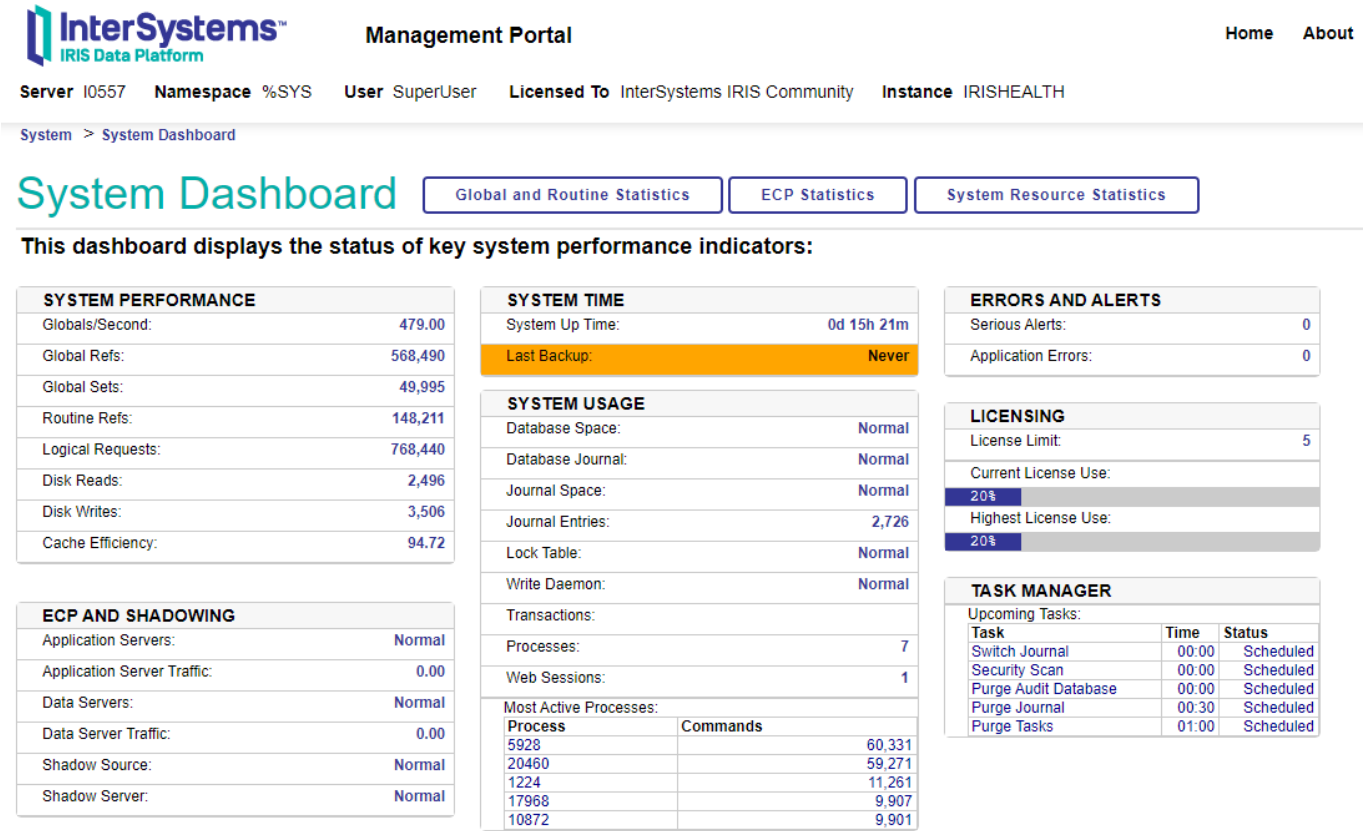


文章
Michael Lei · 七月 2, 2022 阅读大约需 4 分钟

使用嵌入式 python 访问管理门户仪表盘并在网页上显示 cache 表数据

Hi 社区,

在这篇文章中, 我将解释如何通过使用嵌入式python访问管理门户系统的仪表盘信息和表数据。



访问管理门户仪表盘

以下是几个步骤:

- 1: 改变 Namespace 到 %SYS
- 2: 导入 IRIS python 模块并实例化 "SYS.Stats.Dashboard" 类
- 3: 访问实例化class类属性

