

Modifying a Record

// 1st get records / MyType temp;

recordH = Dm GetRecord (myDBref, *index);

record = MemHandleLock(recordH);

// Copy temp = *record;

// Modify temp.field = val;

Dm Write(record, 0, &temp, sizeof(temp));

// check if need to
release

Be aware that if (*index > 0) ~~temp = MemHandleLock(Dm Query Record~~

do not
forget
use

Dm Move Record

MemHandleUnlock(recordH)

Dm Release Record (myDBref, index, true)

myDBref, *index, i)
Dm Find Sort Position

Tables

Recall resource for table looks like:

~~FOR MYID MyForm @ 0 160 100~~

~~USABLE
BEGIN
PART~~

TABLE ID MyTable AT

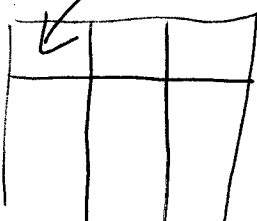
(0 16, 160, 121) ROWS 11

COLUMNS

COLUMN WIDTHS 12, 25 12

~~10 30 20 20~~

each table item



can be
viewed as a container for some
other type

To find where a record belongs use

UInt16 DmFindSortPosition (DmOpenRef dBP, void *
newRec, SortRecordInfoPtr, newRecordInfo,
DmCompareF *compare, Int16 other)

can
pass
NULL

To create a new record use

MemHandle DmNewRecord (DmOpenRef dBP, UInt16 *atP,
UInt32 size)

// create record newRecord then
~~Ex~~ Ex) index = DmFindSortPosition(
gDB, &newRecord,
NULL, (DmCompareF *)
MyCompareFunc, NULL)
newRecord# = ~~newRecord~~ DmNewRecord
(gDB, &index, sizeof(Record))
newRecordP = MemHandleLock
offset (newRecord#)
DmWrite(newRecordP, 0,
&newRecord, sizeof(newRecord))
MemHandleUnlock(newRecord#)
DmReleaseRecord(gDB, index, true)

To read use MemHandle DmQueryRecord (gDB, index) dirty bit set
To delete use DmDeleteRecord (gDB, index)