

# Root

## Table of Contents

file : [example\\_4.ecl](#) Example : Inheritance across files  
file : [example\\_5.ecl](#)  
file : [test.ecl](#) test module  
file : [example\\_10.ecl](#)  
file : [example\\_3.ecl](#) Documentation Testing Multiline Title  
file : [Math.ecl](#)  
file : [example\\_11.ecl](#)  
file : [example\\_9.ecl](#)  
file : [example\\_7.ecl](#) Basic Type Example  
file : [example\\_2.ecl](#) Basic Inheritance documentation : mod\_3 inherits both mod\_1 and mod\_2  
file : [example.ecl](#) Basic Example with :  
dir : [Inintest](#)  
dir : [intest](#)

# example\_\_4

## IMPORTS

- Inintest.Example\_\_3.mod\_\_1

## DESCRIPTIONS

**MODULE : example\_\_4**

[Up](#)

Example : Inheritance across files mod\_\_1 in Example\_\_4 inherits mod\_\_1 in Example\_\_3

1. [mod\\_\\_1](#)

**MODULE : mod\_\_1**

[Up](#)

1. [v2\\_\\_m1\\_\\_ex3](#)
2. [v2\\_\\_m1\\_\\_ex4](#)

**ATTRIBUTE : v2\_\_m1\_\_ex3**

[Up](#)

**INHERITED : True**

**ATTRIBUTE : v2\_\_m1\_\_ex4**

[Up](#)

example\_\_5

IMPORTS

DESCRIPTIONS

# test

## IMPORTS

## DESCRIPTIONS

**MODULE :** test

[Up](#)

test module

# example\_\_10

## IMPORTS

- intest

## DESCRIPTIONS

**MODULE :** example\_\_10

[Up](#)

1. [mod\\_\\_1](#)

**MODULE :** mod\_\_1

[Up](#)

**INHERITED :** True

# example\_\_3

## IMPORTS

## DESCRIPTIONS

### MODULE : Example\_\_3

[Up](#)

Documentation Testing Multiline Title. [link@myspace.com](mailto:link@myspace.com)

Sentence 1 blablalbla bbbblaaaa

Sentence 2

blablalbla bbbblaaaaaa

bbblaaaaaaaaaa

**Parameter** : first ||| okay\_\_1

**Parameter** : second ||| okay\_\_2

**Parameter** : third ||| okay\_\_3

**Field** : f1 ||| oka\_f1

**Field** : f2 ||| oka\_f2

**Return** : rec\_\_1

**See** : example\_\_1.mod\_\_1

1. [mod\\_\\_1](#)

### MODULE : mod\_\_1

[Up](#)

1. [v1\\_\\_m1](#)
2. [v2\\_\\_m1\\_\\_ex3](#)

3. [long\\_name](#)

**ATTRIBUTE : v1\_m1**

[Up](#)

Doc test 2. Title end by period not newline

ABCD ||||  
CDEF ||||

**ATTRIBUTE : v2\_m1\_ex3**

[Up](#)

DOC Test 3 No Period title

**FUNCTION : long\_name**

(DATASET({REAL8 u}) X, DATASET({REAL8 u}) IntW, DATASET({REAL8 u}) Intb, REAL8  
BETA=0.1, REAL8 sparsityParam=0.1 , REAL8 LAMBDA=0.001, REAL8 ALPHA=0.1, UNSIGNED2  
MaxIter=100)

[Up](#)

# Math

## IMPORTS

## DESCRIPTIONS

**MODULE : Math**

[Up](#)

1. [Infinity](#)
2. [NaN](#)
3. [isInfinite](#)
4. [isNaN](#)
5. [isFinite](#)
6. [FMod](#)
7. [FMatch](#)

**ATTRIBUTE : Infinity**

REAL8

[Up](#)

Return a real "infinity" value.

**ATTRIBUTE : NaN**

REAL8

[Up](#)

Return a non-signalling NaN (Not a Number) value.



