Article
Nicholai Mitchko · Apr 12, 2022 2m read

Embedded Python, Using Output Parameters

Background

In InterSystems IRIS versions >=2021.2 we can use the accompanying irispython binary to directly write python code on top of our IRIS instances. This lets us use python packages, call methods, make SQL queries, and do nearly anything in Objectscript but pythonic.

For example, below I check if a namespace is present:

```
#!/usr/irissys/bin/irispython
import iris
# call arbitrary class methods
result = iris.cls('%SYS.Namespace').Exists('USER')
if result == 1:
    print(f"Namespace USER is present")
```

But, what if my method in IRIS has special parameters such as Output and ByRef parameters? How can we use Output parameters in irispython?

For example, Ens.Director has a lot of Output parameters on it's methods, how can I use these parameters in python?

Ens.Director:GetProductionStatus

```
// An example method stub from Ens.Director
ClassMethod GetProductionStatus(Output pProductionName As %String, Output pState As %
Integer...
```

Trying the normal variable methods

At first glance, you may try the following:

```
import os
# Set namespace the manual way
os.environ['IRISNAMESPACE'] = 'DEMONSTRATION'

import iris

# TEST 1 with output variables
productionName, productionState = None, None
status = iris.cls('Ens.Director').GetProductionStatus(productionName, productionState
)

print("Success? -- {}".format(productionState != None))
```

But both tests won't return the Output Variables! You can try this yourself on any Ensemble namespace

Using iris.ref

The irispython utility iris.ref can be used to capture the Output and ByRef variables.

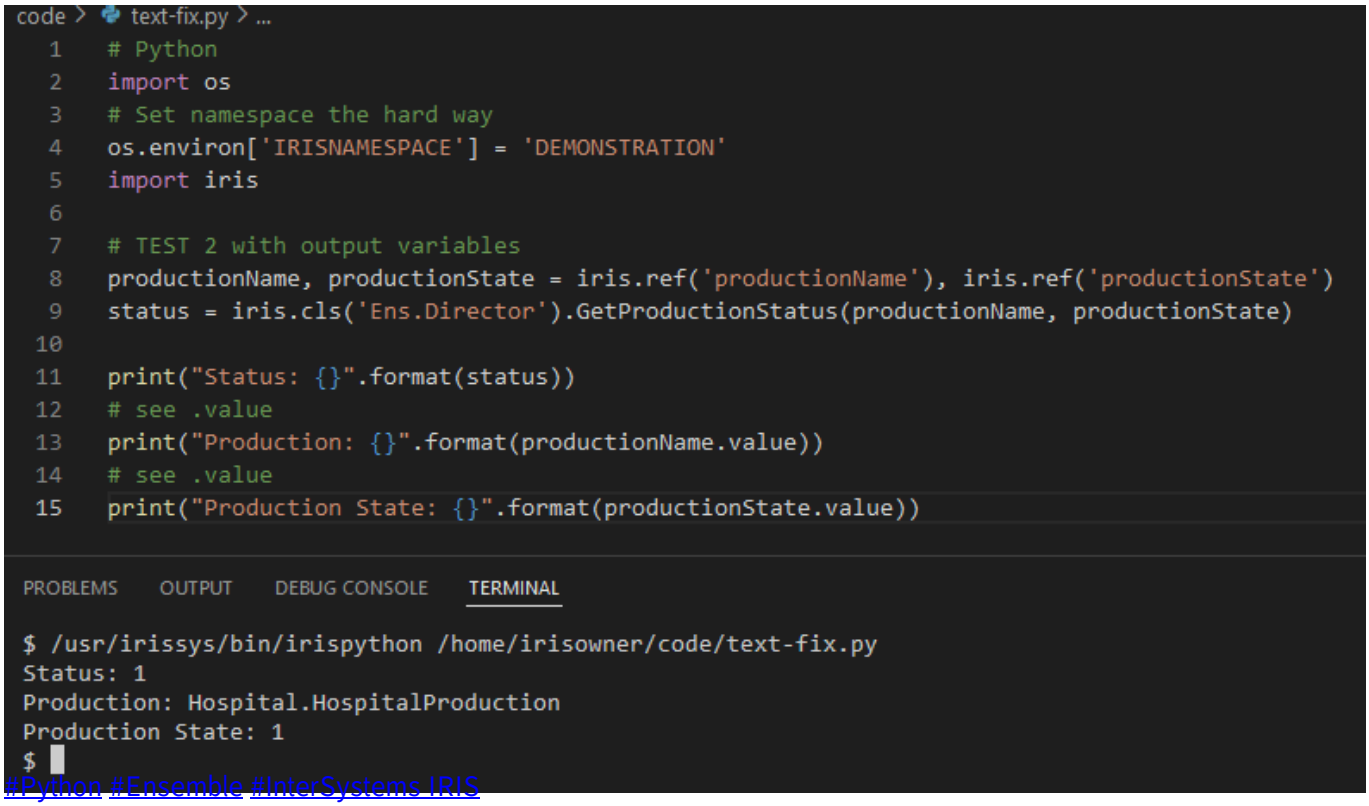
1. Create an iris.ref() object
2. Call your ObjectScript Method
3. Use the .value variable to get the result from that parameter

```
import os
# Set namespace the hard way
os.environ['IRISNAMESPACE'] = 'DEMONSTRATION'

import iris

# TEST 2 with output variables
productionName, productionState = iris.ref('productionName'), iris.ref('productionSta
te')
status = iris.cls('Ens.Director').GetProductionStatus(productionName, productionState
)

print("Status: {}".format(status))
# see .value
print("Production: {}".format(productionName.value))
# see .value
print("Production State: {}".format(productionState.value))
```



#Python #Ensemble #InterSystems IRIS

Source URL:https://community.intersystems.com/post/embedded-python-using-output-parameters