands

Parameters None

Response

"status"	True or False
"Commands"	List of available commands.
"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{port}/Scan/Get_Commands'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP, port)

    Device = NMIDevice(Endpoint)

    Scan = Device.SCAN()

    print(Scan.Get_Commands())
```

Description This method checks if the device is currently scanning.

URL http://{IP}:{Port}/Scan/Get_IsScanning

Parameters None

Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{port}/Scan/Get_IsScanning'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP, port)

    Device = NMIDevice(Endpoint)

    Scan = Device.SCAN()

    print(Scan.Get_IsScanning())
```

Description	This method retrieves the current scan error status.	
URL	http://{IP}:{Port}/Scan/Get_ScanError	
Parameters	None	
Response	"status"	True or False
	"ScanError"	The current scan error status (e.g., "ScanStop_NoFail").
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/Scan/Get_ScanError'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Scan = Device.SCAN()

    print(Scan.Get_ScanError())
```

Get Scan Line Index

Description	This method retrieves the current scan line index.	
URL	http://{IP}:{Port}/Scan/Get_ScanLineIndex	
Parameters	None	
Response	"status"	True or False
	"ScanLineIndex"	The current scan line index as a string (e.g., "0").
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/Scan/Get_ScanLineIndex'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Scan = Device.SCAN()

    print(Scan.Get_ScanLineIndex())
```

Description	This method retrieves the current scan index.	
URL	http://{IP}:{Port}/Scan/Get_ScanIndex	
Parameters	None	
Response	"status"	True or False
	"ScanIndex"	The current scan index as a string (e.g., "0").
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/Scan/Get_ScanIndex'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Scan = Device.SCAN()

    print(Scan.Get_ScanIndex())
```


Description	This method retrieves the current X-axis offset value.	
URL	http://{IP}:{Port}/Scan/Get_XOffset	
Parameters	None	
Response	"status"	True or False
	"XOffset"	The current X-axis offset value (e.g., 0).
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/Scan/Get_XOffset'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Scan = Device.SCAN()

    print(Scan.Get_XOffset())
```

Description	This method retrieves the current Y-axis offset value.	
URL	http://{IP}:{Port}/Scan/Get_YOffset	
Parameters	None	
Response	"status"	True or False
	"YOffset"	The current Y-axis offset value (e.g., 0).
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/Scan/Get_YOffset'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Scan = Device.SCAN()

    print(Scan.Get_YOffset())
```

Description	This method retrieves the width of the scan in pixels.	
URL	http://{IP}:{Port}/Scan/Get_ScanWidthPixel	
Parameters	None	
Response	"status"	True or False
	"ScanWidthPixel"	The width of the scan in pixels (e.g., 256).
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/Scan/Get_ScanWidthPixel'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Scan = Device.SCAN()

    print(Scan.Get_ScanWidthPixel())
```

Get Scan Height Pixel

Description	This method retrieves the height of the scan in pixels.	
URL	http://{IP}:{Port}/Scan/Get_ScanHeightPixel	
Parameters	None	
Response	"status"	True or False
	"ScanHeightPixel"	The height of the scan in pixels (e.g., 256).
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/Scan/Get_ScanHeightPixel'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Scan = Device.SCAN()

    print(Scan.Get_ScanHeightPixel())
```

Get Image Width

Description	This method retrieves the width of the image in specified units (e.g., nanometers).	
URL	http://{IP}:{Port}/Scan/Get_ImageWidth	
Parameters	None	
Response	"status"	True or False
	"ImageWidth"	The width of the image as a float (e.g., 4799.9999999999991)
	"ImageWidthUnit"	The unit of measurement for the image width (e.g., "nm").
	"msg"	"OK!"

Request Example

```
import requests
headers = {'accept': 'application/json',}
uri = 'http://{IP}:{Port}/Scan/Get_ImageWidth'
response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice
if __name__ == "__main__":
    Endpoint = NMIEndpoint(IP, port)
    Device = NMIDevice(Endpoint)
    Scan = Device.SCAN()
    print(Scan.Get_ImageWidth())
```

Get Image Height

Description	This method retrieves the width of the image in specified units (e.g., nanometers).	
URL	http://{IP}:{Port}/Scan/Get_ImageHeight	
Parameters	None	
Response	"status"	True or False
	"ImageHeight"	The height of the image as a float (e.g., 4799.9999999999991)
	"ImageHeightUnit"	The unit of measurement for the image height (e.g., "nm").
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/Scan/Get_ImageHeight '

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Scan = Device.SCAN()

    print(Scan.Get_ImageHeight())
```

Get Is Image Square

Description This method checks whether the image is square.

URL `http://{IP}:{Port}/Scan/Get_IsImageSquare`

Parameters None

Response	"status"	True or False
	"IsImageSquare"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/Scan/Get_IsImageSquare'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP, port)

    Device = NMIDevice(Endpoint)

    Scan = Device.SCAN()

    print(Scan.Get_IsImageSquare())
```

Description	This method retrieves the current scan angle value.	
URL	http://{IP}:{Port}/Scan/Get_ScanAngle	
Parameters	None	
Response	"status"	True or False
	"ScanAngle"	The current scan angle as an integer (e.g., 270).
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/Scan/Get_ScanAngle'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Scan = Device.SCAN()

    print(Scan.Get_ScanAngle())
```


Get Scan Speed

Description	This method retrieves the current scan speed.	
URL	http://{IP}:{Port}/Scan/Get_ScanSpeed	
Parameters	None	
Response	"status"	True or False
	"ScanSpeed"	The current scan speed as a float (e.g., 9999999.9999999963).
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/Scan/Get_ScanSpeed'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Scan = Device.SCAN()

    print(Scan.Get_ScanSpeed())
```

Get Scan Number of Averages

Description	This method retrieves the number of averages for the current scan.	
URL	http://{IP}:{Port}/Scan/Get_ScanNumberOfAverages	
Parameters	None	
Response	"status"	True or False
	"ScanNumberOfAverages"	The number of averages as an integer (e.g., 4).
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/Scan/Get_ScanNumberOfAverages'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Scan = Device.SCAN()

    print(Scan.Get_ScanNumberOfAverages())
```

Get Number of Scans

Description	This method retrieves the total number of scans configured.	
URL	http://{IP}:{Port}/Scan/Get_NumberOfScans	
Parameters	None	
Response	"status"	True or False
	"NumberOfScans"	The total number of scans as an integer (e.g., 10).
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/Scan/Get_NumberOfScans'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Scan = Device.SCAN()

    print(Scan.Get_NumberOfScans())
```

Description	This method retrieves the current offset position of the scan.	
URL	http://{IP}:{Port}/Scan/Get_OffsetPosition	
Parameters	None	
Response	"status"	True or False
	"OffsetPosition"	A string indicating the offset position (e.g., "BottomLeft").
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/Scan/Get_OffsetPosition'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Scan = Device.SCAN()

    print(Scan.Get_OffsetPosition())
```

Get Scan Direction

Description	This method retrieves the current scan direction.	
URL	http://{IP}:{Port}/Scan/Get_ScanDirection	
Parameters	None	
Response	"status"	True or False
	"ScanDirection"	A string indicating the scan direction (e.g., "BottomToTop").
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/Scan/Get_ScanDirection'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Scan = Device.SCAN()

    print(Scan.Get_ScanDirection())
```

Get Is Roundtrip Scan

Description This method checks if the current scan is configured as a roundtrip scan.

URL `http://{IP}:{Port}/Scan/Get_IsRoundtripScan`

Parameters None

Response	"status"	True or False
	"IsRoundtripScan"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/Scan/Get_IsRoundtripScan'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP, port)

    Device = NMIDevice(Endpoint)

    Scan = Device.SCAN()

    print(Scan.Get_IsRoundtripScan())
```

Get Is Save Scanned Images

Description This method checks whether saving scanned images is enabled.

URL `http://{IP}:{Port}/Scan/Get_IsSaveScannedImages`

Parameters None

Response	"status"	True or False
	"IsSaveScannedImages"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/Scan/Get_IsSaveScannedImages'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Scan = Device.SCAN()

    print(Scan.Get_IsSaveScannedImages())
```

Description This method retrieves the capture pixel value for the scan.

URL `http://{IP}:{Port}/Scan/Get_CapturePixel`

Parameters None

Response "status" True or False
 "msg" "OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/Scan/Get_CapturePixel'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudioimport NMIEndpoint,NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Scan = Device.SCAN()

    print(Scan.Get_CapturePixel())
```


Get Capture Pixel Unit Text

Description This method retrieves the unit text for the capture pixel (e.g., "pixels").

