

Cielo Explorer Session – Creating Classes in a Script

Use the Cielo Explorer “)edit my_script_name” to create a Cielo Explorer script. Within that script it is possible to create classes each of which can contain functions. Refer to the example provided below:

CieloSession - Microsoft Visual Studio (Administrator)

File Edit View Project Build Debug Tools Window Help

Start Page Cielo Explorer

Cielo Explorer for Visual Studio - Professional v1
Cielo 1.0.7001
Copyright © VisualCielo. All rights reserved.
clear session

edit sc1

Script 'sc1' updated
classes

```
math      useMath    useMath_2
m=new math()
m.add(10,20)
30
n=new useMath()
n.fn(10,20)
30
```

useMath.fn

```
// Script: sc1 created on 5/25/2009 4:12:30 AM by Joe
public class math {
    public function add(a, b) {
        return a + b
    }
}
public class useMath {

    public function fn(a, b) {
        m = new math()
        return (m.add(a, b))
    }
}
```

Error List

Ready

The documentation installed with VisualAPL includes the ""Language_Interaction.chm" file. The Cielo Explorer commands section about the)classes command contains some typographical errors indicated below. These will be corrected in a future version of that documentation.

New Window Caption

Hide Back Forward Print

Contents Search

- Native File Access
- Cielo Explorer
 - Session Commands
 -)cd
 -)classes**
 -)clear
 -)edit
 -)fns
 -)load
 -)off
 -)run
 -)unf
 -)scripts
 -)vars
 -)xload
 -)xmlout- Menu Reference
- Code Flow Control
- Syntax Characters
- System Functions, Constants, and Variables
- Primitive Scalar Operators
- Component File Access

)classes

Shows the current list of classes which have been defined in the session.

Syntax:

```
)classes
```

Remarks:

The `)classes` command shows the list of classes which have been created in the session.

Classes are most commonly created in the session by running a script file.

Example:

Here is an example script which contains the definition of two classes:

```
// Script: sc1
public class math {
    function add(a, b) {
        return a + b
    }

    function subtract(a, b) {
        return a - b
    }
}

public class useMath {
    function fn(a, b) {
        m = math()
        return m.add(a, b)
    }
}
```

Should be:
public function add(a, b) { ... }

Should be:
public function fn(a,b) { ... }

Should be:
m = new math()

Now lets run the script to create the classes in the session:

```
// display the contents of the )classes list
)classes

// the list is empty

// load and run the script 'sc1'
)load sc1

// display )classes again
)classes
math usemath
// the two classes now exist in the session.

// run the useMath class:
um = useMath() ← Should be:  
um = new useMath()
```

30