



ORGANISASI FILE

ADI RACHMANTO, S.KOM., M.KOM

REKAYASA DESAIN SYSTEM

PROGRAM STUDI AKUNTANSI UNIKOM

PENDAHULUAN

Penyimpanan ataupun penulisan karakter demi karakter yang ada didalam eksternal memori, harus diatur sedemikian rupa sehingga komputer bisa dengan mudah menemukan kembali data-data yang tersimpan didalamnya. Aturan inilah kemudian dikenal sebagai organisasi file.

TIPE – TIPE ORGANISASI FILE

Ada beberapa tipe organisasi yang digunakan, yaitu :

1. Organisasi File Berurut (*Sequential*)
2. Organisasi File Urut Berindek (*Indexed Sequential*)
3. Organisasi File Acak (*Random*)

ORGANISASI FILE BERURUT (*SEQUENTIAL*)

- Suatu cara atau metode penyimpanan dan pembacaan data yang dilakukan secara berurutan.
- Metode ini mempunyai ciri-ciri :
 - ✓ Rekaman disimpan berdasarkan suatu kunci
 - ✓ Pencarian rekaman tertentu dilakukan record demi record sesuai kuncinya.

ORGANISASI FILE URUT BERINDEK (INDEXED SEQUENTIAL)

- Teknik penyimpanan yang dilakukan menggunakan suatu indeks yang isinya berupa bagian dari data yang sudah bersortir.
- Ciri-cirinya adalah :
 - ☑ Merupakan gabungan antara metode file berurut dan acak.
 - ☑ Record disimpan secara berurutan menggunakan kunci.
 - ☑ Masing-masing record diberi indeks.
 - ☑ Pengalamatan dilakukan secara acak.
 - ☑ Perlu penyimpanan tambahan, yaitu file indeks.

ORGANISASI FILE ACAK (RANDOM)

Random file merupakan suatu cara ataupun suatu metode penyimpanan dan pembacaan data yang dilakukan secara random atau langsung. Dengan demikian, random file juga disebut sebagai *Direct Access File*. Dalam hal ini, tempat penyimpanan data sudah diatur sedemikian rupa, sehingga setiap data akan tersimpan didalam tempat-tempat yang telah ditentukan sesuai dengan nomor data yang dimilikinya.

VISUALISASI ORGANISASI FILE [1]

Untuk membayangkan penyimpanan dan pembacaan data secara *sequential* (berurut), kita bisa melihat rekaman lagu yang tersimpan pada kaset.

Untuk mendengarkan lagu kelima, maka harus melalui lagu kesatu, dua, tiga dan empat terlebih dahulu. Pembacaan seperti inilah yang disebut sebagai *sequential* atau berurutan.

VISUALISASI ORGANISASI FILE [2]

Apabila lagu-lagu yang ada kemudian disimpan di dalam *compact-disk*, maka untuk mendengarkan lagu yang ke lima bisa langsung dilakukan. Disamping itu, dengan *compact-disk* juga bisa dilakukan pembacaan secara berurutan atau sequential.

Compactdisk menyimpan lagu secara *random*.

Untuk membayangkan penyimpanan data dengan menggunakan teknik index sequential ini, kita bisa melihat daftar isi pada sebuah buku. Pada bagian disebelah kiri disebut sebagai index data yang berisi bagian dari data yang ada. Index data kemudian diakhiri dengan pointer yang menunjukkan posisi keseluruhan isi data.

ORGANISASI FILE URUT BERINDEK (INDEXED SEQUENTIAL)

Hall dan Singleton (2007) mendefinisikan file urut berindeksi (*indexed sequential file*) adalah struktur data yang mendasari model penggunaan kunci melekat untuk menyediakan hubungan antara tabel-tabel berkaitan. Struktur file berurutan berindeks menggunakan satu indeks berkaitan dengan organisasi file berurutan.