## FUNCTION : isInfinite

BOOLEAN (REAL8 val)

[Up](#)

Return whether a real value is infinite (positive or negative).

**Parameter** : val ||| The value to test.

## FUNCTION : isNaN

BOOLEAN (REAL8 val)

[Up](#)

Return whether a real value is a NaN (not a number) value.

**Parameter** : val ||| The value to test.

## FUNCTION : isFinite

BOOLEAN (REAL8 val)

[Up](#)

Return whether a real value is a valid value (neither infinite not NaN).

**Parameter** : val ||| The value to test.

## FUNCTION : FMod

REAL8 (REAL8 numer, REAL8 denom)

[Up](#)

Returns the floating-point remainder of numer/denom (rounded towards zero). If denom is zero, the result depends on the -fdivideByZero flag: 'zero' or unset: return zero. 'nan': return a non-signalling NaN value 'fail': throw an exception

**Parameter** : numer ||| The numerator.

**Parameter** : denom ||| The denominator.

## FUNCTION : FMatch

BOOLEAN (REAL8 a, REAL8 b, REAL8 epsilon=0.0)

[Up](#)

Returns whether two floating point values are the same, within margin of error epsilon.

**Parameter :** a ||| The first value.

**Parameter :** b ||| The second value.

**Parameter :** epsilon ||| The allowable margin of error.

# example\_\_11

## IMPORTS

- Inintest
- Example\_3
- intest.Example\_3
- intest.inintest.Example\_3
- Inintest.Example\_3

## DESCRIPTIONS

**MODULE : example\_\_11**

[Up](#)

1. [Example\\_3](#)

**MODULE : Example\_3**

[Up](#)

**OVERRIDE : True**

1. [mod\\_1](#)

**MODULE : mod\_1**

[Up](#)

1. [v2\\_\\_m1\\_\\_ex3](#)

**ATTRIBUTE : v2\_m1\_ex3**

Up

# example\_\_9

## IMPORTS

- example\_8
- example\_8.mod\_1

## DESCRIPTIONS

# example\_\_7

## IMPORTS

## DESCRIPTIONS

**MODULE : example\_\_7**

[Up](#)

Basic Type Example Source Code copied from ECL Documentation

1. [R](#)

**RECORD : R**

[Up](#)

# example\_\_2

## IMPORTS

## DESCRIPTIONS

**MODULE : example\_\_2**

[Up](#)

Basic Inheritance documentation : mod\_3 inherits both mod\_1 and mod\_2 . Inherits v2\_m1, v2\_m2, Overrides v1\_m1, new locals v2\_m3 . Interface Inheritance : mod\_4 inherits interface iface\_1, overrides v1\_i1

1. [rec\\_1](#)
2. [rec\\_2](#)
3. [rec\\_3](#)
4. [mod\\_1](#)
5. [mod\\_2](#)
6. [mod\\_3](#)
7. [iface\\_1](#)
8. [mod\\_4](#)

**RECORD : rec\_\_1**

[Up](#)

**RECORD : rec\_\_2**

[Up](#)

**RECORD : rec\_\_3**

[Up](#)

**MODULE : mod\_\_1**

[Up](#)

1. [v1\\_\\_m1](#)

2. [v2\\_\\_m1](#)

**ATTRIBUTE : v1\_\_m1**

real8

[Up](#)

**ATTRIBUTE : v2\_\_m1**

[Up](#)

**MODULE : mod\_\_2**

[Up](#)

1. [v1\\_\\_m1](#)

2. [v2\\_\\_m2](#)

**ATTRIBUTE : v1\_\_m1**

[Up](#)

**ATTRIBUTE : v2\_\_m2**

[Up](#)



**MODULE : mod\_3**

[Up](#)

1. [v2\\_m1](#)
2. [v2\\_m2](#)
3. [v1\\_m1](#)
4. [v2\\_m3](#)

