

H5E_CLEAR2

[Expand all](#) [Collapse all](#)

- [Jump to ...](#)
- [Summary](#)
- [Description](#)
- [Example](#)
- [Switch language ...](#)
- [C](#)
- [C++](#)
- [FORTRAN](#)
- [JAVA](#)

[Summary](#)
[Description](#)
[Example](#)
[JAVA](#)
[FORTRAN](#)
[C++](#)
[C](#)

H5E_CLEAR2

Clears the specified error stack or the error stack for the current thread

Procedure:

H5E_CLEAR2(estack_id)

Signature:

```
herr_t H5Eclear2(hid_t estack_id)
```

```
SUBROUTINE h5eclear_f(hdferr)  
  IMPLICIT NONE  
  INTEGER, INTENT(OUT) :: hdferr ! Error code  
  
END SUBROUTINE h5eclear_f
```

Parameters:

<i>hid_t</i> estack_id	IN: Error stack identifier
------------------------	----------------------------

Description:

H5E_CLEAR2 clears the error stack specified by `estack_id`, or, if `estack_id` is set to `H5E_DEFAULT`, the error stack for the current thread.

`estack_id` is an error stack identifier, such as that returned by `H5E_GET_CURRENT_STACK`.

The current error stack is also cleared whenever an API function is called, with certain exceptions (for instance, H5E_PRINT1 or H5E_PRINT2).

Returns:

Returns a non-negative value if successful; otherwise returns a negative value.

Example:

```
test / error_test.c [484:519] 1.10/master HDF5V/hdf5
err_num = H5Eget_num(estack_id);
    if (err_num != 0)
        TEST_ERROR

    /* Push an error with a long description */
    if (H5Epush(estack_id, __FILE__, err_func, __LINE__, ERR_CLS, ERR_MAJ_TEST,
ERR_MIN_SUBROUTINE, "%s", err_msg) < 0)
        TEST_ERROR;

    /* Check the number of errors on stack */
    err_num = H5Eget_num(estack_id);
    if (err_num != 1)
        TEST_ERROR

    /* Clear the error stack */
    if (H5Eclear2(estack_id) < 0)
        TEST_ERROR

    /* Check the number of errors on stack */
    err_num = H5Eget_num(estack_id);
    if (err_num != 0)
        TEST_ERROR

    /* Close error stack */
    if(H5Eclose_stack(estack_id) < 0)
        TEST_ERROR

    return 0;

error:
    return -1;
} /* end test_create() */

/*-----
* Function:    test_copy
*
* Purpose:    Test copyinging an error stack
```

Fortran example coming soon!

History:

Release	Change
---------	--------

1.8.0

Function introduced in this release.

--- Last Modified: January 08, 2020 | 02:11 PM