URL URL: http://{IP}:{Port}/Scan/Get_CapturePixelUnitText

Parameters None

Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/Scan/Get_CapturePixelUnitText'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP, port)

    Device = NMIDevice(Endpoint)

    Scan = Device.SCAN()

    print(Scan.Get_CapturePixelUnitText())
```

Set Use Default Scan Options

Description This method sets whether to use the default scan options for the scan.

URL `http://{IP}:{Port}/Scan/Set_UseDefaultScanOptions`

Parameters `useDefaultScanOptions(bool)`

Response	<code>"status"</code> <code>"msg"</code>	True or False "OK!"
-----------------	---	------------------------

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri='http://{IP}:{Port}/Scan/Set_UseDefaultScanOptions'

payload = {"reg0": "True", "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Scan = Device.SCAN()

    Scan.Set_UseDefaultScanOptions(True)
```

Description	This method sets the X-axis offset value for the scan.	
URL	http://{IP}:{Port}/Scan/Set_XOffset	
Parameters	xOffset(float)	The X-axis offset value to set.
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/Scan/Set_XOffset'

payload = {"reg0": "1", "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Scan = Device.SCAN()

    Scan.Set_XOffset(1)

    print(Scan.Get_XOffset()) #control
```

Description	This method sets the Y-axis offset value for the scan.	
URL	http://{IP}:{Port}/Scan/Set_YOffset	
Parameters	yOffset(float)	The Y-axis offset value to set.
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/Scan/Set_XOffset'

payload = {"reg0": "1", "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Scan = Device.SCAN()

    Scan.Set_YOffset(1)

    print(Scan.Get_YOffset()) #control
```

Description	This method retrieves the width of the scan in pixels.	
URL	http://{IP}:{Port}/Scan/Get_ScanWidthPixel	
Parameters	pixel(int)	The scan width in pixels (e.g., 256).
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/Scan/Get_ScanWidthPixel'

payload = {"reg0": "512", "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Scan = Device.SCAN()

    Scan.Set_ScanWidthPixel(1024)

    print(Scan.Get_ScanWidthPixel()) #control
```

Set Scan Height Pixel

Description	This method retrieves the width of the scan in pixels.	
URL	http://{IP}:{Port}/Scan/Set_ScanHeightPixel	
Parameters	pixel(int)	The scan height in pixels (e.g., 256).
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/Scan/Set_ScanHeightPixel'

payload = {"reg0": "512", "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Scan = Device.SCAN()

    Scan.Set_ScanHeightPixel(1024)

    print(Scan.Get_ScanHeightPixel()) #control
```

Set Image Width

Description	This method sets the width of the image in the specified unit (e.g., nanometers).	
URL	http://{IP}:{Port}/Scan/Set_ImageWidth	
Parameters	width(float)	The image width value (e.g., 256.0).
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/Scan/Set_ImageWidth'

payload = {"reg0": "256", "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Scan = Device.SCAN()

    Scan.Set_ImageWidth(256)

    print(Scan.Get_ImageWidth()) #control
```

Set Image Height

Description	This method sets the height of the image in the specified unit (e.g., nanometers).	
URL	http://{IP}:{Port}/Scan/Set_ImageHeight	
Parameters	height(float)	The image height value (e.g., 256.0).
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/Scan/Set_ImageHeight'

payload = {"reg0": "256", "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Scan = Device.SCAN()

    Scan.Set_ImageHeight(256)

    print(Scan.Get_ImageHeight()) #control
```


Set Is Image Square

Description This method sets whether the image should be square.

URL `http://{IP}:{Port}/Scan/Set_IsImageSquare`

Parameters `isImageSquare(bool)`

Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/Scan/Set_IsImageSquare'

payload = {"reg0": True, "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Scan = Device.SCAN()

    Scan.Set_IsImageSquare(True)

    print(Scan.Get_IsImageSquare()) #control
```

Description	This method sets the scan angle for the current scan operation.	
URL	http://{IP}:{Port}/Scan/Set_ScanAngle	
Parameters	scanAngle(float)	The scan angle value to set (e.g., 210.0).
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/Scan/Set_ScanAngle'

payload = {"reg0": 210, "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Scan = Device.SCAN()

    Scan.Set_ScanAngle(210)

    print(Scan.Get_ScanAngle()) #control
```

Set Scan Speed

Description	This method sets the scan speed for the current scan operation.	
URL	http://{IP}:{Port}/Scan/Set_ScanSpeed	
Parameters	scanSpeed(float)	The scan speed value to set (e.g., 250.0).
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/Scan/Set_ScanSpeed'

payload = {"reg0": 250.0, "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Scan = Device.SCAN()

    Scan.Set_ScanSpeed(250.0)

    print(Scan.Get_ScanSpeed()) #control
```

Set Scan Number of Averages

Description	This method sets the number of averages for the current scan.	
URL	http://{IP}:{Port}/Scan/Set_ScanNumberOfAverages	
Parameters	numberOfAverage(int)	The number of averages to set (e.g., 100).
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/Scan/Set_ScanNumberOfAverages'

payload = {"reg0": 100, "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Scan = Device.SCAN()

    Scan.Set_ScanNumberOfAverages(100)

    print(Scan.Get_ScanNumberOfAverages()) #control
```

Set Number Of Scans

Description	This method sets the offset position for the current scan.	
URL	http://{IP}:{Port}/Scan/Set_NumberOfScans	
Parameters	numberOfScan(int)	The number of scans to be executed.
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/Scan/Set_NumberOfScans'

payload = {"reg0": 1, "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Scan = Device.SCAN()

    Scan.Set_NumberOfScans(1)

    print(Scan.Get_NumberOfScans()) #control
```

Description	This method sets the offset position for the current scan.	
URL	http://{IP}:{Port}/Scan/Set_OffsetPosition	
Parameters	offsetPosition(str)	The offset position value (e.g., "BottomLeft" or "Center").
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/Scan/Set_OffsetPosition'

payload = {"reg0": "Center", "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Scan = Device.SCAN()

    Scan.Set_OffsetPosition("Center")

    print(Scan.Get_OffsetPosition()) #control
```

Description	This method sets the scan direction for the current scan operation.	
URL	http://{IP}:{Port}/Scan/Set_ScanDirection	
Parameters	scanDirection(str)	The scan direction value (e.g., "BottomToTop" or "TopToBottom").
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/Scan/Set_ScanDirection'

payload = {"reg0": "BottomToTop", "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Scan = Device.SCAN()

    Scan.Set_ScanDirection("BottomToTop")

    print(Scan.Get_ScanDirection()) #control
```

Set Is Roundtrip Scan

Description This method sets whether the scan should operate in a roundtrip mode.

URL `http://{IP}:{Port}/Scan/Set_IsRoundtripScan`

Parameters `isRoundtripScan(bool)`

Response

"status"	True or False
"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/Scan/Set_IsRoundtripScan'

payload = {"reg0": True, "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Scan = Device.SCAN()

    Scan.Set_IsRoundtripScan(True)

    print(Scan.Get_IsRoundtripScan()) #control
```


Set Is Saved Scanned Images

Description This method sets whether scanned images should be saved.

URL `http://{IP}:{Port}/Scan/Set_IsSaveScannedImages`

Parameters `isSaveImages(bool)`

Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/Scan/Set_IsSaveScannedImages'

payload = {"reg0": True, "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Scan = Device.SCAN()

    Scan.Set_IsSaveScannedImages(True)

    print(Scan.Get_IsSaveScannedImages()) #control
```

Description This method starts the scan process.

URL http://{IP}:{Port}/Scan/StartScan

Parameters status(bool)

Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/Scan/StartScan'

payload = {"reg0": True, "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Scan = Device.SCAN()

    Scan.StartScan(True)
```

Description This method stops the current scan operation.

URL http://{IP}:{Port}/Scan/StopScan

Parameters status(bool)

Response "status" True or False
"msg" "OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/Scan/StopScan'

payload = {"reg0": True, "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Scan = Device.SCAN()

    Scan.StopScan(True)
```

System Readings Controller

Get Commands

Description This method retrieves the list of available commands.

URL http://{IP}:{Port}/SystemReadings/Get_Commands

Parameters None

Response

"status"	True or False
"Commands"	List of available commands.
"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/SystemReadings/Get_Commands'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    SysRead= Device.SYSTEMREADINGS()

    print(SysRead.Get_Commands())
```