第1步: 改变 Namespace 到 %SYS

为了获得系统仪表盘信息, 我们需要从%SYS命名空间访问 "SYS.Stats.Dashboard" 类。如果不在%SYS命名空间, 那么我们必须改变命名空间。

创建Embedded.Utils类来获取或设置命名空间:

```
///  
///This class will be used to get or set Namespace
```

```
Class Embedded.Utills
{
//Get Namespace
ClassMethod GetNameSpace() As %Status
{

    Return $namespace
}
//Set Namespace
ClassMethod SetNameSpace(pNameSpace) As %Status
{
    zn pNameSpace
    Return $namespace
}
}
```

用 Python 代码写下以下脚本来改变命名空间

```
#Import iris liabrary
import iris

#Set Namespoace to %SYS if current namespace is not %SYS
if iris.cls("Embedded.Utills").GetNameSpace() != "%SYS":
    iris.cls("Embedded.Utills").SetNameSpace("%SYS")
```

第2步：导入 IRIS python 模块并实例化类

调用类方法 'Sample'来访问 "SYS.Stats.Dashboard" 的属性：

```
?
#Import iris liabrary
import iris

#Set Namespoace to %SYS if current namespace is not %SYS
if iris.cls("Embedded.Utills").GetNameSpace() != "%SYS":
    iris.cls("Embedded.Utills").SetNameSpace("%SYS")

#A class may be instantiated by calling the class method 'Sample', and then the current value of each property can be accessed.
ref = iris.cls("SYS.Stats.Dashboard").Sample()
```

第3步：访问实例化类属性

这个类包含以下可以使用以下代码访问的属性：

Properties			
ApplicationErrors	CSPSessions	CacheEfficiency	DatabaseSpace
DiskReads	DiskWrites	ECPAppServer	ECPAppSrvRate
ECPDataServer	ECPDataSrvRate	GloRefs	GloRefsPerSec
GloSets	JournalEntries	JournalSpace	JournalStatus
LastBackup	LicenseCurrent	LicenseCurrentPct	LicenseHigh
LicenseHighPct	LicenseLimit	LicenseType	LockTable
LogicalReads	Processes	RouRefs	SeriousAlerts
ShadowServer	ShadowSource	SystemUpTime	WriteDaemon

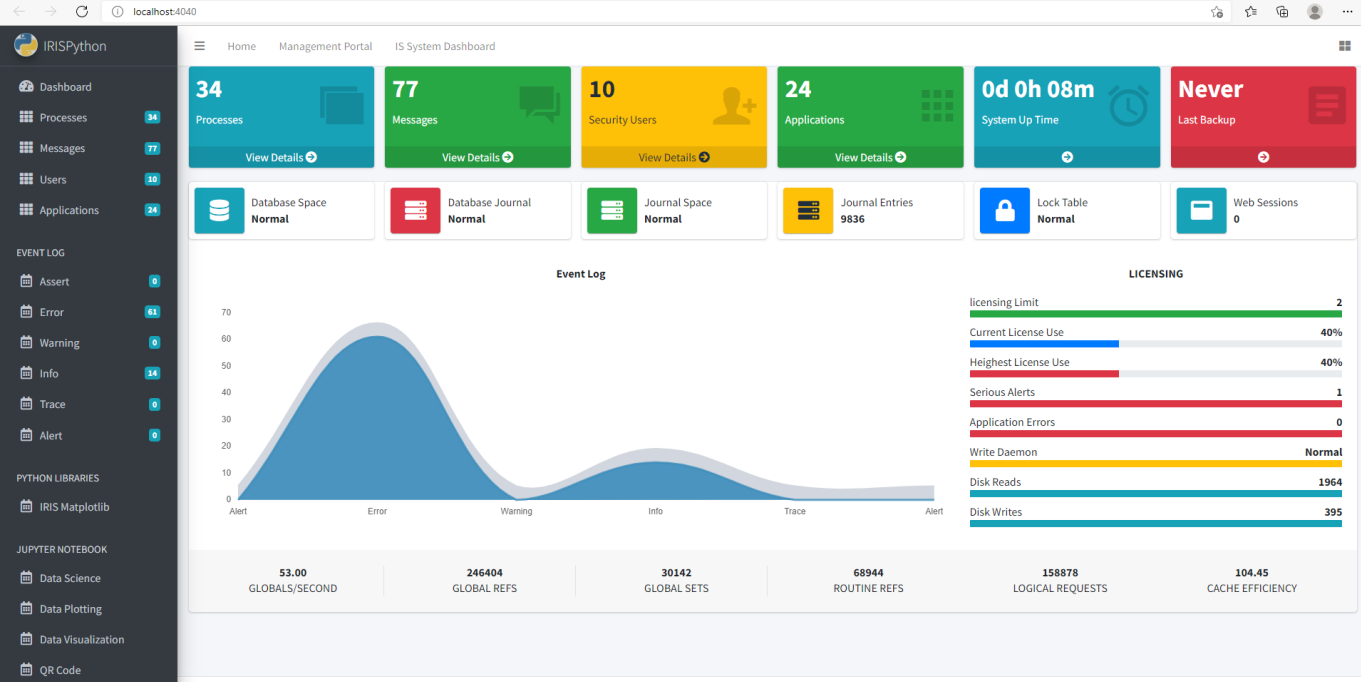
```
?#Import iris liabrury
import iris

#Set Namespoace to %SYS if current namespace is not %SYS
if iris.cls("Embedded.Utills").GetNameSpace() != "%SYS":
    iris.cls("Embedded.Utills").SetNameSpace("%SYS")

#A class may be instantiated by calling the class method 'Sample', and then the curre
nt value of each property can be accessed.
ref = iris.cls("SYS.Stats.Dashboard").Sample()

#display ApplicationErrors
print(ref.ApplicationErrors)
#display CSP Sessions
print(ref.CSPSessions)
```

