

# H5TB\_READ\_RECORDS

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

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

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

# H5TB\_READ\_RECORDS

Reads records

## Procedure:

H5TB\_READ\_RECORDS (loc\_id, table\_name, start, nrecords, type\_size, field\_offset, dst\_sizes, data )

## Signature:

```
herr_t H5TBread_records ( hid_t loc_id, const char *table_name, hsize_t start, hsize_t nrecords,  
                        size_t type_size, const size_t *field_offset, const size_t *dst_sizes, void *data )
```

## Parameters:

<i>hid_t</i> loc_id	IN: Identifier of the file or group to read the table within
<i>const char *</i> table_name	IN: The name of the dataset to read
<i>hsize_t</i> start	IN: The start record to read from
<i>hsize_t</i> nrecords	IN: The number of records to read
<i>size_t</i> type_size	IN: The size of the structure type, as calculated by sizeof()
<i>const size_t *</i> field_offset	IN: An array containing the offsets of the fields. These offsets can be calculated with the HOFFSET macro.
<i>const size_t *</i> dst_sizes	IN: An array containing the size in bytes of the fields

`void *data`

OUT: Buffer with data

**Description:**

H5TB\_READ\_RECORDS reads some records identified from a dataset named `table_name` attached to the object specified by the identifier `loc_id`.

**Returns:**

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

**Example:**

hl / test / test\_table.c [559:565]

hdf5\_1\_12

HDFV/hdf5

```
rstart=0;
rrecords=8;
if (H5TBread_records(fid,tname,rstart,rrecords,type_size_mem,field_offset,
    field_size,rbuf)<0)
    goto out;
```

--- Last Modified: December 19, 2019 | 10:03 AM