Get System Readings

Description	This method retrieves the current system readings.	
URL	http://{IP}:{Port}/SystemReadings/Get_SystemReadings	
Parameters	None	
Response	"status" "Fnx10" "Fn" "FI" "Ft" "Spare1, Spare2, Spare4" "It" "VHall" "lHall" "Backplane2, Backplane3" "Vpd" "VpdRef" "VpdSig" "Vbias" "Lia1RMS" "Lia1Phase" "Lia2RMS" "Lia2Phase" "FastAdc1, FastAdc2" "Vz, Vx, Vy" "DeltaF" "FiberPZT" "ExternalNPX, ExternalNPY, ExternalNPZ" "Spare4_LMT87" "EncoderX, EncoderY, EncoderZ" "msg"	True or False Voltage (scaled force measurement). Voltage measurement. Voltage measurement. Voltage measurement. Reserved or additional readings. Tunneling current (nanoAmps). Hall voltage. Hall current. Backplane measurements. Photodiode voltage. Reference photodiode voltage. Signal photodiode voltage. Bias voltage. RMS value of Lock-In Amplifier 1. Phase of Lock-In Amplifier 1. RMS value of Lock-In Amplifier 2. Phase of Lock-In Amplifier 2. Fast ADC values. Voltage applied in the X, Y, Z axes. Frequency change. Fiber PZT value. External position measurements. Sensor temperature reading. Encoder readings. "OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/SystemReadings/Get_SystemReadings'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    SysRead= Device.SYSTEMREADINGS()

    print(SysRead.Get_SystemReadings())
```

Get System Readings Unit Text

Description	This method retrieves the unit text for the system readings.	
URL	http://{IP}:{Port}/SystemReadings/Get_SystemReadingsUnitText	
Parameters	None	
Response	"status"	True or False
	"Fnx10"	V
	"Fn"	V
	"FI"	V
	"Ft"	V
	"Spare1, Spare2, Spare4"	mV
	"It"	nA
	"VHall"	mV
	"IHall"	nA
	"Backplane2, Backplane3"	mV
	"Vpd"	V
	"VpdRef"	V
	"VpdSig"	V
	"Vbias"	V
	"Lia1RMS"	V
	"Lia1Phase"	°
	"Lia2RMS"	V
	"Lia2Phase"	°
	"FastAdc1, FastAdc2"	V
	"Vz, Vx, Vy"	V
	"DeltaF"	Hz
	"FiberPZT"	V
	"ExternalNPX, ExternalNPY, ExternalNPZ"	-
	"Spare4_LMT87"	-
	"EncoderX, EncoderY, EncoderZ"	-
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/SystemReadings/Get_SystemReadingsUnitText'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    SysRead= Device.SYSTEMREADINGS()

    print(SysRead.Get_SystemReadingsUnitText()) #control
```


ScannedImagesController

Get Commands

Description This method retrieves the list of available commands.

URL http://{IP}:{Port}/ScannedImages/Get_Commands

Parameters None

Response

"status"	True or False
"Commands"	List of available commands.
"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/ScannedImages/Get_Commands'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP, port)

    Device = NMIDevice(Endpoint)

    ScannedImages = Device.SCANNEDIMAGES()

    print(ScannedImages.Get_Commands())
```

Get Nmi Containers

Description	This method Retrieves a list of NMI Containers available in the system.	
URL	http://{IP}:{Port}/ScannedImages/Get_NmiContainerList	
Parameters	None	
Response	"status"	True or False
	"NmiContainerList"	A list of available NMI containers (e.g., "container_1;container_2;container_3").
	"msg"	"OK!"

Request Example

```
import requests
headers = {'accept': 'application/json',}
uri = 'http://{IP}:{Port}/ScannedImages/Get_NmiContainerList'
response = requests.get(uri, headers=headers, json=payload)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice
if __name__ == "__main__":
    Endpoint = NMIEndpoint(IP,port)
    Device = NMIDevice(Endpoint)
    ScannedImages = Device.SCANNEDIMAGES()
    print(ScannedImages.Get_NmiContainers())
```

Get Selected Container Image List

Description	This method retrieves a list of images from the specified NMI container.	
URL	http://{IP}:{Port}/ScannedImages/Get_SelectedContainerImageList	
Parameters	containerName(str)	The name of the NMI container.
Response	"status"	True or False
	"SelectedContainerImageList"	A list of images in the container (e.g., "Image_1 (256x256); Image_2 (256x256)").
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/ScannedImages/Get_SelectedContainerImageList'

payload = {"reg0": "example_container_name", "reg1": "", "reg2": "",
"reg3": ""}

response = requests.get(uri, headers=headers, json=payload)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    ScannedImages = Device.SCANNEDIMAGES()

    print(ScannedImages.Get_SelectedContainerImageList("example_container_name"))
```

Get Selected Container Image

Description	This method retrieves a specific image from the specified NMI container by its name.	
URL	<code>http://{IP}:{Port}/ScannedImages/Get_SelectedContainerImage</code>	
Parameters	<code>containerName(str)</code>	The name of the NMI container.
	<code>imageName(str)</code>	The name of the image to fetch from the container.
Response	<code>"status"</code>	True or False
	<code>"RawBuffer"</code>	Raw data buffer for the image.
	<code>"RealBuffer"</code>	Processed (real) data buffer for the image.
	<code>"RealHeightUnit"</code>	Unit of real height (e.g., "nm").
	<code>"RealWidthUnit"</code>	Unit of real width (e.g., "nm").
	<code>"RealHeightUnitPrefix"</code>	Prefix for real height unit (e.g., "nano").
	<code>"RealWidthUnitPrefix"</code>	Prefix for real width unit (e.g., "nano").
	<code>"RealHeight"</code>	Real-world height of the image.
	<code>"RealWidth"</code>	Real-world width of the image.
	<code>"WidthPixel"</code>	Width of the image in pixels.
	<code>"HeightPixel"</code>	Height of the image in pixels.
	<code>"msg"</code>	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/ScannedImages/Get_SelectedContainerImage'

payload = {"reg0": "example_container_name", "reg1": "example_image_name",
"reg2": "", "reg3": ""}

response = requests.get(uri, headers=headers, json=payload)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudioimport NMIEndpoint,NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    ScannedImages = Device.SCANNEDIMAGES()

    print(ScannedImages.Get_SelectedContainerImage("example_container_name",
"example_image_name"))
```

Get Selected Container Forward Image List

Description	This method retrieves a list of forward images stored in a specified NMI Container.	
URL	http://{IP}:{Port}/ScannedImages/Get_SelectedContainerForwardImageList	
Parameters	containerName(str)	The name of the NMI container.
Response	"status" "SelectedContainerForwardImageList" "msg"	True or False A list of forward images within the specified container. "OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri=
'http://{IP}:{Port}/ScannedImages/Get_SelectedContainerForwardImageList'

payload = {"reg0": "example_container_name", "reg1": "", "reg2": "",
"reg3": ""}

response = requests.get(uri, headers=headers, json=payload)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    ScannedImages = Device.SCANNEDIMAGES()

    print(ScannedImages.Get_SelectedContainerForwardImageList("example_container_name"))
```

Get Selected Container Backward Image List

Description	This method retrieves a list of backward images stored in a specified NMI Container.	
URL	http://{IP}:{Port}/ScannedImages/Get_SelectedContainerBackwardImageList	
Parameters	containerName(str)	The name of the NMI container.
Response	"status" "SelectedContainerBackwardImageList" "msg"	True or False A list of backward images within the specified container. "OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri =
'http://{IP}:{Port}/ScannedImages/Get_SelectedContainerBackwardImageList'

payload = {"reg0": "example_container_name", "reg1": "", "reg2": "",
"reg3": ""}

response = requests.get(uri, headers=headers, json=payload)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    ScannedImages = Device.SCANNEDIMAGES()

    print(ScannedImages.Get_SelectedContainerBackwardImageList("example_container_name"))
```

XYOffsetController:

Get Commands

Description This method retrieves the list of available commands.

URL http://{IP}:{Port}/XYOffset/Get_Commands

Parameters None

Response

"status"	True or False
"Commands"	List of available commands.
"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/XYOffset/Get_Commands'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP, port)

    Device = NMIDevice(Endpoint)

    Offset = Device.XYOFFSET()

    print(Offset.Get_Commands())
```


Get Is Offset Update Available

Description This method checks if an offset update is currently available.

URL `http://{IP}:{Port}/XYOffset/Get_IsOffsetUpdateAvailable`

Parameters None

Response	"status"	True or False
	"IsOffsetUpdateAvailable"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/XYOffset/Get_IsOffsetUpdateAvailable'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Offset = Device.XYOFFSET()

    print(Offset.Get_IsOffsetUpdateAvailable())
```

Description	This method retrieves the current X-axis offset value.	
URL	http://{IP}:{Port}/XYOffset/Get_XOffset	
Parameters	None	
Response	"status" "XOffset" "XOffsetPrefix" o Prefix : o Symbol : o Scalar : "msg"	True or False The current X-axis offset value (e.g., 0). Details about the prefix used for the offset value: Unit prefix (e.g., "nano"). Symbol of the prefix (e.g., "n"). Scalar multiplier for the prefix (e.g., 1.0000000000000001E-09). "OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/XYOffset/Get_XOffset'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP, port)

    Device = NMIDevice(Endpoint)

    Offset = Device.XYOFFSET()

    print(Offset.Get_XOffset())
```

Description	This method retrieves the current Y-axis offset value.	
URL	http://{IP}:{Port}/XYOffset/Get_YOffset	
Parameters	None	
Response	"status"	True or False
	"YOffset"	The current Y-axis offset value (e.g., 0).
	"YOffsetPrefix"	Details about the prefix used for the offset value:
	o Prefix :	Unit prefix (e.g., "nano").
	o Symbol :	Symbol of the prefix (e.g., "n").
	o Scalar :	Scalar multiplier for the prefix (e.g., 1.0000000000000001E-09).
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/XYOffset/Get_YOffset'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP, port)

    Device = NMIDevice(Endpoint)

