HEAP FILE

TupleDesc
File
#Pages
➜ **Tuple Iterator (given)**
➜ **ReadPage**
This is just the organizing class that provides access methods read and write to pages of a file from the disk, but it's the Buffer Pool that Manage the retrieve of single pages.

ReadPage: used by the BP to rietrieve a page not in memory

HEAP PAGE

TupleDesc
Header[int]
Tuple[]
#Slots
➜ **Tuple Iterator**
Provide access methods read/write to a page of a file. Fixed num of byte sto store tuples in a slot, and additional bytes fora n header that has 1 bit for each slot to map if empty or not.

RECORD ID

Type
FieldName

TUPLE

TupleDesc
Field[]
RecordId
➜ **Field Iterator**
Object created by the access method HeapPage

TUPLE DESC

#Field
TDItem[]
➜ **TDItem Iterator**
Schema of the tuple also the header of the table.

TD ITEM

Type
FieldName

Iterator: gives the control to the buffer pool to get the page, and initialize his iterator.

GetDBFile: used by BP and operators to retrieve a Table File object (with all the corresponding access methods)

BUFFER POOL

HashMap <PageId, HeapPage>
➜ **GetPage**
Collection of pages that are currently resident in memory (that is the cache of a DBMS).

GetPage:Only for pages I don't have, I want this File to use the method to read this specific pageId

CATALOG

HashMap <TableId, Name>
HashMap <TableId, Pkey>
HashMap <TableId, DbFile>
➜ **GetDBFile**
List of all tables present in the DB (disk and memory). He is like the guardian, and everybody ask to him to have specific objects.

SEQSCAN

TransactionID
TableId
TableAlias
➜ **Tuple Iterator (given)**
Operator responsible for the actual exec of querce. He ask fora n iterator through the tuples of a table he wants.

SeqScan: ask for an iterator the tuples of this tableId I give you

- Methods in blue
- Iterator methods in green