**ATTRIBUTE : v2\_m1**

[Up](#)

**INHERITED : True**

**ATTRIBUTE : v2\_m2**

[Up](#)

**INHERITED : True**

**ATTRIBUTE : v1\_m1**

[Up](#)

**OVERRIDE : True**

**ATTRIBUTE : v2\_m3**

[Up](#)

**INTERFACE : iface\_\_1**

[Up](#)

1. [v1\\_i1](#)

**ATTRIBUTE : v1\_i1**

real8

[Up](#)

**MODULE : mod\_\_4**

[Up](#)

1. [v1\\_i1](#)

2. [v2\\_m4](#)

**ATTRIBUTE : v1\_i1**

[Up](#)

**OVERRIDE : True**

**ATTRIBUTE : v2\_m4**

STRING20

[Up](#)

# example

## IMPORTS

## DESCRIPTIONS

**MODULE : example**

[Up](#)

Basic Example with : records, interface, function, modules, transform, embed, macros and functionmacro

1. [rec\\_1](#)
2. [rec\\_2](#)
3. [interface\\_ex](#)
4. [func\\_1](#)
5. [func\\_2](#)
6. [mod\\_1](#)
7. [mod\\_2](#)
8. [cpp\\_1](#)
9. [funcmacro\\_1](#)
10. [macro\\_1](#)
11. [macro\\_2](#)

**RECORD : rec\_\_1**

[Up](#)

**RECORD : rec\_\_2**

[Up](#)

**INTERFACE : interface\_\_ex**

[Up](#)

1. [iface\\_\\_v3](#)

**ATTRIBUTE : iface\_\_v3**

STRING25

[Up](#)

**FUNCTION : func\_\_1**

(REAL8 x, STRING25 y)

[Up](#)

**FUNCTION : func\_\_2**

DATASET(rec\_2) (DATASET(rec\_1) d)

[Up](#)

**MODULE : mod\_\_1**

(REAL8 a)

[Up](#)

1. [pi\\_\\_w](#)

**ATTRIBUTE : pi\_\_w**

[Up](#)

**MODULE : mod\_\_2**

[Up](#)

1. [pi\\_\\_wo](#)

**ATTRIBUTE : pi\_\_wo**

[Up](#)

**EMBED : cpp\_1**

DATA (REAL8 varcpp)

[Up](#)

**MACRO : funcmacro\_1**

(num)

[Up](#)

**MACRO : macro\_1**

(num\_1, num\_2)

[Up](#)

**MACRO : macro\_2**

[Up](#)

# Inintest

## Table of Contents

file : [Example\\_3.ecl](#)

# Inintest.Example\_\_3

## IMPORTS

## DESCRIPTIONS

**MODULE : Example\_\_3**

[Up](#)

1. [mod\\_\\_1](#)

**MODULE : mod\_\_1**

[Up](#)

1. [v2\\_\\_m1\\_\\_ex3](#)

**ATTRIBUTE : v2\_\_m1\_\_ex3**

[Up](#)

# intest

## Table of Contents

file : [example\\_4.ecl](#) Example : Inheritance across files  
file : [example\\_5.ecl](#)  
file : [example\\_11.ecl](#)  
file : [example\\_9.ecl](#)  
file : [example\\_7.ecl](#) Basic Type Example  
file : [example\\_2.ecl](#) Basic Inheritance documentation : mod\_3 inherits both mod\_1 and mod\_2  
file : [example\\_3.ecl](#) Example : Inheritance across files  
dir : [inlintest](#)  
dir : [inintest](#)



# intest.example\_\_4

## IMPORTS

- [Example\\_3.mod\\_1](#)

## DESCRIPTIONS

**MODULE : example\_\_4**

[Up](#)

Example : Inheritance across files [mod\\_1](#) in [Example\\_4](#) inherits [mod\\_1](#) in [Example\\_3](#)

1. [mod\\_1](#)