    Offset = Device.XYOFFSET()

    print(Offset.Get_YOffset())
```

Description	This method retrieves the current X-axis and Y-axis offset values.	
URL	http://{IP}:{Port}/XYOffset/Get_XYOffset	
Parameters	None	
Response	"status" "XOffset" "XOffsetPrefix" ○ Prefix : ○ Symbol : ○ Scalar : "XOffsetUnitText" "YOffset" "YOffsetPrefix" ○ Prefix : ○ Symbol : ○ Scalar : "YOffsetUnitText" "msg"	True or False The current X-axis offset value (e.g., 0). Details about the prefix used for the X-axis offset: Unit prefix (e.g., "nano"). Symbol of the prefix (e.g., "n"). Scalar multiplier for the prefix (e.g., 1.0000000000000001E-09). Unit of measurement for the X-axis offset (e.g., "nm"). The current Y-axis offset value (e.g., 0). Details about the prefix used for the Y-axis offset: Unit prefix (e.g., "nano"). Symbol of the prefix (e.g., "n"). Scalar multiplier for the prefix (e.g., 1.0000000000000001E-09). Unit of measurement for the Y-axis offset (e.g., "nm"). "OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = http://{IP}:{Port}/XYOffset/Get_XYOffset'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Offset = Device.XYOFFSET()

    print(Offset.Get_XYOffset())
```

Description	This method retrieves the positive and negative limit values.	
URL	http://{IP}:{Port}/XYOffset/Get_XOffsetLimit	
Parameters	None	
Response	"status"	True or False
	"XOffsetPositiveLimit"	The positive limit for the X-axis offset (e.g., 49984.7949999999962).
	"XOffsetNegativeLimit"	The negative limit for the X-axis offset (e.g., -49984.7949999999962).
	"XOffsetUnitText"	Unit of measurement for the X-axis offset limits (e.g., "nm").
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/XYOffset/Get_XOffsetLimit'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Offset = Device.XYOFFSET()

    print(Offset.Get_XOffsetLimit())
```

Description	This method retrieves the positive and negative limit values.	
URL	http://{IP}:{Port}/XYOffset/Get_YOffsetLimit	
Parameters	None	
Response	"status"	True or False
	"YOffsetPositiveLimit"	The positive limit for the Y-axis offset (e.g., 49984.794999999962).
	"YOffsetNegativeLimit"	The negative limit for the Y-axis offset (e.g., -49984.794999999962).
	"YOffsetUnitText"	Unit of measurement for the Y-axis offset limits (e.g., "nm").
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/XYOffset/Get_YOffsetLimit'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Offset = Device.XYOFFSET()

    print(Offset.Get_YOffsetLimit())
```

Get XY Offset Limit

Description	This method retrieves the positive and negative limit values for both X-axis and Y-axis offsets	
URL	http://{IP}:{Port}/XYOffset/Get_XYOffsetLimit	
Parameters	None	
Response	"status"	True or False
	"XOffsetPositiveLimit"	The positive limit for the X-axis offset (e.g., 49984.794999999962).
	"XOffsetNegativeLimit"	The negative limit for the X-axis offset (e.g., -49984.794999999962).
	"XOffsetUnitText"	Unit of measurement for the X-axis offset limits (e.g., "nm").
	"YOffsetPositiveLimit"	The positive limit for the Y-axis offset (e.g., 49984.794999999962).
	"YOffsetNegativeLimit"	The negative limit for the Y-axis offset (e.g., -49984.794999999962).
	"YOffsetUnitText"	Unit of measurement for the Y-axis offset limits (e.g., "nm").
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/XYOffset/Get_XYOffsetLimit'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudioimport NMIEndpoint,NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Offset = Device.XYOFFSET()

    print(Offset.Get_XYOffsetLimit())
```


Description	This method sets the X-axis offset value.	
URL	http://{IP}:{Port}/XYOffset/Set_XOffset	
Parameters	"xOffset"	The X-axis offset value to set. (e.g., 120.5).
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/XYOffset/Set_XOffset'

payload = {"reg0": 120.5, "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    offset = Device.XYOFFSET()

    offset.Set_XOffset(120.5)

    print(offset.Get_XOffset()) #control
```

Description	This method sets the Y-axis offset value.	
URL	http://{IP}:{Port}/XYOffset/Set_YOffset	
Parameters	"yOffset"	The Y-axis offset value to set. (e.g., 150.5).
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/XYOffset/Set_YOffset'

payload = {"reg0": 150.5, "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP, port)

    Device = NMIDevice(Endpoint)

    offset = Device.XYOFFSET()

    offset.Set_YOffset(150.5)

    print(offset.Get_YOffset()) #control
```

Description	This method sets the for both X-axis and Y-axis offset value.	
URL	http://{IP}:{Port}/XYOffset/Set_XYOffset	
Parameters	"xOffset"	The X-axis offset value to set. (e.g., 120.5).
	"yOffset"	The Y-axis offset value to set. (e.g., 150.5).
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/XYOffset/Set_XYOffset'

payload = {"reg0": 120.5, "reg1": 150.5, "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudioimport NMIEndpoint,NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Offset = Device.XYOFFSET()

    Offset.Set_XYOffset(120.5, 150.5)

    print(Offset.Get_XYOffset()) #control
```

PIDController

Get Commands

Description This method retrieves the list of available commands.

URL http://{IP}:{Port}/PID/Get_Commands

Parameters None

Response

"status"	True or False
"Commands"	List of available commands.
"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/PID/Get_Commands'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP, port)

    Device = NMIDevice(Endpoint)

    Pid = Device.PID()

    print(Pid.Get_Commands())
```

Description	This method retrieves the available PID types	
URL	http://{IP}:{Port}/PID/Get_PidTypes	
Parameters	None	
Response	"status"	True or False
	"Types"	A semicolon-separated string listing the available PID types.
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/PID/Get_Commands'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Pid = Device.PID()

    print(Pid.Get_PidTypes())
```

Description	This method retrieves PID parameters based on the provided PID index.	
URL	http://{IP}:{Port}/PID/Get_Pid	
Parameters	pidIndex(int)	The index of the PID to retrieve (e.g., 0, 1, 2, 3).
Response	"status"	True or False
	"PID"	A dictionary containing the PID parameters for the requested index.
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/PID/Get_Pid'

payload = payload = {"reg0": 0, "reg1": "", "reg2": "", "reg3": ""}

response = requests.get(uri, headers=headers, json=payload)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Pid = Device.PID()

    print(Pid.Get_PID(0))
```

Description	This method sets the P Value for the specified PID index.	
URL	http://{IP}:{Port}/PID/Set_Pid_PValue	
Parameters	pidIndex(int)	The index of the PID to retrieve (e.g., 0, 1, 2, 3).
	pValue(int)	The P Value to set (e.g., 10).
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/PID/Set_Pid_PValue'

payload = {"reg0": 0, "reg1": 10, "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Pid = Device.PID()

    Pid.Set_Pid_PValue(0,10)

    print(Pid.Get_PID(0).get("P_Value")) #control
```

Description	This method sets the I Value for the specified PID index.	
URL	http://{IP}:{Port}/PID/Set_Pid_IValue	
Parameters	pidIndex(int)	The index of the PID to retrieve (e.g., 0, 1, 2, 3).
	iValue(int)	The I Value to set (e.g., 20).
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/PID/Set_Pid_IValue'

payload = {"reg0": 0, "reg1": 20, "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Pid = Device.PID()

    Pid.Set_Pid_IValue(0,20)

    print(Pid.Get_PID(0).get("I_Value")) #control
```


Description	This method sets the D Value for the specified PID index.	
URL	http://{IP}:{Port}/PID/Set_Pid_DValue	
Parameters	pidIndex(int)	The index of the PID to retrieve (e.g., 0, 1, 2, 3).
	dValue(int)	The D Value to set (e.g., 30).
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/PID/Set_Pid_DValue'

payload = {"reg0": 0, "reg1": 30, "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Pid = Device.PID()

    Pid.Set_Pid_DValue(0,30)

    print(Pid.Get_PID(0).get("D_Value")) #control
```

Description	This method sets the Set Value for the specified PID index.	
URL	http://{IP}:{Port}/PID/Set_Pid_SetValue	
Parameters	pidIndex(int)	The index of the PID to retrieve (e.g., 0, 1, 2, 3).
	setValue(float)	The Set Value to set (e.g., 40).
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/PID/Set_Pid_SetValue'

payload = {"reg0": 0, "reg1": 40, "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Pid = Device.PID()

    Pid.Set_Pid_SetValue(0,40)

    print(Pid.Get_PID(0).get("Set_Value")) #control
```

Description	This method sets the Set Value for the specified PID index.	
URL	http://{IP}:{Port}/PID/Set_Pid_Enable	
Parameters	pidIndex(int)	The index of the PID to retrieve (e.g., 0, 1, 2, 3).
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/PID/Set_Pid_Enable'

payload = {"reg0": 0, "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Pid = Device.PID()

    Pid.Set_Pid_Enable(0)

    print(Pid.Get_PID(0).get("IsEnable")) #control
```

Set Pid Negative Polarity Enable

Description This method enables negative polarity for the specified PID index.

URL `http://{IP}:{Port}/PID/Set_Pid_NegativePolarity_Enable`

Parameters `pidIndex(int)` The index of the PID to retrieve (e.g., 0, 1, 2, 3).