Suatu struktur disebut struktur berindeks karena selain file data aktual, terdapat indeks terpisah yang juga merupakan file alamat record. Indeks ini berisi nilai numerik dari lokasi penyimpanan disk fisik (silinder, permukaan, dan blok record) untuk setiap record dalam file data terakhir. File data itu sendiri bisa diatur secara berurutan atau acak.

ORGANISASI FILE

Beberapa konsep dasar

- **Field**
Satuan informasi terkecil yang menyusun record
- **Record**
Kumpulan dari field yang berhubungan satu sama lain
- **File**
Kumpulan dari record-record
- **Basis data**
Kumpulan file yang digunakan oleh program aplikasi serta membentuk hubungan tertentu di antara record-record di file-file tersebut

ORGANISASI FILE

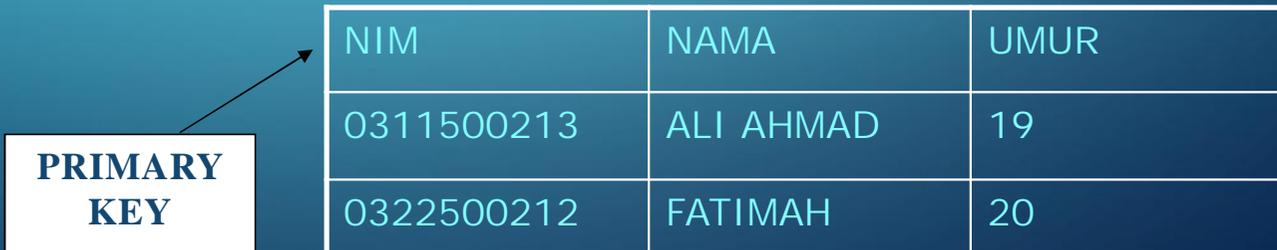
➤ Key

Elemen record yang dipakai untuk menemukan record tersebut pada waktu akses

➤ Jenis-jenis key:

- Primary key
- Field yang mengidentifikasi sebuah record dalam file
- Bersifat unik

