

# H5T\_DECODE

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

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

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

# H5T\_DECODE

Decodes a binary object description of datatype and return a new object handle

## Procedure:

H5T\_DECODE(buf)

## Signature:

```
hid_t H5Tdecode (unsigned char *buf)
```

```
SUBROUTINE h5tdecode_f(buf, obj_id, hdferr)
  IMPLICIT NONE
  CHARACTER(LEN=*) , INTENT(IN) :: buf ! Data space object buffer to be decoded
  INTEGER(HID_T) , INTENT(OUT) :: obj_id! Object identifier
  INTEGER , INTENT(OUT) :: hdferr ! Error code
  ! 0 on success and -1 on failure
END SUBROUTINE h5tdecode_f
```

## Parameters:

<i>unsigned char</i> *buf	IN: Buffer for the datatype object to be decoded
---------------------------	--

## Description:

Given an object description of datatype in binary in a buffer, H5T\_DECODE reconstructs the HDF5 datatype object and returns a new object handle for it. The binary description of the object is encoded by H5T\_ENCODE . User is responsible for passing in the right buffer.

The datatype identifier returned by this function can be released with H5T\_CLOSE when the identifier is no longer needed so that resource leaks will not develop.

**Returns:**

Returns an object identifier (non-negative) if successful; otherwise returns a negative value.

**Example:**

Coming Soon!

--- Last Modified: May 10, 2019 | 02:34 PM