Response

<code>"status"</code>	True or False
<code>"msg"</code>	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/PID/Set_Pid_NegativePolarity_Enable'

payload = {"reg0": 0, "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudioimport NMIEndpoint,NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Pid = Device.PID()

    Pid.Set_Pid_NegativePolarity_Enable(0)

    print(Pid.Get_PID(0).get("IsNegativePolarity")) #control
```

Set Pid Invert Vz Enable

Description This method enables the Invert Vz functionality for the specified PID index.

URL `http://{IP}:{Port}/PID/Set_Pid_InvertVz_Enable`

Parameters `pidIndex(int)` The index of the PID to retrieve (e.g., 0, 1, 2, 3).

Response

<code>"status"</code>	True or False
<code>"msg"</code>	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/PID/Set_Pid_InvertVz_Enable'

payload = {"reg0": 0, "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudioimport NMIEndpoint,NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Pid = Device.PID()

    Pid.Set_Pid_InvertVz_Enable(0)

    print(Pid.Get_PID(0).get("IsInvertedVz")) #control
```

Description	This method disables the specified PID based on the provided PID index.	
URL	http://{IP}:{Port}/PID/Set_Pid_Disable	
Parameters	pidIndex(int)	The index of the PID to retrieve (e.g., 0, 1, 2, 3).
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/PID/Set_Pid_Disable'

payload = {"reg0": 0, "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    Pid = Device.PID()

    Pid.Set_Pid_Disable(0)

    print(Pid.Get_PID(0).get("IsEnable")) #control
```

Set Pid Negative Polarity Disable

Description	This method disables the negative polarity for the specified PID based on the provided PID index.	
URL	http://{IP}:{Port}/PID/Set_Pid_NegativePolarity_Disable	
Parameters	pidIndex(int)	The index of the PID to retrieve (e.g., 0, 1, 2, 3).
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/PID/Set_Pid_NegativePolarity_Disable'

payload = {"reg0": 0, "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP, port)

    Device = NMIDevice(Endpoint)

    Pid = Device.PID()

    Pid.Set_Pid_NegativePolarity_Disable(0)

    print(Pid.Get_PID(0).get("IsNegativePolarity")) #control
```

Set Pid Invert Vz Disable

Description	This method disables the Invert Vz functionality for the specified PID based on the provided PID index.	
URL	http://{IP}:{Port}/PID/Set_Pid_InvertVz_Disable	
Parameters	pidIndex(int)	The index of the PID to retrieve (e.g., 0, 1, 2, 3).
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/PID/Set_Pid_InvertVz_Disable'

payload = {"reg0": 0, "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP, port)

    Device = NMIDevice(Endpoint)

    Pid = Device.PID()

    Pid.Set_Pid_InvertVz_Disable(0)

    print(Pid.Get_PID(0).get("IsInvertedVz")) #control
```


AutoTune Controller

Get Commands

Description This method retrieves the list of available commands.

URL http://{IP}:{Port}/AutoTune/Get_Commands

Parameters None

Response

"status"	True or False
"Commands"	List of available commands.
"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/AutoTune/Get_Commands'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP, port)

    Device = NMIDevice(Endpoint)

    AutoTune = Device.AUTOTUNE()

    print(AutoTune.Get_Commands())
```

Description	This method checks if the AutoTune system has been initialized.	
URL	http://{IP}:{Port}/AutoTune/Get_IsInit	
Parameters	None	
Response	"status"	True or False
	"IsInit"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/AutoTune/Get_IsInit'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    AutoTune = Device.AUTOTUNE()

    print(AutoTune.Get_IsInit())
```

Description	This method retrieves the current excitation value.	
URL	http://{IP}:{Port}/AutoTune/Get_Excitation	
Parameters	None	
Response	"status"	True or False
	"Excitation"	The current excitation value (e.g., 6).
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/AutoTune/Get_Excitation'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    AutoTune = Device.AUTOTUNE()

    print(AutoTune.Get_Excitation())
```

Get Excitation Percent

Description	This method retrieves the current excitation value as a percentage.	
URL	http://{IP}:{Port}/AutoTune/Get_ExcitationPercent	
Parameters	None	
Response	"status"	True or False
	"Excitation"	The current excitation value as a percentage (e.g., 30).
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/AutoTune/Get_ExcitationPercent'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    AutoTune = Device.AUTOTUNE()

    print(AutoTune.Get_ExcitationPercent())
```

Get Frequency Start In Hertz

Description	This method retrieves the starting frequency in hertz (Hz).	
URL	http://{IP}:{Port}/AutoTune/Get_FrequencyStartInHertz	
Parameters	None	
Response	"status"	True or False
	"StartInHertz"	A value representing the starting frequency in hertz (e.g., 100000 Hz).
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/AutoTune/Get_FrequencyStartInHertz'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    AutoTune = Device.AUTOTUNE()

    print(AutoTune.Get_FrequencyStartInHertz())
```

Get Frequency End In Hertz

Description	This method retrieves the ending frequency in hertz (Hz).	
URL	http://{IP}:{Port}/AutoTune/Get_FrequencyEndInHertz	
Parameters	None	
Response	"status"	True or False
	"EndInHertz"	A value representing the ending frequency in hertz (e.g., 300000 Hz).
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/AutoTune/Get_FrequencyEndInHertz'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    AutoTune = Device.AUTOTUNE()

    print(AutoTune.Get_FrequencyEndInHertz())
```

Get Frequency Increment In Hertz

Description	This method retrieves the frequency increment in hertz (Hz).	
URL	http://{IP}:{Port}/AutoTune/Get_FrequencyIncrementInHertz	
Parameters	None	
Response	"status"	True or False
	"IncrementInHertz"	A value representing the frequency increment in hertz (e.g.,1000 Hz).
	"msg"	"OK!"

Request Example

```
import requests
headers = {'accept': 'application/json',}
uri = 'http://{IP}:{Port}/AutoTune/Get_FrequencyIncrementInHertz'
response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMEndpoint, NMIDevice
if __name__ == "__main__":
    Endpoint = NMEndpoint(IP,port)
    Device = NMIDevice(Endpoint)
    AutoTune = Device.AUTOTUNE()
    print(AutoTune.Get_FrequencyIncrementInHertz())
```

Get Frequency Slope Types

Description	This method retrieves the available slope types.	
URL	http://{IP}:{Port}/AutoTune/Get_FrequencySlopeTypes	
Parameters	None	
Response	"status"	True or False
	"SlopeTypes"	A string listing the available slope types (e.g., "MaxSlope;MinSlope;MaxRms;Custom").
	"msg"	"OK!"

Request Example

```
import requests
headers = {'accept': 'application/json',}
uri = 'http://{IP}:{Port}/AutoTune/Get_FrequencySlopeTypes'
response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice
if __name__ == "__main__":
    Endpoint = NMIEndpoint(IP,port)
    Device = NMIDevice(Endpoint)
    AutoTune = Device.AUTOTUNE()
    print(AutoTune.Get_FrequencySlopeTypes())
```


Get Frequency Slope Type

Description	This method retrieves the currently selected slope type.	
URL	http://{IP}:{Port}/AutoTune/Get_FrequencySlopeType	
Parameters	None	
Response	"status"	True or False
	"SlopeType"	A string representing the currently selected slope type (e.g., "MinSlope").
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/AutoTune/Get_FrequencySlopeType'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    AutoTune = Device.AUTOTUNE()

    print(AutoTune.Get_FrequencySlopeType())
```

Get Is Center Span

Description	This method checks if the system is currently configured to use the Center Span mode.	
URL	http://{IP}:{Port}/AutoTune/Get_IsCenterSpan	
Parameters	None	
Response	"status"	True or False
	"IsCenterSpan"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/AutoTune/Get_IsCenterSpan'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    AutoTune = Device.AUTOTUNE()

    print(AutoTune.Get_IsCenterSpan())
```

Description	This method retrieves the start delay value.	
URL	http://{IP}:{Port}/AutoTune/Get_StartDelay	
Parameters	None	
Response	"status"	True or False
	"StartDelay"	A value representing the start delay in milliseconds (e.g., 300 ms).
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/AutoTune/Get_StartDelay'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    AutoTune = Device.AUTOTUNE()

    print(AutoTune.Get_StartDelay())
```

Description	This method retrieves the delay value.	
URL	http://{IP}:{Port}/AutoTune/Get_Delay	
Parameters	None	
Response	"status"	True or False
	"Delay"	A value representing the delay in milliseconds (e.g., 1 ms).
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/AutoTune/Get_Delay'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    AutoTune = Device.AUTOTUNE()

    print(AutoTune.Get_Delay())
```

Description	This method checks if the AutoTune system is currently in the process of tuning.	
URL	http://{IP}:{Port}/AutoTune/Get_IsTunning	
Parameters	None	
Response	"status"	True or False
	"Rms"	A value representing the max rms value (e.g., 0).
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/AutoTune/Get_IsTunning'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP, port)

    Device = NMIDevice(Endpoint)

    AutoTune = Device.AUTOTUNE()

    print(AutoTune.StartTune())

    time.sleep(5)

    while(True):

        isTunning = (bool)(AutoTune.Get_IsTunning()["IsTunning"])

        if(isTunning==False):

            break

        time.sleep(0.5)

        print(isTunning)
```

