

FILE PADA COBOL

OBJEKTIF :

1. Mahasiswa Mengetahui Jenis File pada Bahasa COBOL.
2. Mahasiswa Mampu Membuat dan Menjalankan File pada Program Bahasa COBOL.
3. Mahasiswa Mampu Memahami Struktur File Bahasa COBOL.

10.1 ISTILAH ISTILAH DALAM FILE-HANDLING

Untuk memahami File Handling dalam COBOL, sangat penting untuk mengetahui ketentuan dasarnya. Dengan ketentuan ini pun nantinya akan memahami dasar dasar file handling, yang nantinya akan dibahas lebih jauh lagi.

Berikut contoh untuk membantu memahami istilah istilah tersebut :

```
01 DATA-MAHASISWA_.  
05 NPM          PIC 9(8)_.  
05 NAMA         PIC X(30)_.  
05 NILAI        PIC 99.
```

Contoh diatas merupakan file struktur yang dideklarasikan pada FILE SECTION.

FIELD			} RECORD
NPM	NAMA	NILAI	
11111111	ANI	100	
22222222	BUDI	90	
33333333	CITA	56	

File Data Mahasiswa

FIELD

Field atau data item adalah bagian terkecil dari suatu record. Seperti yang ditunjukkan pada contoh diatas yaitu npm, nama dan nilai merupakan sebuah field. Field memiliki beberapa atribut, yaitu :

- Primary keys, untuk field yang bersifat unik pada setiap record dan digunakan untuk mengidentifikasi suatu record. Misalnya, pada file data mahasiswa setiap mahasiswa akan memiliki npm yang unik dan berbeda nilai nya disetiap mahasiswa, inilah yang menunjukkan sebuah primary key.
- Secondary keys, dapat berupa fields yang unik dan tidak unik yang digunakan untuk mencari data yang berkaitan. Contohnya, pada file data mahasiswa nama mahasiswa dapat dijadikan secondary keys jika npm mahasiswa tidak dikenal.
- Descriptors, digunakan untuk mendeskripsikan suatu entitas. Misalnya, pada file data mahasiswa nilai disebut sebagai descriptor.

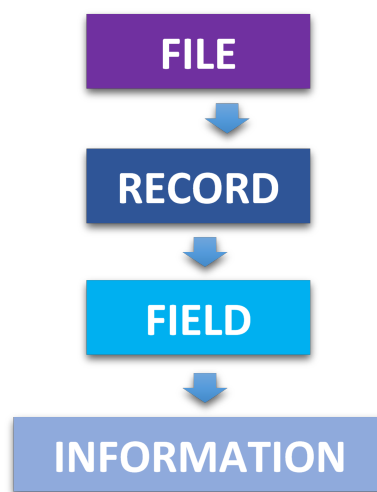
RECORD

Record merupakan kumpulan field yang digunakan untuk mendeskripsikan suatu entitas. Satu atau beberapa field akan membentuk suatu record. Misalnya, dalam file data mahasiswa npm, nama, nilai membentuk suatu record.

FILE

File terdiri dari beberapa record dan record terdiri dari beberapa field. Sebagai contoh, pada file data mahasiswa terdiri dari record semua mahasiswa. File adalah data yang disimpan di lokasi memori, dengan menyimpan data secara permanen dan terstruktur. File biasanya disimpan dalam DISK atau TAPES pada mainframe environment. Maksimal file yang digunakan pada program cobol adalah 255 files.

Perhatikan hirarki file berikut :



Secara hirarki, file dapat didefinisikan sebagai suatu set record yang disimpan sebagai *instances* atau *occurrences*. Record didefinisikan sebagai kombinasi field-field yang *sequence*. Dan Fields dapat didefinisikan sebagai kumpulan informasi yang perlu disimpan.

10.2 FILE SEQUENTIAL

File sequential merupakan bagian dari skema jenis organisasi file. Organisasi file menunjukkan bagaimana suatu record di organisasikan didalam file. Setiap perbedaan jenis file bertujuan untuk meningkatkan efisiensinya dalam mengakses records. File sequential memungkinkan pengaksesan record dalam file secara berurutan. Urutan data record didalam file sama dengan urutan sewaktu data tersebut direkamkan. Setiap data baru yang ditambahkan, urutan ini tidak akan berubah karena data baru yang direkam akan menempati urutan record selanjutnya setelah record terakhir yang telah ada.

Ada dua macam organisasi file secara urut, yaitu :

- Bentuk organisasi *Sequential*, merupakan bentuk default file yang dibuat oleh program MS-COBOL
- Bentuk organisasi *Line Sequential*, yaitu organisasi file yang tiap recordnya diakhiri dengan carriage return atau pembatas line feed.