**PRIMARY
KEY**

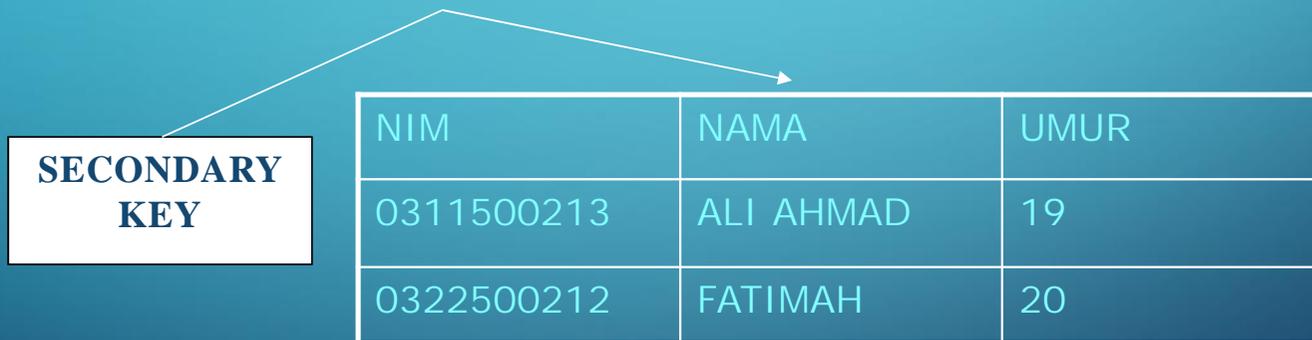


NIM	NAMA	UMUR
0311500213	ALI AHMAD	19
0322500212	FATIMAH	20

ORGANISASI FILE

❖ Secondary key

- ✓ Field yang mengidentifikasi sebuah record dalam file
- ✓ Tidak bersifat unik

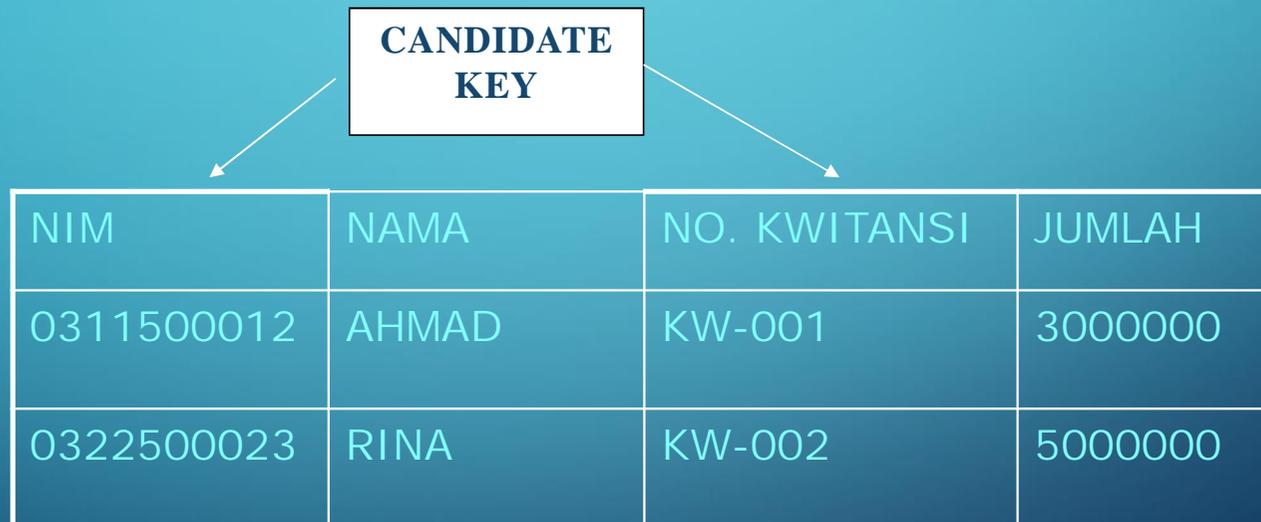


A diagram illustrating a secondary key. A white box with the text "SECONDARY KEY" has an arrow pointing to the "NIM" column of a table. The table has three columns: "NIM", "NAMA", and "UMUR". The first row contains the values "0311500213", "ALI AHMAD", and "19". The second row contains the values "0322500212", "FATIMAH", and "20".

NIM	NAMA	UMUR
0311500213	ALI AHMAD	19
0322500212	FATIMAH	20

ORGANISASI FILE

- Candidate key
Field-field yang bisa dipilih (dipakai) menjadi primary key



ORGANISASI FILE

- Composite key

Primary key yang dibentuk dari beberapa field

HARI	RUANG	MATA KULIAH
SELASA	4.2.2	JARINGAN KOMPUTER
SELASA	4.2.1	SISTEM BASIS DATA 1
RABU	4.2.2	PANCASILA

**COMPOSITE
KEY**



ORGANISASI FILE

- Foreign key

Field yang bukan key, tetapi adalah key pada file yang lain

KD_MK	NM_MK	SKS	KD_DOSEN
K82	SBD-1	2	183
K29	JARKOM	3	200



ORGANISASI FILE

- ✓ **File sekuensial didesign untuk efisiensi pemrosesan record pada saat pengurutan berdasarkan beberapa key**
- ✓ **File dengan data yang tersusun dalam suatu urutan tertentu**
- ✓ **Tiap Record Mempunyai Field Yang Sama & Dengan Susunan Yang Sama**

ORGANISASI FILE

STRUKTUR FILE

- Untuk memungkinkan record tersusun secara urut perlu ditentukan key dari tiap record
- Pembacaan secara serial (satu persatu) sesuai dengan urutan keynya disebut pembacaan secara sequential

Nip	Nama	Pekerjaan
000021	Abu Bakar	Manajer
000032	Fatimah	Sekretaris
000042	Asma	Presiden direktur

