

H5LT_MAKE_DATASET_DOUBLE

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

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

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

H5LT_MAKE_DATASET_DOUBLE

Creates and writes a dataset.

Procedure:

H5LT_MAKE_DATASET_DOUBLE(loc_id, dset_name, rank, dims, buffer)

Signature:

```
herr_t H5LTmake_dataset_double ( hid_t loc_id, const char *dset_name, int rank, const hsize_t *dims, const double*buffer )
```

```
subroutine h5ltmake_dataset_double_f(loc_id, dset_name, rank, dims, &
                                   buf, errcode)
    implicit none
    integer(HID_T), intent(IN) :: loc_id           ! file or group identifier
    character(LEN=*), intent(IN) :: dset_name      ! name of the dataset
    integer, intent(IN) :: rank                   ! rank
    integer(HSIZE_T), dimension(*), intent(IN) :: dims
                                                ! size of the buffer buf
    double precision, intent(IN), dimension(*) :: buf
                                                ! data buffer
    integer :: errcode                            ! error code
end subroutine h5ltmake_dataset_double_f
```

Parameters:

<i>hid_t</i> loc_id	IN: Identifier of the file or group to create the dataset within.
<i>const char</i> *dset_name	IN: The name of the dataset to create.
<i>int</i> rank	IN: Number of dimensions of dataspace.
<i>const hsize_t</i> *dims	IN: An array of the size of each dimension.
<i>const double</i> *buffer	IN: Buffer with data to be written to the dataset.

Description:

H5LTmake_dataset creates and writes a dataset named dset_name attached to the object specified by the identifier loc_id.

The dataset's datatype will be *native floating-point double*, H5T_NATIVE_DOUBLE.

Returns:

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

Example:

Coming Soon!

History:

Release	Change
1.8.7	Fortran subroutine modified in this release to accomodate arrays with more than three dimensions.

--- Last Modified: December 04, 2017 | 07:12 AM