Description	This method retrieves the maximum rms value.	
URL	http://{IP}:{Port}/AutoTune/Get_MaxRms	
Parameters	None	
Response	"status"	True or False
	"Rms"	A value representing the max rms value (e.g., 0).
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/AutoTune/Get_MaxRms'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    AutoTune = Device.AUTOTUNE()

    print(AutoTune.Get_MaxRms())
```

Get Max Rms Frequency

Description	This method retrieves the frequency at which the maximum rms value.	
URL	http://{IP}:{Port}/AutoTune/Get_MaxRmsFrequency	
Parameters	None	
Response	"status"	True or False
	"Frequency"	A value representing the frequency in hertz (Hz) at which the maximum rms was observed (e.g., 30000 Hz).
	"msg"	"OK!"

Request Example

```
import requests
headers = {'accept': 'application/json',}
uri = 'http://{IP}:{Port}/AutoTune/Get_MaxRmsFrequency'
response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice
if __name__ == "__main__":
    Endpoint = NMIEndpoint(IP, port)
    Device = NMIDevice(Endpoint)
    AutoTune = Device.AUTOTUNE()
    print(AutoTune.Get_MaxRmsFrequency())
```

Get Min Slope Frequency

Description	This method retrieves the frequency at which the minimum slope.	
URL	http://{IP}:{Port}/AutoTune/Get_MinSlopeFrequency	
Parameters	None	
Response	"status"	True or False
	"Frequency"	A numeric value representing the frequency in hertz (Hz) at the minimum slope was observed (e.g., 0 Hz).
	"msg"	"OK!"

Request Example

```
import requests
headers = {'accept': 'application/json',}
uri = 'http://{IP}:{Port}/AutoTune/Get_MinSlopeFrequency'
response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice
if __name__ == "__main__":
    Endpoint = NMIEndpoint(IP,port)
    Device = NMIDevice(Endpoint)
    AutoTune = Device.AUTOTUNE()
    print(AutoTune.Get_MinSlopeFrequency())
```


Get Min Slope Rms

Description	This method retrieves the rms value at the frequency where the minimum slope.	
URL	http://{IP}:{Port}/AutoTune/Get_MinSlopeRms	
Parameters	None	
Response	"status"	True or False
	"Rms"	A value representing the rms value at the frequency of the minimum slope (e.g., 0).
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/AutoTune/Get_MinSlopeRms'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    AutoTune = Device.AUTOTUNE()

    print(AutoTune.Get_MinSlopeRms())
```

Get Max Slope Rms

Description	This method retrieves the rms value at the frequency where the maximum slope.	
URL	http://{IP}:{Port}/AutoTune/Get_MaxSlopeRms	
Parameters	None	
Response	"status"	True or False
	"Rms"	A value representing the rms value at the frequency of the maximum slope (e.g., 0).
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/AutoTune/Get_MaxSlopeRms'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    AutoTune = Device.AUTOTUNE()

    print(AutoTune.Get_MaxSlopeRms())
```

Get Max Slope Frequency

Description	This method retrieves the frequency at which the maximum slope.	
URL	http://{IP}:{Port}/AutoTune/Get_MaxSlopeFrequency	
Parameters	None	
Response	"status"	True or False
	"Frequency"	A numeric value representing the frequency in hertz (Hz) at the maximum slope was observed.
	"msg"	"OK!"

Request Example

```
import requests
headers = {'accept': 'application/json',}
uri = 'http://{IP}:{Port}/AutoTune/Get_MaxSlopeFrequency'
response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice
if __name__ == "__main__":
    Endpoint = NMIEndpoint(IP, port)
    Device = NMIDevice(Endpoint)
    AutoTune = Device.AUTOTUNE()
    print(AutoTune.Get_MaxSlopeFrequency())
```

Get Coarse Rms Series

Description	This method retrieves the coarse rms series data.	
URL	http://{IP}:{Port}/AutoTune/Get_CoarseRmsSeries	
Parameters	None	
Response	"status"	True or False
	"Keys"	A list of frequency values representing the series keys.
	"Values"	A list of rms values corresponding to the keys.
	"msg"	"OK!"

Request Example

```
import requests
headers = {'accept': 'application/json',}
uri = 'http://{IP}:{Port}/AutoTune/Get_CoarseRmsSeries'
response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice
if __name__ == "__main__":
    Endpoint = NMIEndpoint(IP,port)
    Device = NMIDevice(Endpoint)
    AutoTune = Device.AUTOTUNE()
    print(AutoTune.Get_CoarseRmsSeries())
```

Get Coarse Phase Series

Description	This method retrieves the coarse phase series data.	
URL	http://{IP}:{Port}/AutoTune/Get_CoarsePhaseSeries	
Parameters	None	
Response	"status"	True or False
	"Keys"	A list of frequency values representing the series keys.
	"Values"	A list of phase values corresponding to the keys.
	"msg"	"OK!"

Request Example

```
import requests
headers = {'accept': 'application/json',}
uri = 'http://{IP}:{Port}/AutoTune/Get_CoarsePhaseSeries'
response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice
if __name__ == "__main__":
    Endpoint = NMIEndpoint(IP,port)
    Device = NMIDevice(Endpoint)
    AutoTune = Device.AUTOTUNE()
    print(AutoTune.Get_CoarsePhaseSeries())
```

Description	This method retrieves the fine RMS series data.	
URL	http://{IP}:{Port}/AutoTune/Get_FineRmsSeries	
Parameters	None	
Response	"status"	True or False
	"Keys"	A list of frequency values representing the series keys.
	"Values"	A list of rms values corresponding to the keys.
	"msg"	"OK!"

Request Example

```
import requests
headers = {'accept': 'application/json',}
uri = 'http://{IP}:{Port}/AutoTune/Get_FineRmsSeries'
response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice
if __name__ == "__main__":
    Endpoint = NMIEndpoint(IP,port)
    Device = NMIDevice(Endpoint)
    AutoTune = Device.AUTOTUNE()
    print(AutoTune.Get_FineRmsSeries())
```

Description	This method retrieves the fine phase series data.	
URL	http://{IP}:{Port}/AutoTune/Get_FinePhaseSeries	
Parameters	None	
Response	"status"	True or False
	"Keys"	A list of frequency values representing the series keys.
	"Values"	A list of rms values corresponding to the keys.
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/AutoTune/Get_FinePhaseSeries'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    AutoTune = Device.AUTOTUNE()

    print(AutoTune.Get_FinePhaseSeries())
```

Description This method initializes the AutoTune system.

URL `http://{IP}:{Port}/AutoTune/Set_Initialize`

Parameters None

Response

"status"	True or False
"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/AutoTune/Set_Initialize'

response = requests.post(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP, port)

    Device = NMIDevice(Endpoint)

    AutoTune = Device.AUTOTUNE()

    AutoTune.Set_Initialize()

    print(AutoTune.Get_IsInit()) #control
```


Set Excitation Percent

Description	This method sets the excitation percentage.	
URL	http://{IP}:{Port}/AutoTune/Set_ExcitationPercent	
Parameters	excitationPercent(float)	The excitation percentage to set (e.g., 30.0).
Response	"status" "msg"	True or False "OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/AutoTune/Set_ExcitationPercent'

payload = {"reg0": 30, "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP, port)

    Device = NMIDevice(Endpoint)

    AutoTune = Device.AUTOTUNE()

    AutoTune.Set_ExcitationPercent(30)

    print(AutoTune.Get_ExcitationPercent()) #control
```

Description	This method sets the excitation value for the current process.	
URL	http://{IP}:{Port}/AutoTune/Set_Excitation	
Parameters	excitation(float)	The excitation value to set (e.g., 1.0).
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/AutoTune/Set_Excitation'

payload = {"reg0": 1, "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    AutoTune = Device.AUTOTUNE()

    AutoTune.Set_Excitation(1)

    print(AutoTune.Get_Excitation()) #control
```

Set Frequency Start In Hertz

Description	This method sets the starting frequency in hertz (Hz).	
URL	http://{IP}:{Port}/AutoTune/Set_FrequencyStartInHertz	
Parameters	hertz(float)	The starting frequency in hertz (e.g., 100000.0).
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/AutoTune/Set_FrequencyStartInHertz'

payload = {"reg0": 100000, "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    AutoTune = Device.AUTOTUNE()

    AutoTune.Set_FrequencyStartInHertz(100000)

    print(AutoTune.Get_FrequencyStartInHertz()) #control
```

Set Frequency End In Hertz

Description	This method sets the ending frequency in hertz (Hz).	
URL	http://{IP}:{Port}/AutoTune/Set_FrequencyEndInHertz	
Parameters	hertz(float)	The ending frequency in hertz (e.g., 300000.0).
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/AutoTune/Set_FrequencyEndInHertz'

payload = {"reg0": 300000, "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    AutoTune = Device.AUTOTUNE()

