Indexing in DBMS

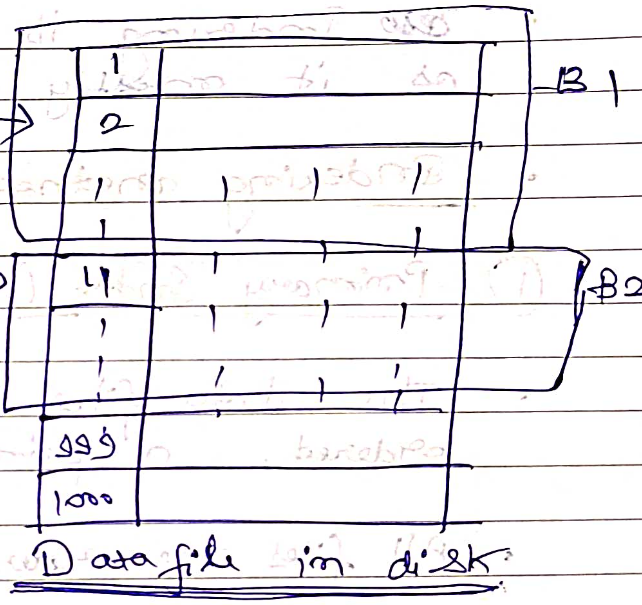
- It is an efficient way to optimise the performance of a database by minimissing the number of disk accesses required.

- We form an index, which is a type of data structure.

- Index is always sorted.

Index file

Roll no	BP
1	B1
11	B2
21	B3
⋮	⋮
991	B100



Data file in disk

Each block (B1) holds 10 records.

- Search Key :- Here "Roll no.", is a Search key, it usually contains primary key or candidate key.

- Data reference :- Pointer holding the address of disk block. Here "BP" is the pointer.

- Here we made an "Index file" which stores the data file info into shorter form. It holds the rollno and its associated Block pointer where it actually stores in the memory block.

- It reduces our time to search for an particular record. As instead of applying search operation ~~to~~ like Binary Search to the whole data file, we can apply it to our sorted and less data file index and then after we go to that particular Block and finds the record.

- ~~[illegible]~~ Indexing is an optional ~~[illegible]~~ method as it mostly helpful in large data base.

- Indexing methods

① Primary Index (clustering Index)

- The data files of the records is sequentially ordered, a primary index is used.

- All files are ordered sequentially on some search key

- Dense indias
- It contains an index record for every search key value.

Index

Rollno.	BP
1	1
2	1
3	1
⋮	1
1000	⋮

- Sparse index

- An index record appears for only some of the search-key values.

Index

Roll no.	BP
1	1
11	2
21	3
:	:

- Dense indices takes more space as compared to sparse index.

- Based on key attribute :-
- Data file is sorted with respect to primary key.
- PK will be used as search-key in index.
- Sparse index will be formed,
  no. of entries in index file = no. of blocks in a data file.

- Based on Non-key attribute :-
- Data file is sorted w.r.t to Non-primary key.
- No. of entries in the index = unique non-key attributes values in the data file

- dense indices is formed here.
  e.g.⇒ A company recruited many employee in various departments. In this case, clustering indexing in DBMS is create according to the department.

- Multi-level index
- Index with two or more level. It is a index for index.

② Secondary index (Non-clustering index)

- Datafile is unsorted. Primary indexing is not possible.

- No. of entries in index = no. of records in data file.

- example of an Dense index.

* Advantages of indexing:
- Faster access and retrieval of data.

* Limitation of indexing
- Additional space to store index table.