**MODULE : mod\_\_1**

[Up](#)

1. [v2\\_\\_m1\\_\\_ex4](#)
2. [v1\\_\\_m1](#)
3. [v2\\_\\_m1\\_\\_ex3](#)
4. [long\\_\\_name](#)

**ATTRIBUTE : v2\_\_m1\_\_ex4**

[Up](#)

**ATTRIBUTE : v1\_\_m1**

[Up](#)

Doc test 2. Title end by period not newline

ABCD ||||  
CDEF ||||

**INHERITED** : True

**ATTRIBUTE** : v2\_\_m1\_\_ex3

[Up](#)

DOC Test 3 No Period title

**INHERITED** : True

**FUNCTION** : long\_\_name

(DATASET({REAL8 u}) X, DATASET({REAL8 u}) IntW, DATASET({REAL8 u}) Intb, REAL8  
BETA=0.1, REAL8 sparsityParam=0.1 , REAL8 LAMBDA=0.001, REAL8 ALPHA=0.1, UNSIGNED2  
MaxIter=100)

[Up](#)

**INHERITED** : True

intest.example\_\_5

IMPORTS

DESCRIPTIONS

# intest.example\_\_11

## IMPORTS

- std
- intest
- Example\_3
- intest.Example\_3
- intest.inintest
- intest.inintest.Example\_3
- test
- Inintest
- Inintest.Example\_3

## DESCRIPTIONS

**MODULE : example\_\_11**

[Up](#)

# intest.example\_\_9

## IMPORTS

- example\_8
- example\_8.mod\_1

## DESCRIPTIONS

# intest.example\_\_7

## IMPORTS

## DESCRIPTIONS

**MODULE : example\_\_7**

[Up](#)

Basic Type Example Source Code copied from ECL Documentation

1. [R](#)

**RECORD : R**

[Up](#)

# intest.example\_\_2

## IMPORTS

## DESCRIPTIONS

**MODULE : example\_\_2**

[Up](#)

Basic Inheritance documentation : mod\_3 inherits both mod\_1 and mod\_2 . Inherits v2\_m1, v2\_m2, Overrides v1\_m1, new locals v2\_m3 . Interface Inheritance : mod\_4 inherits interface iface\_1, overrides v1\_i1

1. [rec\\_\\_1](#)
2. [rec\\_\\_2](#)
3. [rec\\_\\_3](#)
4. [mod\\_\\_1](#)
5. [mod\\_\\_2](#)
6. [mod\\_\\_3](#)
7. [iface\\_\\_1](#)
8. [mod\\_\\_4](#)

**RECORD : rec\_\_1**

[Up](#)

**RECORD : rec\_\_2**

[Up](#)

**RECORD : rec\_3**

[Up](#)

**MODULE : mod\_1**

[Up](#)

1. [v1\\_m1](#)

2. [v2\\_m1](#)

**ATTRIBUTE : v1\_m1**

real8

[Up](#)

**ATTRIBUTE : v2\_m1**

[Up](#)

**MODULE : mod\_2**

[Up](#)

1. [v1\\_m1](#)

2. [v2\\_m2](#)

**ATTRIBUTE : v1\_m1**

[Up](#)

**ATTRIBUTE : v2\_m2**

[Up](#)



**MODULE : mod\_3**

[Up](#)

1. [v2\\_m1](#)
2. [v2\\_m2](#)
3. [v1\\_m1](#)
4. [v2\\_m3](#)

**ATTRIBUTE : v2\_m1**

[Up](#)

**INHERITED : True**

**ATTRIBUTE : v2\_m2**

[Up](#)

**INHERITED : True**

**ATTRIBUTE : v1\_m1**

[Up](#)

**OVERRIDE : True**

**ATTRIBUTE : v2\_m3**

[Up](#)

**INTERFACE : iface\_\_1**

[Up](#)

1. [v1\\_i1](#)

**ATTRIBUTE : v1\_i1**

real8

[Up](#)