更多信息欢迎访问 [iris-python-apps](#) 应用



使用jquery 数据表在web 网页显示cache表数据

以下是显示Security.Users 表的步骤

- 1: **改变 Namespace 到 %SYS**
- 2: **导入 IRIS python 模块并使用iris.sql.exec() 功能来运行 SQL SELECT 语句来获得结果集**
- 3: **调用结果集 dataframe 功能来获得 dataframe**
- 4: **从dataframe获得header 和列JSON 数据并传给网页**
- 5: **显示数据到jquery数据表**

第1步: 改变 Namespace 到 %SYS

同样上面的 Embedded.Utils 类可以使用来获取或者设置命名空间

```
#Import iris library
import iris
```

```
#Set Namespoace to %SYS if current namespace is not %SYS
if iris.cls("Embedded.Utils").GetNameSpace() != "%SYS":
    iris.cls("Embedded.Utils").SetNameSpace("%SYS")
```

第2步: 导入 IRIS python 模块并使用iris.sql.exec() 功能来运行 SQL SELECT 语句来获得结果集

导入IRIS模块后我们使用 iris.sql.exec() 来运行 SQL SELECT 语句来获得结果集,

```
import iris

statement = '''SELECT
ID, AccountNeverExpires, AutheEnabled, ChangePassword, CreateDateTime AS DateCreated,
Enabled, ExpirationDate, Flags, Name
FROM Security.Users'''
#Calling embedded python iris.sql.exec class to get result set
resultSet = iris.sql.exec(mySql)
```

第3步: 调用结果集 dataframe 功能来获得 dataframe

```
#Get dataframe by calling resultset dataframe function
dataframe = statement.dataframe()
```

第4步: 从dataframe获得header 和列JSON 数据并传给Web网页

```
#Convert and data to Json by using to_json dataframe method and json loads function
my_data=json.loads(dataframe.to_json(orient="split"))["data"]
#Get columns details
my_cols=[{"title": str(col)} for col in
json.loads(df.to_json(orient="split"))["columns"]]
#render html by passing my_data and my_cols variables which will be used to generate
datatable
return render_template('tablesdata.html', my_data = my_data, my_cols = my_cols)
```

第5步: 显示数据到jquery数据表

用id "myTable" 和javascript来定义表, 并填入传递的header 和列数据

```
<table id="myTable" class="table table-bordered table-striped">
</table>
```

```
<script>
$(document).ready(function() {
    // parse the data to local variable passed from app.py file
    let my_data = JSON.parse('{{ my_data | tojson }}');
    let my_cols = JSON.parse('{{ my_cols | tojson }}');

    $('#myTable').DataTable( {
        "data": my_data,
        "columns": my_cols, "}" );
    } );
```

这样就好了。更多信息欢迎访问 [iris-python-apps](#) 应用

嵌入式Python官方文档 [Embedded Python Overview](#).

谢谢！

[##嵌入式 Python #SQL #管理门户 #InterSystems IRIS for Health](#)

源

URL:

<https://cn.community.intersystems.com/post/%E4%BD%BF%E7%94%A8%E5%B5%8C%E5%85%A5%E5%BC%8F-python-%E8%AE%BF%E9%97%AE%E7%AE%A1%E7%90%86%E9%97%A8%E6%88%B7%E4%BB%AA%E8%A1%A8%E7%9B%98%E5%B9%B6%E5%9C%A8%E7%BD%91%E9%A1%B5%E4%B8%8A%E6%98%BE%E7%A4%BA-cache-%E8%A1%A8%E6%95%B0%E6%8D%AE>