

# H5G\_CREATE2

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

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

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

# H5G\_CREATE2

Creates a new group and links it into the file

## Procedure:

H5G\_CREATE2(loc\_id, name, lcpl\_id, gcpl\_id, gapl\_id)

## Signature:

```
hid_t H5Gcreate2( hid_t loc_id, const char *name, hid_t lcpl_id, hid_t gcpl_id, hid_t gapl_id )
```

```

SUBROUTINE h5gcreate_f(loc_id, name, grp_id, hdferr, &
                    size_hint, lcpl_id, gcpl_id, gapl_id)
    IMPLICIT NONE
    INTEGER(HID_T), INTENT(IN) :: loc_id    ! File or group identifier
    CHARACTER(LEN=*), INTENT(IN) :: name    ! Name of the group
    INTEGER(HID_T), INTENT(OUT) :: grp_id    ! Group identifier
    INTEGER, INTENT(OUT) :: hdferr          ! Error code
                                           ! 0 on success and -1 on failure
    INTEGER(SIZE_T), OPTIONAL, INTENT(IN) :: size_hint
                                           ! Parameter indicating the number of
                                           ! bytes to reserve for the names that
                                           ! will appear in the group.
                                           ! Note, set to OBJECT_NAMELEN_DEFAULT_F
                                           ! if using any of the optional
                                           ! parameters lcpl_id, gcpl_id,
                                           ! and/or gapl_id when not
                                           ! using keywords in specifying the
                                           ! optional parameters.
    INTEGER(HID_T), OPTIONAL, INTENT(IN) :: lcpl_id
                                           ! Property list for link creation
    INTEGER(HID_T), OPTIONAL, INTENT(IN) :: gcpl_id
                                           ! Property list for group creation
    INTEGER(HID_T), OPTIONAL, INTENT(IN) :: gapl_id
                                           ! Property list for group access
END SUBROUTINE h5gcreate_f

```

#### Parameters:

<i>hid_t</i> loc_id	IN: Location identifier; may be a file, group, dataset, named datatype or attribute
<i>const char</i> *name	IN: Absolute or relative name of the link to the new group
<i>hid_t</i> lcpl_id	IN: Link creation property list identifier
<i>hid_t</i> gcpl_id	IN: Group creation property list identifier
<i>hid_t</i> gapl_id	IN: Group access property list identifier (No group access properties have been implemented at this time; use H5P_DEFAULT.)

#### Description:

H5G\_CREATE2 creates a new group in a file. After a group has been created, links to datasets and to other groups can be added.

The `loc_id` and `name` parameters specify where the group is located. `loc_id` may be a file, group, dataset, named datatype or attribute in the file. If an attribute, dataset, or named datatype is specified for `loc_id` then the group will be created at the location where the attribute, dataset, or named datatype is attached. `name` is the link to the group; `name` may be either an absolute path in the file (the links from the root group to the new group) or a relative path from `loc_id` (the link(s) from the group specified by `loc_id` to the new group). See the [Accessing objects by location and name](#) topic for more information.

`lcpl_id`, `gcpl_id`, and `gapl_id` are property list identifiers. These property lists govern how the link to the group is created, how the group is created, and how the group can be accessed in the future, respectively. `H5P_DEFAULT` can be passed in if the default properties are appropriate for these property lists. Currently, there are no APIs for the group access property list; use `H5P_DEFAULT`. See [H5P: Property List Interface](#) for the functions that can be used with each property list.

To conserve and release resources, the group should be closed when access is no longer required.

#### Returns:

Returns a group identifier if successful; otherwise returns a negative value.

#### Example:

examples / h5\_crtgrp.c [30:31]

1.10/master HDFFV/hdf5

```
/* Create a group named "/MyGroup" in the file. */  
group_id = H5Gcreate2(file_id, "/MyGroup", H5P_DEFAULT, H5P_DEFAULT, H5P_DEFAULT);
```

fortran / examples / h5\_crtgrp.f90 [43:46]

1.10/master HDFFV/hdf

5

```
!  
! Create a group named "/MyGroup" in the file.  
!  
CALL h5gcreate_f(file_id, groupname, group_id, error)
```

#### History:

Release	Change
1.8.0	Function introduced in this release.

--- Last Modified: August 13, 2019 | 08:46 AM