**MODULE : mod\_\_4**

[Up](#)

1. [v1\\_i1](#)

2. [v2\\_m4](#)

**ATTRIBUTE : v1\_i1**

[Up](#)

**OVERRIDE : True**

**ATTRIBUTE : v2\_m4**

STRING20

[Up](#)

# intest.example\_\_3

## IMPORTS

## DESCRIPTIONS

**MODULE : Example\_\_3**

[Up](#)

Example : Inheritance across files mod\_\_1 in Example\_\_4 inherits mod\_\_1 in Example\_\_3

1. [mod\\_\\_1](#)

**MODULE : mod\_\_1**

[Up](#)

1. [v1\\_\\_m1](#)
2. [v2\\_\\_m1\\_\\_ex3](#)

**ATTRIBUTE : v1\_\_m1**

[Up](#)

**ATTRIBUTE : v2\_\_m1\_\_ex3**

[Up](#)

# inlintest

## Table of Contents

file : [example\\_4.ecl](#) Example : Inheritance across files

file : [example\\_5.ecl](#)

file : [example\\_9.ecl](#)

file : [example\\_7.ecl](#) Basic Type Example

file : [example\\_2.ecl](#) Basic Inheritance documentation : mod\_3 inherits both mod\_1 and mod\_2

file : [example\\_3.ecl](#) Example : Inheritance across files

# intest.in1intest.example\_\_4

## IMPORTS

- Example\_3.mod\_1

## DESCRIPTIONS

**MODULE : example\_\_4**

[Up](#)

Example : Inheritance across files mod\_1 in Example\_4 inherits mod\_1 in Example\_3

1. [mod\\_1](#)

**MODULE : mod\_\_1**

[Up](#)

1. [v2\\_\\_m1\\_\\_ex4](#)
2. [v1\\_\\_m1](#)
3. [v2\\_\\_m1\\_\\_ex3](#)
4. [long\\_\\_name](#)

**ATTRIBUTE : v2\_\_m1\_\_ex4**

[Up](#)

**ATTRIBUTE : v1\_\_m1**

[Up](#)

Doc test 2. Title end by period not newline

ABCD ||||  
CDEF ||||

**INHERITED** : True

**ATTRIBUTE** : v2\_\_m1\_\_ex3

[Up](#)

DOC Test 3 No Period title

**INHERITED** : True

**FUNCTION** : long\_\_name

(DATASET({REAL8 u}) X, DATASET({REAL8 u}) IntW, DATASET({REAL8 u}) Intb, REAL8  
BETA=0.1, REAL8 sparsityParam=0.1 , REAL8 LAMBDA=0.001, REAL8 ALPHA=0.1, UNSIGNED2  
MaxIter=100)

[Up](#)

**INHERITED** : True

intest.in1intest.example\_\_5

IMPORTS

DESCRIPTIONS

# intest.in1intest.example\_\_9

## IMPORTS

- example\_8
- example\_8.mod\_1

## DESCRIPTIONS



# intest.in1intest.example\_\_7

## IMPORTS

## DESCRIPTIONS

**MODULE : example\_\_7**

[Up](#)

Basic Type Example Source Code copied from ECL Documentation

1. [R](#)

**RECORD : R**

[Up](#)

# intest.in1intest.example\_\_2

## IMPORTS

## DESCRIPTIONS

**MODULE : example\_\_2**

[Up](#)

Basic Inheritance documentation : mod\_3 inherits both mod\_1 and mod\_2 . Inherits v2\_m1, v2\_m2, Overrides v1\_m1, new locals v2\_m3 . Interface Inheritance : mod\_4 inherits interface iface\_1, overrides v1\_i1

1. [rec\\_\\_1](#)
2. [rec\\_\\_2](#)
3. [rec\\_\\_3](#)
4. [mod\\_\\_1](#)
5. [mod\\_\\_2](#)
6. [mod\\_\\_3](#)
7. [iface\\_\\_1](#)
8. [mod\\_\\_4](#)

