

# H5P\_SET\_ISTORE\_K

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

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

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

# H5P\_SET\_ISTORE\_K

Sets the size of the parameter used to control the B-trees for indexing chunked datasets

## Procedure:

H5P\_SET\_ISTORE\_K ( fcpl\_id, ik )

## Signature:

```
herr_t H5Pset_istore_k(  
    hid_t fcpl_id,  
    unsigned ik  
)
```

Fortran90 Interface: h5pset\_istore\_k\_f

```
SUBROUTINE h5pset_istore_k_f (prp_id, ik, hdferr)  
    IMPLICIT NONE  
    INTEGER(HID_T), INTENT(IN) :: prp_id ! Property list identifier  
    INTEGER, INTENT(IN) :: ik          ! 1/2 rank of chunked storage B-tree  
    INTEGER, INTENT(OUT) :: hdferr     ! Error code  
                                        ! 0 on success and -1 on failure  
END SUBROUTINE h5pset_istore_k_f
```

## Parameters:

---

<i>hid_fcpl_id</i>	IN: File creation property list identifier
<i>unsigned ik</i>	IN: 1/2 rank of chunked storage B-tree

**Description:**

H5P\_SET\_ISTORE\_K sets the size of the parameter used to control the B-trees for indexing chunked datasets. This function is valid only for file creation property lists.

*ik* is one half the rank of a tree that stores chunked raw data. On average, such a tree will be 75% full, or have an average rank of 1.5 times the value of *ik*.

The HDF5 library uses (*ik*\*2) as the maximum # of entries before splitting a B-tree node. Since only 2 bytes are used in storing # of entries for a B-tree node in an HDF5 file, (*ik*\*2) cannot exceed 65536. The default value for *ik* is 32.

**Returns:**

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

**Example:**

Coming Soon!

**History:**

Release	Change
1.6.4	<i>ik</i> parameter type changed to <i>unsigned</i> .