

# system.tag.addTag

This function is used in **Python Scripting**.

## Description

Adds a new tag in Ignition. You can add OPC, memory, expression, query, folder, and UDT instance tags. You can't add Client Tags, because those can vary from project to project.

## Client Permission Restrictions

Permission Type: Tag Editing

Client access to this scripting function is blocked to users that do not meet the role/zone requirements for the above permission type. This function is unaffected when run in the Gateway scope.

## Syntax

This function accepts [keyword arguments](#).

**system.tag.addTag(parentPath, name, tagType, dataType, accessRights, enabled, value, attributes, parameters, overrides, alarmList, alarmConfig)**

- Parameters

[String](#) parentPath - The folder to add the tag to. Leave blank for the root folder.

[String](#) name - The name of the tag.

[String](#) tagType - The type of tag to create. Possible values are OPC, MEMORY, EXPRESSION, QUERY, Folder, and UDT\_INST.

[String](#) dataType - The data type of the tag. Not used for UDT instances or folders.

Possible basic values are Int1, Int2, Int4, Int8, Float4, Float8, Boolean, String, DataSet, and DateTime.

Possible array values are Int4Array, Int8Array, Float8Array, BooleanArray, StringArray, DateTimeArray.

[String](#) accessRights - The access rights for a tag. Possible values are Read\_Only, Read\_Write, and Custom.

[boolean](#) enabled - If true, the tag will be enabled.

[Object](#) value - The value of the tag. Used for memory tags.

[PyDictionary](#) attributes - The tag's configuration attributes.

[PyDictionary](#) parameters - The parameters for a UDT instance tag.

[PyDictionary](#) overrides - All of the overrides for a UDT instance tag.

[String](#) alarmList - List of legacy alarms for the tag. The legacy alarm system was retired in 7.6.0, so the alarmConfig parameter should be utilized on newer versions.

[PyDictionary](#) alarmConfig - The [alarm configuration](#) for the tag. See [editAlarmConfig](#) for details on how to use this parameter.

- Returns

Nothing

- Scope

All

If called in the gateway scope, a tag provider must be specified.

### Associated attributes:

- Complete list of the acceptable [Tag Properties](#) and [alarmConfig](#).

## Code Examples

### Code Snippet

# Example: Add OPC tag

```
system.tag.addTag(parentPath='', name="TagOPC", tagType="OPC",
dataType="Int2", attributes={"OPCServer": "Ignition OPC-UA
Server", "OPCItemPath": "[MLX]N7:0"})
```

### Code Snippet

# Example: Add OPC tag with two alarms

```
system.tag.addTag( parentPath='',
    name="TagOPCALarm",
    tagType="OPC",
    dataType="Int2",
    attributes={"OPCServer": "Ignition OPC-UA Server",
        "OPCItemPath": "[MLX]N7:0"},
    alarmConfig={ "Alarm": [
        ["name", "Value", "Alarm"],
        ["setpointA", "Value", 1.0],
        ["CustomEmailSubject", "Value", "My Subject"]
    ],
    "Alarm2": [
        ["name", "Value", "Alarm2"],
        ["parameterName", "Value", "newValue"],
        ["enabled", "Expression", "1=2"],
        ["CustomEmailMessage", "Value", "My Message"]
    ]
    }
    )
```

### Code Snippet

# Example: Add Folder

```
system.tag.addTag(parentPath='', name="Folder",
tagType="Folder")
```

### Code Snippet

# Example: Add Memory tag

```
system.tag.addTag(parentPath='', name="TagMemory",
tagType="MEMORY", dataType="Int2", value=25)
```

### Code Snippet

```
# Example: Add Memory tag with an Alarm using the alarmList
parameter
alert = "myAlert;Medium_High;0.0;50.0;0;;;2.0;SEC$"

system.tag.addTag(parentPath = '', name = "TagMemory", tagType =
"MEMORY", dataType = "Int2", value = 25, alarmList = alert)
```

### Code Snippet

```
# Example: Add Expression tag

system.tag.addTag(parentPath='', name="TagExpression",
tagType="EXPRESSION", dataType="Int2",
attributes={"Expression": "[~]Tag1 * 20.5"})
```

### Code Snippet

```
# Example: Add Query tag
system.tag.addTag(parentPath='', name="TagQuery",
tagType="QUERY", dataType="DateTime",
attributes={"Expression": "SELECT CURRENT_TIMESTAMP",
"SQLBindingDatasource": "MySQL"})
```

### Code Snippet

```
# Example: Add Memory tag to a provider other than the default
tag provider

system.tag.addTag(parentPath="[ProviderName]Folder",
name="TagMemoryProvider", tagType="MEMORY", dataType="Int2",
value=42)
```

### Code Snippet

```
# Example: Add UDT instance tag

# Before running this script, there must be a UDT named 'Motor'
# that has a string parameter 'DeviceName' and an integer
parameter
# 'MotorNumber'.
system.tag.addTag(parentPath='', name="TagUDT",
tagType="UDT_INST", attributes={"UDTParentType": "Motor"},
parameters={"DeviceName": "CLX", "MotorNumber": 1})
```

### Code Snippet

```
# Example: Add UDT instance tag and override multiple parameters
on status tag.
```

```
# Before running this script, there must be a UDT named
'SecondUDT'
# that has a string parameter 'DeviceName', an integer parameter
# 'MotorNumber', and a tag named "STATUS".
system.tag.addTag(parentPath='', name="TagUDTParameters",
tagType="UDT_INST", attributes={"UDTParentType":"SecondUDT"},
parameters={"DeviceName":"CLX", "MotorNumber":2},
overrides={"STATUS":{"ScanClass":"Default Historical",
"Enabled":"false"}})
```

### Code Snippet

```
# Example: Add UDT instance tag and override the scan class of a
tag inside of another UDT
```

```
# Before running this script, there must be a UDT named
'ThirdUDT'.
# ThirdUDT must contain a UDT 'SecondUDT',
# which has a string parameter 'DeviceName', an integer
parameter
# 'MotorNumber' and a tag named "STATUS".
# Note the nested SecondUDT has the name 'Child_Instance' in the
ThirdUDT.
system.tag.addTag(parentPath='', name="TagUDTScanClass",
tagType="UDT_INST", attributes={"UDTParentType":"ThirdUDT"},
parameters={"Motor/DeviceName":"CLX", "Motor/MotorNumber":1},
overrides={"Child_Instance/STATUS":{"ScanClass":"Default
Historical"}})
```

### Code Snippet

```
# Example: Create a memory tag with a type of DataSet, and pass
in a dataset as the initial value of the newly created tag
```

```
# create an initial dataset to pass the tag
initData=system.dataset.fromCSV('#NAMES"\n"Col
1"\n"#TYPES"\n"I"\n"#ROWS", "1"\n"0"\n')
```

```
# create the tag
system.tag.addTag(parentPath='', value=initData,
name="TagMemoryDataset", tagType="MEMORY", dataType="DataSet")
```

### Code Snippet

```
# Example: Add an Array type Memory tag to the default tag
provider

# create a dataset
dataset = system.dataset.toDataSet(["values"], [[1],[2],[3]])

# create the tag
system.tag.addTag(parentPath='', name="TagMemoryArray",
tagType="MEMORY", dataType="Int4Array", value=dataset)
```