ORGANISASI FILE SEKUENTIAL

Insert sebuah record

- ✓ **Insert berarti menambahkan sebuah data baru ke dalam file**
- ✓ **Insert pada ujung akhir sebuah file, hanyalah menambah banyaknya data waktu yang dibutuhkan.**

1	2	3	4	5	6	7	8	9	...
A	B	C	D	E	F

INSERT X PADA AKHIR RECORD

1	2	3	4	5	6	7	8	9	...
A	B	C	D	E	F	X

ORGANISASI FILE SEKUENTIAL

Insert ditengah file mengakibatkan pergeseran ataupun perubahan struktur data yang tidak sederhana

1	2	3	4	5	6	7	8	9	...
A	B	C	D	E	F

INSERT X PADA RECORD KE 3

1	2	3	4	5	6	7	8	9	...
A	B	X	C	D	E	F

→ RECORD KE-3 DST BERGESER

ORGANISASI FILE SEKUENTIAL

DELETE SEBUAH RECORD

- ✓ Menghapus sebuah record
- ✓ Mencari lokasi data & menghapus isinya, agar bisa dipakai oleh data yang lain
- ✓ Setelah itu dilakukan pergeseran ataupun pengaturan struktur data kembali

1	2	3	4	5	6	7	8	9	...
A	B	C	D	E	F

→ HAPUS

BILA RECORD D DIHAPUS, MAKA AKAN TERJADI PEMBACAAN DAN PENULISAN ULANG RECORD E, F, DST

1	2	3	4	5	6	7	8	9	...
A	B	C	E	F

ORGANISASI FILE SEKUENTIAL

Kadangkala delete dilakukan dengan hanya memberi tanda saja (tombstone / flag), tanpa dilakukan penghapusan ataupun pengaturan struktur datanya

1	2	3	4	5	6	7	8	9	...
A	B	C	D	E	F

→ HAPUS

1	2	3	4	5	6	7	8	9	...
A	B	C	*	E	F

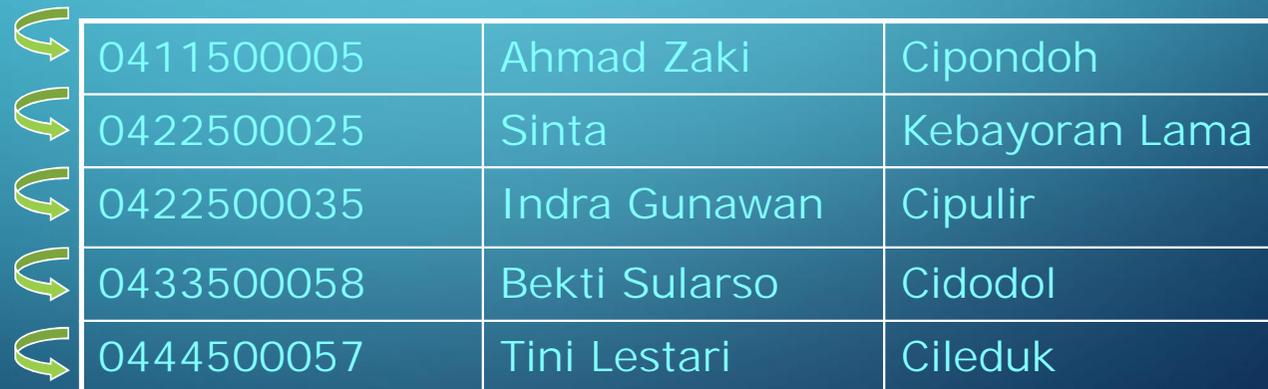
→ record yang sudah dihapus "Delete"

ORGANISASI FILE INDEX

KONSEP DASAR

- Sebuah File Akan Terus Diakses Untuk Mencari Datanya (*Fetch Data*) Untuk Kemudian Data Tersebut Diambil Dari File (*Retrieve Data*)
- Untuk mencari data pada sebuah tabel dapat dilakukan secara sekuensial. Namun cara pencarian ini akan memakan waktu lama jika file terdiri dari banyak record