Attribut key dalam file sequential :

- Record hanya bisa dibaca dan ditulis secara sequential (berurutan), jadi untuk membaca record ke 10, ke 9 record sebelumnya harus sudah dibaca
- Record yang dimasukkan tidak akan mungkin dimodifikasi, modifikasi yang dimaksud seperti menghapus, memperpendek atau memperpanjang record
- Urutan record tidak bisa diubah, setiap data baru ditambahkan urutan tidak akan berubah karena data baru akan menempati urutan record selanjutnya setelah record terakhir yang ada
- Record dapat di update dengan menyimpan recordnya (overwrite) dengan record lain

Informasi mengenai organisasi file yang digunakan dalam COBOL terdapat pada INPUT OUTPUT SECTION di ENVIRONMENT DIVISION.

Bentuk umum File Sequential :

```
INPUT-OUTPUT SECTION_
FILE CONTROL_
    SELECT nama-file ASSIGN TO {DISK/PRINT}
    ORGANIZATION IS SEQUENTIAL
```

MODE AKSES FILE SEQUENTIAL

Sequential Access

Ketika record diakses dalam urutan yang sama dengan saat dimasukkan.

Bentuk umum :

```
ENVIRONMENT DIVISION_
INPUT-OUTPUT SECTION_
FILE CONTROL_
    SELECT nama-file ASSIGN TO {DISK/PRINT}
    ORGANIZATION IS SEQUENTIAL
    ACCESS MODE IS SEQUENTIAL
```

Dynamic Access

Mendukung kedua jenis akses yaitu sequential dan random dalam program yang sama.

Bentuk umum :

```
ENVIRONMENT DIVISION_
INPUT-OUTPUT SECTION_
FILE CONTROL_
    SELECT nama-file ASSIGN TO {DISK/PRINT}
    ORGANIZATION IS SEQUENTIAL
    ACCESS MODE IS DYNAMIC
    RECORD KEY IS primary-key
    ALTERNATE RECORD KEY IS rec-key
```

FILE HANDLING VERBS

Digunakan untuk melakukan berbagai operasi pada file. Dan file handling verbs digunakan untuk setiap jenis file, yaitu file sequential, file index, dan file relative.

Open verb

Open verb adalah operasi file pertama yang harus dilakukan, karena digunakan untuk membuka file yang akan diakses.

Bentuk umum :

```
OPEN 'mode' nama-file
```

File dapat dibuka dalam mode berikut :

Mode	Penjelasan
INPUT	Digunakan pada file yang sudah ada
OUTPUT	Digunakan untuk memasukkan record ke dalam file
EXTEND	Digunakan untuk menambahkan record kedalam file sequential
I-O	Digunakan untuk read dan rewrite record dalam file

Penjelasan :

- Pada mode input, Hanya dapat membaca file, tidak ada operasi lain yang diperbolehkan pada file tersebut.
- Pada mode output, Jika **file sequential** digunakan dan file tersebut menyimpan beberapa record, maka record yang ada akan dihapus terlebih dahulu dan kemudian record baru akan dimasukkan ke dalam file tersebut. Ini tidak akan terjadi dalam kasus **file index** atau **file relatif**.
- Pada mode Extend, record disisipkan di bagian akhir. Jika mode akses file adalah **random** atau **Dinamis**, maka mode extend tidak dapat digunakan.

Read verb

Read verb digunakan untuk membaca record dalam field. Fungsinya mengambil record dari sebuah file. Pada setiap verb yang dideklarasikan, hanya satu record yang dapat dibaca ke dalam struktur file.

Bentuk umum :

```
READ nama-file [NEXT] RECORD [INTO] nama-data  
[AT END statement]  
[NOT AT END statemet]  
END READ_
```

Keterangan :

- **Next Record**, bersifat opsional ditentukan ketika file index sequential sedang dibaca berurutan
- **INTO** clause bersifat optional.
- Kondisi pada **AT END** menjadi TRUE, ketika akhir file tercapai

Write verb

Digunakan untuk menyisipkan record dalam sebuah file. Dan menggunakan opsional FROM untuk langsung menulis record dari data name pada working storage.

Bentuk umum :

```
WRITE nama-record  
  [FROM nama-data]  
  [INVALID KEY statement]  
  [NOT INVALID KEY statement]  
END-WRITE_.
```

Jika mode akses sequential, maka untuk menulis record, file harus dibuka dalam mode Output atau mode Extend. Jika mode aksesnya random atau dynamic, maka untuk menulis record, file harus dibuka dalam mode Output atau mode IO. Penggunaan opsional INVALID dan NOT INVALID KEY digunakan untuk membaca record file Ketika file berorganisasi file indeks dan relative.

Rewrite verb

Digunakan untuk meng-update record. Pernyataan rewrite hanya dapat digunakan setelah operasi Read berhasil dan menimpa record terakhir yang dibaca. Rewrite harus digunakan ketika file dibuka dengan mode I-O.

Bentuk umum :

```
REWRITE nama-record  
  [FROM nama-data]  
  [INVALID KEY statement]  
  [NOT INVALID KEY statement]  
END-REWRITE_.
```

Penggunaan opsional INVALID dan NOT INVALID KEY digunakan untuk membaca record file Ketika file berorganisasi file indeks dan relatif.