**RECORD : rec\_\_1**

[Up](#)

**RECORD : rec\_\_2**

[Up](#)

**RECORD : rec\_3**

[Up](#)

**MODULE : mod\_1**

[Up](#)

1. [v1\\_m1](#)

2. [v2\\_m1](#)

**ATTRIBUTE : v1\_m1**

real8

[Up](#)

**ATTRIBUTE : v2\_m1**

[Up](#)

**MODULE : mod\_2**

[Up](#)

1. [v1\\_m1](#)

2. [v2\\_m2](#)

**ATTRIBUTE : v1\_m1**

[Up](#)

**ATTRIBUTE : v2\_m2**

[Up](#)

**MODULE : mod\_3**

[Up](#)

1. [v2\\_m1](#)
2. [v2\\_m2](#)
3. [v1\\_m1](#)
4. [v2\\_m3](#)

**ATTRIBUTE : v2\_m1**

[Up](#)

**INHERITED : True**

**ATTRIBUTE : v2\_m2**

[Up](#)

**INHERITED : True**

**ATTRIBUTE : v1\_m1**

[Up](#)

**OVERRIDE : True**

**ATTRIBUTE : v2\_m3**

[Up](#)

**INTERFACE : iface\_\_1**

[Up](#)

1. [v1\\_i1](#)

**ATTRIBUTE : v1\_i1**

real8

[Up](#)

**MODULE : mod\_\_4**

[Up](#)

1. [v1\\_i1](#)

2. [v2\\_m4](#)

**ATTRIBUTE : v1\_i1**

[Up](#)

**OVERRIDE : True**

**ATTRIBUTE : v2\_m4**

STRING20

[Up](#)

# intest.in1intest.example\_\_3

## IMPORTS

## DESCRIPTIONS

**MODULE : Example\_\_3**

[Up](#)

Example : Inheritance across files mod\_\_1 in Example\_\_4 inherits mod\_\_1 in Example\_\_3

1. [mod\\_\\_1](#)

**MODULE : mod\_\_1**

[Up](#)

1. [v1\\_\\_m1](#)
2. [v2\\_\\_m1\\_\\_ex3](#)

**ATTRIBUTE : v1\_\_m1**

[Up](#)

**ATTRIBUTE : v2\_\_m1\_\_ex3**

[Up](#)

# inintest

## Table of Contents

file : [example\\_4.ecl](#) Example : Inheritance across files

file : [example\\_5.ecl](#)

file : [example\\_9.ecl](#)

file : [example\\_7.ecl](#) Basic Type Example

file : [example\\_2.ecl](#) Basic Inheritance documentation : mod\_3 inherits both mod\_1 and mod\_2

file : [example\\_3.ecl](#) Example : Inheritance across files

# intest.inintest.example\_\_4

## IMPORTS

- `Example_3.mod_1`

## DESCRIPTIONS

**MODULE : example\_\_4**

[Up](#)

Example : Inheritance across files `mod_1` in `Example_4` inherits `mod_1` in `Example_3`

1. [mod\\_1](#)

**MODULE : mod\_\_1**

[Up](#)

1. [v2\\_\\_m1\\_\\_ex4](#)
2. [v1\\_\\_m1](#)
3. [v2\\_\\_m1\\_\\_ex3](#)
4. [long\\_\\_name](#)

**ATTRIBUTE : v2\_\_m1\_\_ex4**

[Up](#)

**ATTRIBUTE : v1\_\_m1**

[Up](#)



Doc test 2. Title end by period not newline

ABCD ||||  
CDEF ||||

**INHERITED** : True

**ATTRIBUTE** : v2\_\_m1\_\_ex3

[Up](#)

