

PENANGANAN KESALAHAN & HELP

Tujuan Pembelajaran

- Mampu menyebutkan dan menjelaskan jenis-jenis kesalahan.
- Mampu melakukan pencegahan
- Mampu memperbaiki terhadap kesalahan yang muncul.
- Mengetahui dan mampu menyebutkan jenis dokumen dan alat Bantu.
- Mengetahui isu presentasi
- Mampu menjelaskan tipe dari dokumen dan help, user model, dan pengaturan dokumen

PENANGANAN KESALAHAN

- Kesalahan dalam interaksi dibagi dua:
 1. Kesalahan pada saat implementasi program.
 2. Kesalahan logika ketika program sedang dijalankan atau *run-time error*.

PENANGANAN KESALAHAN

- **Kesalahan pada saat implementasi program:**
adalah kesalahan sintaksis yang secara langsung akan dideteksi oleh *compiler* disebut *compile-time error*. Program tidak dapat di eksekusi sebelum kesalahan dibetulkan.

PENANGANAN KESALAHAN

- Kesalahan logika ketika program sedang dijalankan atau *run-time error*:
adalah kesalahan yg dapat menyebabkan program terhenti secara abnormal (*abnormal termination*) Kesalahan logika tidak bisa dideteksi langsung oleh *compiler*, sehingga penanganannya kadang-kadang cukup sulit.

PENANGANAN KESALAHAN

- **Penyebab kesalahan logika:**
 - Datang dari pengguna, misalnya pengguna mengisikan data yang tidak sah. Contohnya ?
 - Berasal dari proses eksekusi, misalnya ketika program perlu membaca data dari suatu *file* ternyata *file* tidak ada, atau ketika terjadi pembagian dengan nol, atau terjadi kekurangan memori ketika program akan mengalokasikan peubah dinamis.

PENANGANAN KESALAHAN

- Penanganan kesalahan dilakukan dengan memberi pesan umpan balik kepada pengguna akan keadaan eksekusi program pada saat itu. Didalam program ada modul perangkap (*error trapping*), contoh:

```
var A, B, C1: integer;
```

```
Begin
```

```
readln (A);
```

```
readln (B);
```

```
If B=0 then
```

```
writeln ('Pembagian dengan nol dilarang');
```

```
else
```

```
begin
```

```
C1:= A div B;
```

```
writeln (C1);
```

```
end
```

```
end
```

MERANCANG PESAN KESALAHAN

- Dalam merancang suatu pesan kesalahan, pesan tersebut harus bisa memberikan informasi dengan singkat dan jelas mengenai:
 - Dimana terjadi kesalahan?
 - Apa yang salah?
 - Bagaimana memperbaiki kesalahan?

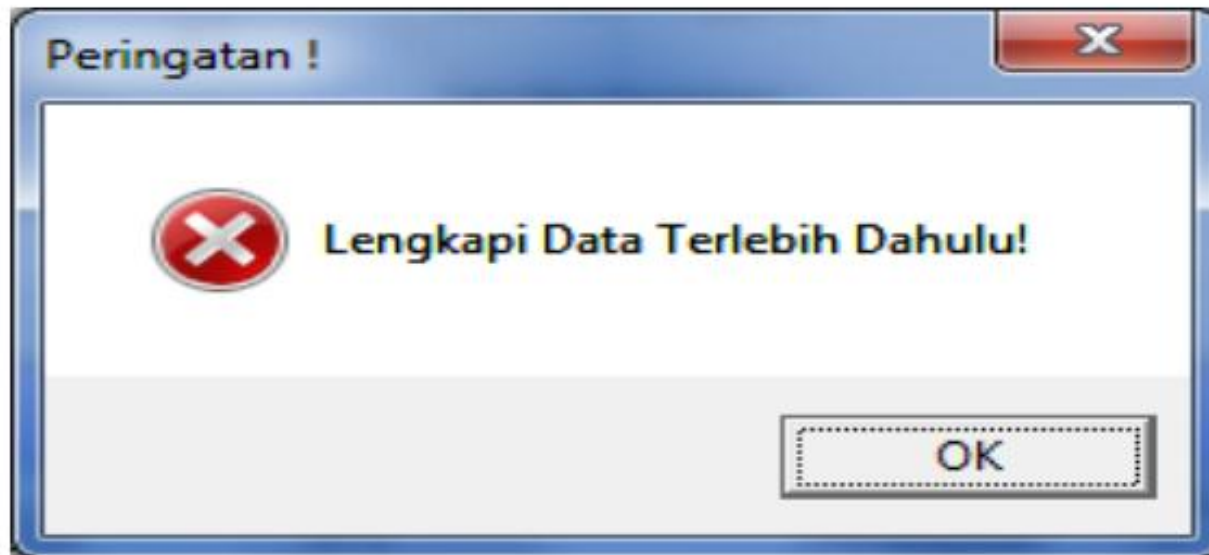
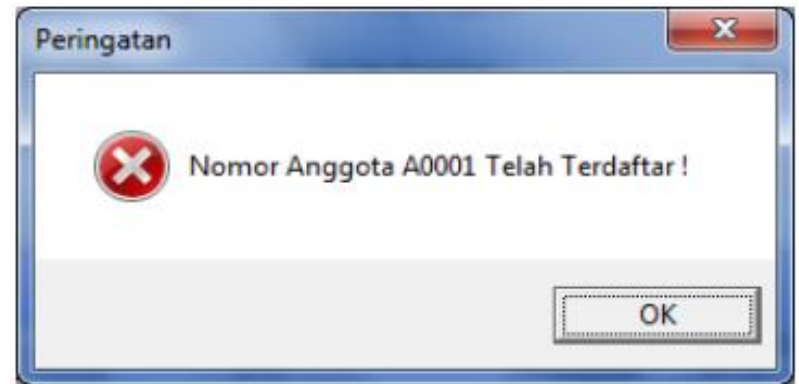
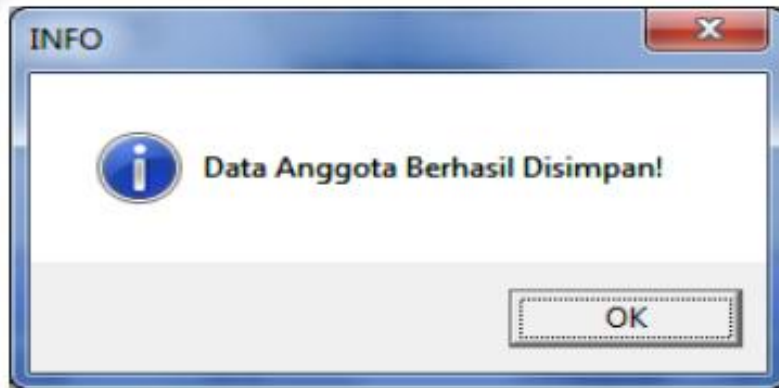
MERANCANG PESAN KESALAHAN

