

# Repository Function GETMODULE

## Description

Method for retrieving all module identifiers from a repository entity.

## Syntax

```
retval = Repository("GETMODULE" , entID)
```

## Parameters

The **GETMODULE** method has the following parameters.

| Parameter      | Description   |
|----------------|---|
| <i>Message</i> | "CLEARMODULE"   |
| <i>entID</i>   | <i>entID</i> consists of four elements, which are '*' (asterisk) delimited:   |
|                | <ul style="list-style-type: none"><li>• Application name</li><li>• Type ID</li><li>• Class ID</li><li>• Entity name</li></ul> |

## Returns

An @vm-delimited array of module names.

## Remarks

The module name is field 8 within the SYSREPOS record for the Entity. The module name allows for the creation of named modules within OpenInsight. The RDK tool has an option within the Repository View creation to create RDKs based on Modules. The RDK tool will then build deployments based on the module name.

This is available in OpenInsight 9.2 and above. In 9.2 it is a programmatic function only. In 9.2.1, the [Module Manager](#) was introduced to create and organize Modules.

**Note:** Always call the [Get\\_Status function](#) after calling *Repository*.

## See also

[Repository\(\)](#) function, [SETMODULE](#) method, [ADDMODULE](#) method, [REMOVEMODULE](#) method, [CLEARMODULE](#) method

## Example

```

* Retrieve Module Names for an Entity

Declare Function Repository

* Create an array of entities to use Module methods against
reposStack = ""
reposStack := @appid<1> : "**OIWIN**CUSTOMERS" : @fm
reposStack := @appid<1> : "**OIWINEXE**CUSTOMERS" : @fm
reposStack := @appid<1> : "**POPUP**CUSTOMERS"

moduleName = "CUSTOMER_INVOICE"

* Retrieve all Module names from the entities in reposStack

rPos = 0
rFlag = ""
module = ""

Loop
  Remove thisEntity From reposStack at rPos setting rFlag
  var = Repository( "GETMODULE", thisEntity )
  If Get_Status(ErrCode) then
    call msg(@window, 'Error Message returned: ' : ErrCode)
  end
  While rFlag
  Repeat

```