    AutoTune.Set_FrequencyEndInHertz(300000)

    print(AutoTune.Get_FrequencyEndInHertz()) #control
```

Set Frequency Increment In Hertz

Description	This method sets the frequency increment in hertz (Hz).	
URL	http://{IP}:{Port}/AutoTune/Set_FrequencyIncrementInHertz	
Parameters	hertz(float)	The frequency increment in hertz (e.g., 1000.0).
Response	"status"	True or False
	"IncrementInHertz"	The frequency increment in hertz.
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/AutoTune/Set_FrequencyIncrementInHertz'

payload = {"reg0": 1000, "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    AutoTune = Device.AUTOTUNE()

    AutoTune.Set_FrequencyIncrementInHertz(1000)

    print(AutoTune.Get_FrequencyIncrementInHertz()) #control
```

Description	This method sets the delay in milliseconds.	
URL	http://{IP}:{Port}/AutoTune/Set_Delay	
Parameters	delay(float)	The delay time in milliseconds (e.g., 1.0).
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/AutoTune/Set_Delay'

payload = {"reg0": 1, "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    AutoTune = Device.AUTOTUNE()

    AutoTune.Set_Delay(1)

    print(AutoTune.Get_Delay()) #control
```

Description	This method sets the start delay in milliseconds.	
URL	http://{IP}:{Port}/AutoTune/Set_StartDelay	
Parameters	delay(float)	The start delay time in milliseconds (e.g., 300.0).
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/AutoTune/Set_StartDelay'

payload = {"reg0": 300, "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    AutoTune = Device.AUTOTUNE()

    AutoTune.Set_StartDelay(300)

    print(AutoTune.Get_StartDelay()) #control
```

Set Frequency Slope

Description	This method sets the frequency slope type.	
URL	http://{IP}:{Port}/AutoTune/Set_FrequencySlope	
Parameters	slopeType(str)	The slope type to set (e.g., "MaxSlope", "MinSlope", "MaxRms", "Custom").
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/AutoTune/Set_FrequencySlope'

payload = {"reg0": "MinSlope", "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    AutoTune = Device.AUTOTUNE()

    AutoTune.Set_FrequencySlope("MinSlope")

    print(AutoTune.Get_FrequencySlopeType()) #control
```


Set Center Span Type

Description This method sets the CenterSpanType mode in the auto tune system.

URL `http://{IP}:{Port}/AutoTune/Set_CenterSpanType`

Parameters None

Response

"status"	True or False
"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/AutoTune/Set_CenterSpanType'

payload = {"reg0": "", "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP, port)

    Device = NMIDevice(Endpoint)

    AutoTune = Device.AUTOTUNE()

    AutoTune.Set_CenterSpanType()
```

Set Start End Type

Description This method sets the StartEndType mode in the auto tune system.

URL `http://{IP}:{Port}/AutoTune/Set_StartEndType`

Parameters None

Response

"status"	True or False
"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/AutoTune/Set_StartEndType'

payload = {"reg0": "", "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    AutoTune = Device.AUTOTUNE()

    AutoTune.Set_StartEndType()
```

Description This method initiates the tuning process.

URL http://{IP}:{Port}/AutoTune/StartTune

Parameters None

Response "status" True or False
 "msg" "OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/AutoTune/StartTune'

response = requests.post(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP, port)

    Device = NMIDevice(Endpoint)

    AutoTune = Device.AUTOTUNE()

    print(AutoTune.StartTune())
```

Stop Tune

Description This method stops the tuning process.

URL `http://{IP}:{Port}/AutoTune/StopTune`

Parameters None

Response

"status"	True or False
"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/AutoTune/StopTune'

response = requests.post(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP, port)

    Device = NMIDevice(Endpoint)

    AutoTune = Device.AUTOTUNE()

    print(AutoTune.StopTune())
```

PhotoDiodeController

Get Commands

Description This method retrieves the list of available commands.

URL http://{IP}:{Port}/PhotoDiode/Get_Commands

Parameters None

Response

"status"	True or False
"Commands"	List of available commands.
"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/PhotoDiode/Get_Commands'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP, port)

    Device = NMIDevice(Endpoint)

    PhotoDiode = Device.PHOTODIODE()

    print(PhotoDiode.Get_Commands())
```

Get Is Laser Enabled

Description	This method checks if the laser is currently enabled.	
URL	http://{IP}:{Port}/PhotoDiode/Get_IsLaserEnabled	
Parameters	None	
Response	"status"	True or False
	"IsLaserEnabled"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/PhotoDiode/Get_IsLaserEnabled'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    PhotoDiode = Device.PHOTODIODE()

    print(PhotoDiode.Get_IsLaserEnabled())
```

Description	This method retrieves the current laser power percentage.		
URL	http://{IP}:{Port}/PhotoDiode/Get_LaserPower		
Parameters	None		
Response	"status"	True or False	
	"LaserPowerPercent"	A value representing the current laser power in percentage (e.g., 33).	
	"msg"	"OK!"	

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/PhotoDiode/Get_LaserPower'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    PhotoDiode = Device.PHOTODIODE()

    print(PhotoDiode.Get_LaserPower())
```

Get Laser RF Frequency

Description	This method retrieves the current RF frequency percentage.	
URL	http://{IP}:{Port}/PhotoDiode/Get_LaserRF_Frequency	
Parameters	None	
Response	"status"	True or False
	"RFFrequencyPercent"	A value representing the RF frequency as a percentage (e.g., 1.8).
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/PhotoDiode/Get_LaserRF_Frequency'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    PhotoDiode = Device.PHOTODIODE()

    print(PhotoDiode.Get_LaserRF_Frequency())
```


Description	This method retrieves the FN10 value.	
URL	http://{IP}:{Port}/PhotoDiode/Get_FN10	
Parameters	None	
Response	"status"	True or False
	"FN10"	A value representing the FN10 parameter (e.g., 0).
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/PhotoDiode/Get_FN10'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    PhotoDiode = Device.PHOTODIODE()

    print(PhotoDiode.Get_FN10())
```

Description	This method retrieves the FN value	
URL	http://{IP}:{Port}/PhotoDiode/Get_FN	
Parameters	None	
Response	"status"	True or False
	"FN"	A value representing the FN parameter (e.g., 0).
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/PhotoDiode/Get_FN'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP, port)

    Device = NMIDevice(Endpoint)

    PhotoDiode = Device.PHOTODIODE()

    print(PhotoDiode.Get_FN())
```

Description	This method retrieves the FT value	
URL	http://{IP}:{Port}/PhotoDiode/Get_FT	
Parameters	None	
Response	"status"	True or False
	"FL"	A value representing the FT parameter (e.g., 0).
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/PhotoDiode/Get_FT'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    PhotoDiode = Device.PHOTODIODE()

    print(PhotoDiode.Get_FT())
```

Description	This method sets the laser power percentage.	
URL	http://{IP}:{Port}/PhotoDiode/Set_LaserPower	
Parameters	power(float)	The laser power percentage to set (e.g., 20.5).
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/PhotoDiode/Set_LaserPower'

payload = {"reg0": 20.5, "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    PhotoDiode = Device.PHOTODIODE()

    PhotoDiode.Set_LaserPower(20.5)

    print(PhotoDiode.Get_LaserPower())
```

Set Laser RF Frequency

Description	This method sets the RF percentage.	
URL	http://{IP}:{Port}/PhotoDiode/Set_LaserRF_Frequency	
Parameters	rf(float)	The RF frequency percentage to set (e.g., 21.5).
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/PhotoDiode/Set_LaserRF_Frequency'

payload = {"reg0": 20.5, "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP, port)

    Device = NMIDevice(Endpoint)

    PhotoDiode = Device.PHOTODIODE()

    PhotoDiode.Set_LaserRF_Frequency(21.5)

    print(PhotoDiode.Get_LaserRF_Frequency()) #control
```

Description	This method nullifies the FL value.	
URL	http://{IP}:{Port}/PhotoDiode/NullFL	
Parameters	None	
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/PhotoDiode/NullFL'

payload = {"reg0": "", "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP, port)

    Device = NMIDevice(Endpoint)

    PhotoDiode = Device.PHOTODIODE()

    print(PhotoDiode.NullFL())
```

Description	This method nullifies the value.	
URL	http://{IP}:{Port}/PhotoDiode/Null10FN	
Parameters	None	
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/PhotoDiode/Null10FN'

payload = {"reg0": "", "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP, port)

    Device = NMIDevice(Endpoint)

    PhotoDiode = Device.PHOTODIODE()

    print(PhotoDiode.Null10FN())
```

Description This method resets the PhotoDiode system to its default state.