SEKUEENTIAL
SEARCH
MULAI DARI
RECORD-1
..... DST



0411500005	Ahmad Zaki	Cipondoh
0422500025	Sinta	Kebayoran Lama
0422500035	Indra Gunawan	Cipulir
0433500058	Bekti Sularso	Cidodol
0444500057	Tini Lestari	Cileduk

ORGANISASI FILE INDEX

PADA DASARNYA TERDAPAT 2 MACAM PENGURUTAN :

1. Pengurutan secara indeks

Berdasarkan urutan dari sebuah nilai

2. Pengurutan secara hash

Berdasarkan fungsi hash yang digunakan

TIAP PENGURUTAN MEMPERHATIKAN FAKTOR-FAKTOR, YAITU :

- **TIPE AKSES**

Tipe akses dalam mencari record. Yang lebih dipilih tentunya yang lebih efisien

- **WAKTU AKSES**

Waktu yang dibutuhkan untuk menemukan sebuah record

- **WAKTU HAPUS**

Waktu yang dibutuhkan untuk menghapus sebuah item

- **RUANG SPASI**

Ruang tambahan yang diminta oleh struktur index.

INDEX YANG TERURUT

- ✓ Untuk mengatasi pencarian record dalam sebuah file secara acak, dapat digunakan struktur index.
- ✓ Tiap struktur index dihubungkan sesuai dengan key yang dicari (search key).
- ✓ Sebuah file dapat mempunyai beberapa file indeks, dengan search key yang ber beda-beda.
- ✓ Jika search key yang dipakai adalah primary key pada sebuah file master maka file index yang dibuat disebut primary indeks.
- ✓ Jika Search Key Yang Dipakai Adalah Bukan Primary Key Pada Sebuah File Master Maka File Index Yang Dibuat Disebut Secondary Indeks.

INDEX YANG TERURUT

- File index terdiri dari nomor record serta field yang digunakan sebagai search key
- Sebelum Mencari Data Pada File Master, Data Dicari Terlebih Dahulu Pada File Index, Jika Data Tersebut Ada, Maka File Index Langsung Menunjuk Lokasi Dari Data Tersebut Pada File Master

INDEX YANG TERURUT

NIM	NO. REC
0233500058	1
0322500025	2
0411500005	3
0422500035	4
0444500057	5



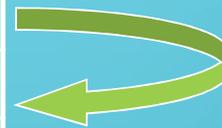
**FILE INDEX YANG
BERJENIS PRIMARY
INDEKS**

**FILE
MASTER
DENGAN NIM
SEBAGAI PK**

NO. REC	NIM	NAMA	ALAMAT
1	0233500058	Ahmad Zaki	Cipondoh
2	0322500025	Sinta	Kebayoran Lama
3	0411500005	Indra Gunawan	Cipulir
4	0422500035	Bekti Sularso	Cidodol
5	0444500057	Tini Lestari	Cileduk

INDEX YANG TERURUT

NAMA	NO. REC
Ahmad Zaki	1
Bekti Sularso	4
Indra Gunawan	3
Sinta	2
Tini Lestari	5



**FILE INDEX YANG
BERJENIS
SECONDARY INDEKS**

**FILE MASTER
DENGAN NIM
SEBAGAI PK**

NO. REC	NIM	NAMA	ALAMAT
1	0411500005	Ahmad Zaki	Cipondoh
2	0322500025	Sinta	Kebayoran Lama
3	0422500035	Indra Gunawan	Cipulir
4	0233500058	Bekti Sularso	Cidodol
5	0444500057	Tini Lestari	Cileduk

RANGKUMAN

1. Akses file (*access file*) adalah suatu metode yang menunjukkan bagaimana suatu program komputer akan membaca record-record dari suatu file.