DOC Test 3 No Period title

**INHERITED** : True

**FUNCTION** : long\_\_name

(DATASET({REAL8 u}) X, DATASET({REAL8 u}) IntW, DATASET({REAL8 u}) Intb, REAL8  
BETA=0.1, REAL8 sparsityParam=0.1 , REAL8 LAMBDA=0.001, REAL8 ALPHA=0.1, UNSIGNED2  
MaxIter=100)

[Up](#)

**INHERITED** : True

intest.inintest.example\_\_5

IMPORTS

DESCRIPTIONS

# intest.inintest.example\_\_9

## IMPORTS

- example\_8
- example\_8.mod\_1

## DESCRIPTIONS

# intest.inintest.example\_\_7

## IMPORTS

## DESCRIPTIONS

**MODULE : example\_\_7**

[Up](#)

Basic Type Example Source Code copied from ECL Documentation

1. [R](#)

**RECORD : R**

[Up](#)

# intest.inintest.example\_\_2

## IMPORTS

## DESCRIPTIONS

**MODULE : example\_\_2**

[Up](#)

Basic Inheritance documentation : mod\_3 inherits both mod\_1 and mod\_2 . Inherits v2\_m1, v2\_m2, Overrides v1\_m1, new locals v2\_m3 . Interface Inheritance : mod\_4 inherits interface iface\_1, overrides v1\_i1

1. [rec\\_\\_1](#)
2. [rec\\_\\_2](#)
3. [rec\\_\\_3](#)
4. [mod\\_\\_1](#)
5. [mod\\_\\_2](#)
6. [mod\\_\\_3](#)
7. [iface\\_\\_1](#)
8. [mod\\_\\_4](#)

**RECORD : rec\_\_1**

[Up](#)

**RECORD : rec\_\_2**

[Up](#)

**RECORD : rec\_\_3**

[Up](#)

**MODULE : mod\_\_1**

[Up](#)

1. [v1\\_\\_m1](#)

2. [v2\\_\\_m1](#)

**ATTRIBUTE : v1\_\_m1**

real8

[Up](#)

**ATTRIBUTE : v2\_\_m1**

[Up](#)

**MODULE : mod\_\_2**

[Up](#)

1. [v1\\_\\_m1](#)

2. [v2\\_\\_m2](#)

**ATTRIBUTE : v1\_\_m1**

[Up](#)

**ATTRIBUTE : v2\_\_m2**

[Up](#)

**MODULE : mod\_3**

[Up](#)

1. [v2\\_m1](#)
2. [v2\\_m2](#)
3. [v1\\_m1](#)
4. [v2\\_m3](#)

**ATTRIBUTE : v2\_m1**

[Up](#)

**INHERITED : True**

**ATTRIBUTE : v2\_m2**

[Up](#)

**INHERITED : True**

**ATTRIBUTE : v1\_m1**

[Up](#)

**OVERRIDE : True**

**ATTRIBUTE : v2\_m3**

[Up](#)

**INTERFACE : iface\_\_1**

[Up](#)

1. [v1\\_i1](#)

**ATTRIBUTE : v1\_i1**

real8

[Up](#)

**MODULE : mod\_\_4**

[Up](#)

1. [v1\\_i1](#)

2. [v2\\_m4](#)

**ATTRIBUTE : v1\_i1**

[Up](#)

**OVERRIDE : True**

**ATTRIBUTE : v2\_m4**

STRING20

[Up](#)



# intest.inintest.example\_\_3

## IMPORTS

- std.Str

## DESCRIPTIONS

**MODULE : Example\_\_3**

[Up](#)

Example : Inheritance across files mod\_\_1 in Example\_4 inherits mod\_\_1 in Example\_\_3

1. [mod\\_\\_1](#)

**MODULE : mod\_\_1**

[Up](#)

1. [v1\\_\\_m1](#)
2. [v2\\_\\_m1\\_\\_ex3](#)

**ATTRIBUTE : v1\_\_m1**

[Up](#)

**ATTRIBUTE : v2\_\_m1\_\_ex3**

[Up](#)