URL http://{IP}:{Port}/PhotoDiode/PhotoDiodeReset

Parameters None

Response "status" True or False
"msg" "OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/PhotoDiode/PhotoDiodeReset'

payload = {"reg0": "", "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP, port)

    Device = NMIDevice(Endpoint)

    PhotoDiode = Device.PHOTODIODE()

    print(PhotoDiode.PhotoDiodeReset())
```


Set Laser Enable

Description	This method enables the laser.	
URL	http://{IP}:{Port}/PhotoDiode/Set_LaserEnable	
Parameters	None	
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/PhotoDiode/Set_LaserEnable'

payload = {"reg0": "", "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP, port)

    Device = NMIDevice(Endpoint)

    PhotoDiode = Device.PHOTODIODE()

    PhotoDiode.Set_LaserEnable()

    print(PhotoDiode.Get_IsLaserEnabled()) #control
```

Description	This method disables the laser.	
URL	http://{IP}:{Port}/PhotoDiode/Set_LaserDisable	
Parameters	None	
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/PhotoDiode/Set_LaserDisable'

payload = {"reg0": "", "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP, port)

    Device = NMIDevice(Endpoint)

    PhotoDiode = Device.PHOTODIODE()

    PhotoDiode.Set_LaserDisable()

    print(PhotoDiode.Get_IsLaserEnabled()) #control
```

Bias Controller

Get Commands

Description This method retrieves the list of available commands.

URL http://{IP}:{Port}/VBias/Get_Commands

Parameters None

Response

"status"	True or False
"Commands"	List of available commands.
"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/VBias/Get_Commands'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP, port)

    Device = NMIDevice(Endpoint)

    VBias = Device.VBIAS()

    print(VBias.Get_Commands())
```

Description	This method retrieves the current DC offset value.	
URL	http://{IP}:{Port}/VBias/Get_DCOffset	
Parameters	None	
Response	"status"	True or False
	"DCOffset"	A value representing the DC offset.
	"Unit"	The unit of measurement.
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/VBias/Get_DCOffset'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    VBias = Device.VBIAS()

    print(VBias.Get_DCOffset())
```

Description	This method retrieves the current AC amplitude value.	
URL	http://{IP}:{Port}/VBias/Get_ACAmplitude	
Parameters	None	
Response	"status"	True or False
	"ACAmplitude"	A value representing the AC amplitude.
	"Unit"	The unit of measurement.
	"msg"	"OK!"

Request Example

```
import requests
headers = {'accept': 'application/json',}
uri = 'http://{IP}:{Port}/VBias/Get_ACAmplitude'
response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice
if __name__ == "__main__":
    Endpoint = NMIEndpoint(IP, port)
    Device = NMIDevice(Endpoint)
    VBias = Device.VBIAS()
    print(VBias.Get_ACAmplitude())
```

Description	This method retrieves the current AC frequency value.	
URL	http://{IP}:{Port}/VBias/Get_ACFrequency	
Parameters	None	
Response	"status"	True or False
	"ACFrequency"	A value representing the AC frequency.
	"Unit"	The unit of measurement.
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/VBias/Get_ACFrequency'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    VBias = Device.VBIAS()

    print(VBias.Get_ACFrequency())
```

Description	This method retrieves the minimum allowable DC offset value.	
URL	http://{IP}:{Port}/VBias/Get_MinDCOffset	
Parameters	None	
Response	"status"	True or False
	"MinDCOffset"	A value representing the minimum allowable DC offset.
	"Unit"	The unit of measurement.
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/VBias/Get_MinDCOffset'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    VBias = Device.VBIAS()

    print(VBias.Get_MinDCOffset())
```

Description	This method retrieves the maximum allowable DC offset value.	
URL	http://{IP}:{Port}/VBias/Get_MaxDCOffset	
Parameters	None	
Response	"status"	True or False
	"MaxDCOffset"	A value representing the maximum allowable DC offset.
	"Unit"	The unit of measurement.
	"msg"	"OK!"

Request Example

```
import requests
headers = {'accept': 'application/json',}
uri = 'http://{IP}:{Port}/VBias/Get_MaxDCOffset'
response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMEndpoint, NMIDevice
if __name__ == "__main__":
    Endpoint = NMEndpoint(IP, port)
    Device = NMIDevice(Endpoint)
    VBias = Device.VBIAS()
    print(VBias.Get_MaxDCOffset())
```


Description	This method sets the DC offset value.	
URL	http://{IP}:{Port}/VBias/Set_DCOffset	
Parameters	dcOffset(float)	The desired DC offset value (e.g., 5).
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/VBias/Set_DCOffset'

payload = {"reg0": 5, "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    VBias = Device.VBIAS()

    VBias.Set_DCOffset(5)

    print(VBias.Get_DCOffset()) #control
```

Set AC Amplitude

Description	This method sets the AC amplitude value	
URL	http://{IP}:{Port}/VBias/Set_ACAmplitude	
Parameters	amplitude(float)	The desired AC amplitude value (e.g., 5).
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/VBias/Set_ACAmplitude'

payload = {"reg0": 5, "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP, port)

    Device = NMIDevice(Endpoint)

    VBias = Device.VBIAS()

    VBias.Set_ACAmplitude(5)

    print(VBias.Get_ACAmplitude()) #control
```

Set AC Frequency

Description	This method sets the AC frequency value	
URL	http://{IP}:{Port}/VBias/Set_ACFrequency	
Parameters	frequency(float)	The desired AC frequency value (e.g., 30000).
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{Port}/VBias/Set_ACFrequency'

payload = {"reg0": 30000, "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    VBias = Device.VBIAS()

    VBias.Set_ACFrequency(30000)

    print(VBias.Get_ACFrequency()) #control
```

WindowController

Get Commands

Description This method retrieves the list of available commands.

URL http://{IP}:{port}/WindowControl/Get_Commands

Parameters None

Response

“status”	True or False
“Commands”	List of available commands.
“msg”	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{port}/WindowControl/Get_Commands'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    WindowControl = Device.WINDOWCONTROLLER()

    print(WindowControl.Get_Commands())
```

Get Opened Windows

Description	This method retrieves a list of all currently opened windows in the application.	
URL	http://{IP}:{port}/WindowControl/Get_OpenedWindows	
Parameters	None	
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{port}/WindowControl/Get_OpenedWindows'

response = requests.get(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP, port)

    Device = NMIDevice(Endpoint)

    WindowControl = Device.WINDOWCONTROLLER()

    print(WindowControl.Get_OpenedWindows())
```

Get Is Opened

Description	This command checks if a specific window is currently opened in the application.	
URL	http://{IP}:{port}/WindowControl/Get_IsOpened	
Parameters	windowTitle(str)	The title of the window to check. (e.g., "Auto Tune")
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{port}/WindowControl/Get_IsOpened'

payload = {"reg0": "Auto Tune", "reg1": "", "reg2": "", "reg3": ""}

response = requests.get(uri, headers=headers, json=payload)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    WindowControl = Device.WINDOWCONTROLLER()

    print(WindowControl.Get_IsOpened("Auto Tune"))
```

Set Minimize Window

Description	This command minimizes a specified window in the application by its title.	
URL	http://{IP}:{port}/WindowControl/Set_MinimizeWindow	
Parameters	windowTitle(str)	The title of the window to minimize (e.g., "Auto Tune").
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{port}/WindowControl/Set_MinimizeWindow'

payload = {"reg0": "Auto Tune", "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    WindowControl = Device.WINDOWCONTROLLER()

    print(WindowControl.Set_MinimizeWindow("Auto Tune"))
```

Set Maximize Window

Description	This command maximizes a specified window in the application by its title.	
URL	http://{IP}:{port}/WindowControl/Set_MaximizeWindow	
Parameters	windowTitle(str)	The title of the window to maximize (e.g., "Auto Tune").
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{port}/WindowControl/Set_MaximizeWindow'

payload = {"reg0": "Auto Tune", "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    WindowControl = Device.WINDOWCONTROLLER()

    print(WindowControl.Set_MaximizeWindow("Auto Tune"))
```


Set Close Window

Description	This command closes a specified window in the application by its title.	
URL	http://{IP}:{port}/WindowControl/Set_CloseWindow	
Parameters	windowTitle(str)	The title of the window to close (e.g., "Auto Tune").
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{port}/WindowControl/Set_CloseWindow'

payload = {"reg0": "Auto Tune", "reg1": "", "reg2": "", "reg3": ""}

response = requests.post(uri, json=payload, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    WindowControl = Device.WINDOWCONTROLLER()

    print(WindowControl.Set_CloseWindow("Auto Tune"))
```

Set Minimize All

Description This command minimizes all currently opened windows in the application.

URL `http://{IP}:{port}/WindowControl/Set_MinimizeAll`

Parameters None

Response "status" True or False
 "msg" "OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{port}/WindowControl/Set_MinimizeAll'

response = requests.post(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP, port)

    Device = NMIDevice(Endpoint)

    WindowControl = Device.WINDOWCONTROLLER()

    print(WindowControl.Set_MinimizeAll())
```

Set Normalize All

Description	This command normalizes all minimized windows in the application, restoring them to their default size and position.	
URL	http://{IP}:{port}/WindowControl/Set_NormalizeAll	
Parameters	None	
Response	"status"	True or False
	"msg"	"OK!"

Request Example

```
import requests

headers = {'accept': 'application/json',}

uri = 'http://{IP}:{port}/WindowControl/Set_NormalizeAll'

response = requests.post(uri, headers=headers)
```

NMIXStudioLibrary Example

Python:

```
from NMIXStudio import NMIEndpoint, NMIDevice

if __name__ == "__main__":

    Endpoint = NMIEndpoint(IP,port)

    Device = NMIDevice(Endpoint)

    WindowControl = Device.WINDOWCONTROLLER()

    print(WindowControl.Set_NormalizeAll())
```