

# Repository Function TCOMPILE

## Description

Method for compiling entities of a given type (stored procedures, events, datasets, or windows) into an executable form, as well as all the entities within it, such as code associated with window events within a window or \$INSERT records in a stored procedure.

## Syntax

```
retval = Repository("TCOMPILE" , entID, parameter1, parameter2, parameter3)
```

The arguments in parameter1, parameter2, and parameter3 vary depending on entity type.

## Parameters

The **TCOMPILE** message has the following parameters

Parameter	Description
<i>Message</i>	<b>"TCOMPILE"</b>
<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>
	Methods are executed as follows: if class specific, execute at the class level; if type specific, execute at the type level; otherwise, execute the method.
<i>parameter1</i>	a boolean indicating whether or not the entity should be compiled if it has not been changed since the last compilation
<i>parameter2</i>	for windows, a boolean indicating whether cleanup of orphan entities should be done. Ignored for other entity types
<i>parameter3</i>	for windows, a boolean indicating whether or not linemarks should be kept.

## Returns

**null** = Compilation failed. Use [Get\\_Status\(\)](#) to test for error return.

**1** = Compilation not needed.

**2** = Compilation successful. Field 2 returns the date and time of compilation.

**Note:** *Stored procedures can be recompiled using the [Recompile\\_Proc](#) subroutine.*

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

## See also

[Repository\(\) function](#), [Recompile\\_Proc](#), [Repository\("COMPILE"\)](#), [Get\\_Repos\\_Entities\(\)](#).

## Example

```

/* Compile the window INV_ENTRY, as well as event code in the window */
Declare Function Repository, Get_Status
AppID = @APPID<1>  ;* current application
TypeID = "OIWIN"  ;* OpenInsight windows
ClassID = ""      ;* Not Applicable
Win_Name = 'INV_ENTRY'
recompile_if_unchanged = ''
cleanup_orphan_entities = ''
creserve_linemarks = ''
entid = AppID : '*' : TypeID : '*' : ClassID : '*' : Win_Name
result = Repository("TCOMPILE", entid, recompile_if_unchanged, cleanup_orphan_entities, preserve_linemarks)
* test result
begin case
  case result = ''
    If Get_Status(ErrCode) then
      call msg(@window, 'Error Message returned: ' : ErrCode)
    end
  case result = 1
    call msg(@window, 'Tree Compilation not needed')
  case result<1> = 2
    call msg(@window, 'Tree compilation successful, compiled at ' : result<2>)
end case

```