Close verb

Digunakan untuk menutup file ketika operasi file selesai diproses. Digunakannya pernyataan close untuk mencegah rusaknya struktur dari file. Setelah melakukan operasi Close, variabel dalam struktur file tidak akan tersedia untuk diproses.

Bentuk umum :

```
CLOSE nama-file
```

10.3 FILE INDEX SEQUENTIAL

File index sequential terdiri dari record yang dapat diakses secara berurutan yang memiliki index. File index sequential juga dapat diakses secara langsung (direct). File ini terdiri dari dua bagian yaitu

- **Data file** berisi record dalam bentuk sequential
- **Index file** berisi primary key dan alamat di data file

Control index berfungsi sebagai penunjuk posisi record tertentu yang berisi data, sehingga memungkinkan menuju ke suatu lokasi record dengan cepat.

Attribut key dalam file index sequential :

- Record dibaca berurutan (sequential)
- Record dapat diakses secara Random (acak) jika primary key diketahui
- Index diurutkan karena menghubungkan nilai kunci dengan posisi record dalam file
- Index alternative juga dapat dibuat untuk mengambil record

Informasi mengenai organisasi file yang digunakan dalam cobol, terdapat pada input output section di Environment Division.

Bentuk umum :

```
INPUT-OUTPUT SECTION_
FILE CONTROL_
    SELECT nama-file ASSIGN TO {DISK/PRINT}
    ORGANIZATION IS INDEXED
    RECORD KEY IS primary-key
    ALTERNATE RECORD KEY IS rec-key
```

MODE AKSES FILE INDEX SEQUENTIAL

Sequential Access

Parameter yang digunakan untuk mengambil record adalah *record key values*.

Bentuk umum :

```
ENVIRONMENT DIVISION_
INPUT-OUTPUT SECTION_
FILE CONTROL_
    SELECT nama-file ASSIGN TO {DISK/PRINT}
    ORGANIZATION IS INDEXED
    ACCESS MODE IS SEQUENTIAL
    RECORD KEY IS primary-key
    ALTERNATE RECORD KEY IS rec-key
```

Random Access

Record diakses berdasarkan nilai yang diletakkan dalam field key yang bisa berupa primary atau alternate key.

Bentuk umum :

```
INPUT-OUTPUT SECTION_
FILE CONTROL_
    SELECT nama-file ASSIGN TO {DISK/PRINT}
    ORGANIZATION IS INDEXED
    ACCESS MODE IS RANDOM
    RECORD KEY IS primary-key
    ALTERNATE RECORD KEY IS rec-key
```

10.4 FILE RELATIVE

File relative terdiri dari record yang diurutkan berdasarkan alamat relatifnya. File secara relative menggunakan direct access dan di organisasikan dengan relative record number. Relative record number tidak mempengaruhi record yang direkam pada file, sehingga panjang record bersifat tetap.

Attribut key dalam file relative :

- Record dapat dibaca berurutan (sequential) seperti pada file sequential dan file index
- Record dapat diakses menggunakan relative key, dimana relative key mewakili lokasi record relatif terhadap alamat awal file.
- Record dapat disisipkan menggunakan relative key.
- File relative memberikan akses tercepat ke record

Kerugian utama dari sistem file ini adalah bahwa jika beberapa record perantara hilang, maka mereka juga akan menempati ruang. Informasi mengenai organisasi file yang digunakan dalam cobol, terdapat pada input output section di Environment Division.

Bentuk umum :

```
INPUT-OUTPUT SECTION_
FILE CONTROL_
    SELECT nama-file ASSIGN TO {DISK/PRINT}
    ORGANIZATION IS RELATIVE
    RELATIVE KEY IS rec-key
```

MODE AKSES FILE RELATIVE

Sequential Access

Relative record key digunakan untuk mengambil record.

Bentuk umum :

```
ENVIRONMENT DIVISION_
INPUT-OUTPUT SECTION_
FILE CONTROL_
    SELECT nama-file ASSIGN TO {DISK/PRINT}
    ORGANIZATION IS RELATIVE
    ACCESS MODE IS SEQUENTIAL
    RELATIVE KEY IS rec-key
```

Random Access

Random Access diambil berdasarkan *relative record keys*.

Bentuk umum :

```
INPUT-OUTPUT SECTION_
FILE CONTROL_
    SELECT nama-file ASSIGN TO {DISK/PRINT}
    ORGANIZATION IS RELATIVE
    ACCESS MODE IS RANDOM
    RELATIVE KEY IS rec-key
```

Dynamic Access

Dalam program yang sama dapat mendukung *sequential* dan *random access*.

Bentuk umum :

```
ENVIRONMENT DIVISION_
INPUT-OUTPUT SECTION_
FILE CONTROL_
    SELECT nama-file ASSIGN TO {DISK/PRINT}
    ORGANIZATION IS RELATIVE
    ACCESS MODE IS DYNAMIC
    RELATIVE KEY IS rec-key
```