2. File dapat diakses dengan dua cara yaitu secara urut (*sequential access*) atau secara langsung (*direct access atau random access*). Metode akses urut (*sequential access method*) dilakukan dengan membaca atau menulis suatu record di file dengan membaca terlebih dahulu mulai dari record pertama, urut sampai dengan record yang diinginkan. Metode akses langsung (*direct access method*) dilakukan dengan cara langsung membaca record pada posisinya di file tanpa membaca dari record pertama terlebih dahulu.

RANGKUMAN [2]

3. Organisasi file adalah pengaturan dari record secara logika di dalam file yang dihubungkan satu dengan yang lainnya. File dapat diorganisasikan secara urut (*sequential organization*) atau secara acak (*random organization*).
4. File urut (*sequential file*) merupakan file dengan organisasi urut (*sequential organization*) dengan pengaksesan secara urut (*sequential access*).
5. File urut berindeks (*indexed sequential file*) atau sering disebut dengan ISAM (*indexed sequential access method*) merupakan file dengan organisasi urut (*sequential organization*) dengan pengaksesan secara langsung (*direct access*).
6. File akses langsung (*direct access file*) atau disebut dengan file alamat langsung (*direct address file*) merupakan file dengan organisasi acak (*random organization*) dengan pengaksesan langsung (*direct access*).

RANGKUMAN [3]

7. Organisasi file database dapat berbentuk struktur data berjenjang (*hierarchical data structure*), struktur data jaringan (*network data structure*) dan struktur data hubungan (*relational data structure*).
8. Struktur data hubungan merupakan organisasi file database yang terbaru dan mudah dipahami.
9. Struktur data hubungan mempunyai karakteristik sebagai berikut :
 - a) File dalam bentuk tabel yang persis dengan file urut.
 - b) Hubungan antara record didasarkan pada nilai dari field kunci, bukan berdasarkan alamat atau pointer.

RANGKUMAN [4]

10. *Sequential file (file urut)* merupakan suatu cara ataupun suatu metode penyimpanan dan pembacaan data yang dilakukan secara berurutan. Dalam hal ini, data yang ada akan disimpan sesuai dengan urutan masuknya. Data pertama dengan nomor berapapun, akan disimpan ditempat pertama, demikian pula dengan data berikutnya yang juga akan disimpan ditempat berikutnya.

11. Struktur data memiliki dua komponen dasar, yaitu organisasi dan metode akses. Organisasi mengacu pada cara record disusun secara fisik pada peralatan penyimpanan sekunder. Hal ini bersifat berurutan atau acak. Record dalam file berurutan disimpan di lokasi yang berdekatan yang menempati area tertentu pada disk. Record dalam file acak disimpan tanpa mempertimbangkan hubungan fisiknya dengan record lainnya dari file yang sama. File acak bisa memiliki record yang terdistribusi di seluruh bagian disk. Metode akses adalah teknik yang digunakan untuk menemukan lokasi record dan bernavigasi di basis data atau file. Ada beberapa teknik khusus yang bisa digunakan, namun secara umum, bisa diklasifikasikan sebagai metode akses langsung atau metode akses berurutan.

RANGKUMAN [5]

12. Random file merupakan suatu cara ataupun suatu metode penyimpanan dan pembacaan data yang dilakukan secara random atau langsung. Dengan demikian, random file juga disebut sebagai **Direct Access File** (*Bisa dibaca secara langsung*).

13. *Index Sequential File (File urut Berindeks)* merupakan perpaduan terbaik dari teknik sequential dan random file. Teknik penyimpanan yang dilakukan, menggunakan suatu index yang isinya berupa bagian dari data yang sudah tersortir. Index ini diakhiri dengan adanya suatu pointer (penunjuk) yang bisa menunjukkan secara jelas posisi data yang selengkapnyanya. Index yang ada juga merupakan *record-key (kunci record)*, sehingga kalau *record key ini* dipanggil, maka seluruh data juga akan ikut terpanggil.

-