- Pertimbangan dan pedoman dalam merancang pesan kesalahan:
 - Pesan kesalahan singkat dan jelas.
 - Tidak menyalahkan pemakai.
 - Terletak pada tempat yang konsisten.
 - Kemunculannya menarik perhatian pemakai, misalnya dengan bunyi tetapi bunyi tersebut bisa dimatikan.
 - Tidak muncul sebelum pemakai selesai memberi perintah/ menyelesaikan transaksi.
 - Diuji coba untuk mengetahui penerimaan pemakai terhadap pesan kesalahan.
 - Tidak terlalu umum/ harus spesifik dan sesuai dengan kesalahan yang terjadi.

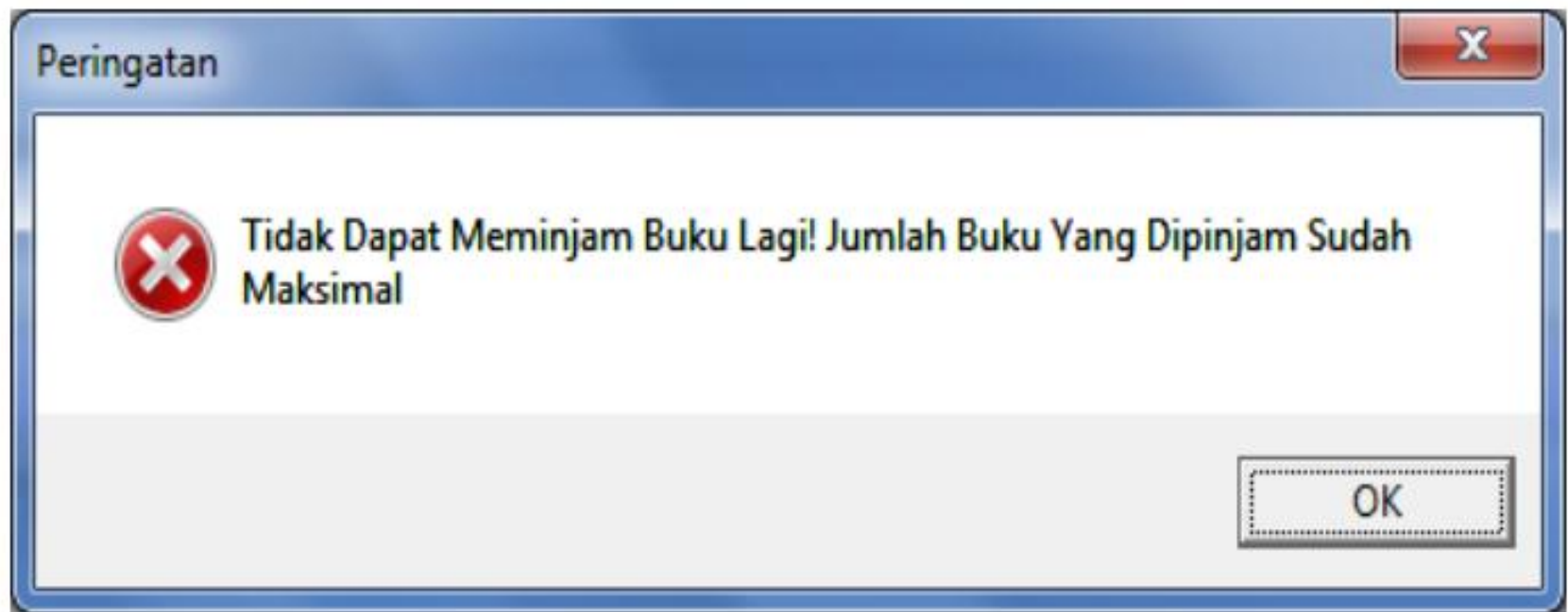
CONTOH



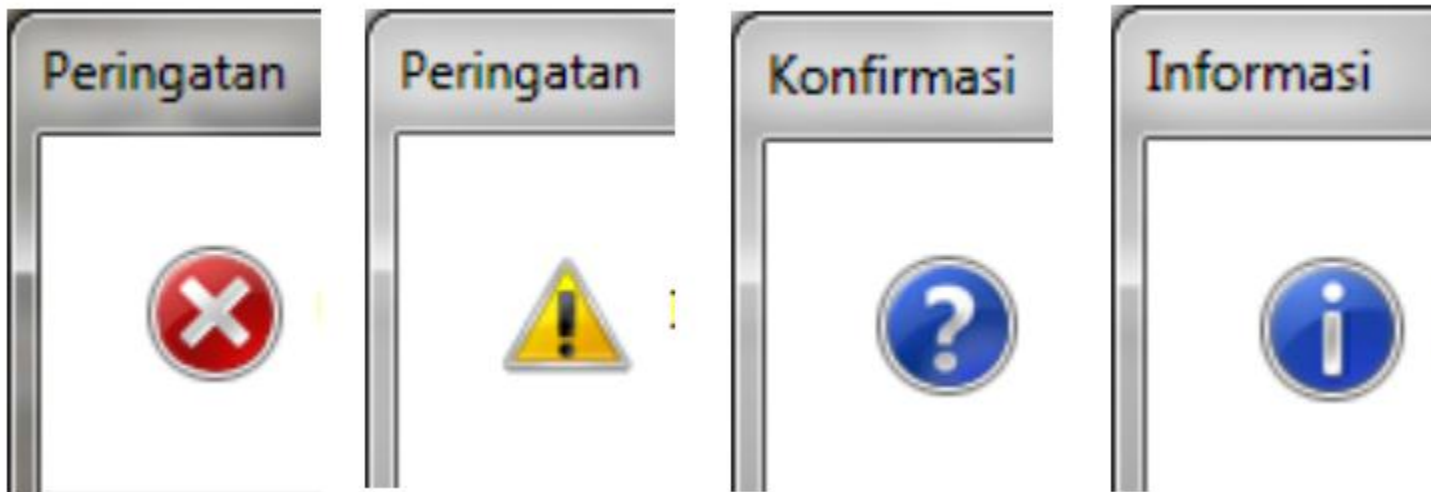
CONTOH



CONTOH



CONTOH



PERANCANGAN FASILITAS HELP & MANUAL

- Bahan-bahan untuk membantu pemakai dalam menggunakan sistem:
 1. Manual (pedoman) pemakai: dokumen yang menjelaskan tentang sistem, berupa:
 - Daftar dan deskripsi perintah.
 - Referensi mengenai sintaks.
 - Pengenalan/ pengajaran bagi pemakai awam.
 - Pedoman konversi yang mengajarkan ciri-ciri dari sistem kepada pemakai yang telah mengetahui sistem sejenis lainnya.

PERANCANGAN FASILITAS HELP & MANUAL

- Bahan-bahan untuk membantu pemakai dalam menggunakan sistem:
 2. Bahan yang dibuat pada komputer, berupa:
 - Pedoman langsung bagi pemakai.
 - Bantuan langsung pada pemakai.
 - Pengajaran langsung pada pemakai.

PERANCANGAN FASILITAS HELP & MANUAL

- Perancangan manual menggunakan media kertas sebaiknya dibuat sebelum implementasi atau bersamaan dengan perancangan sistem antarmuka dengan maksud:
 - Pemakai mengetahui pekerjaan.
 - Pemakai mengetahui pekerjaan dan penyuntingan.
 - Pemakai mengetahui pekerjaan, penyuntingan dan sintaks (**pemakai diingatkan sintaksnya**).

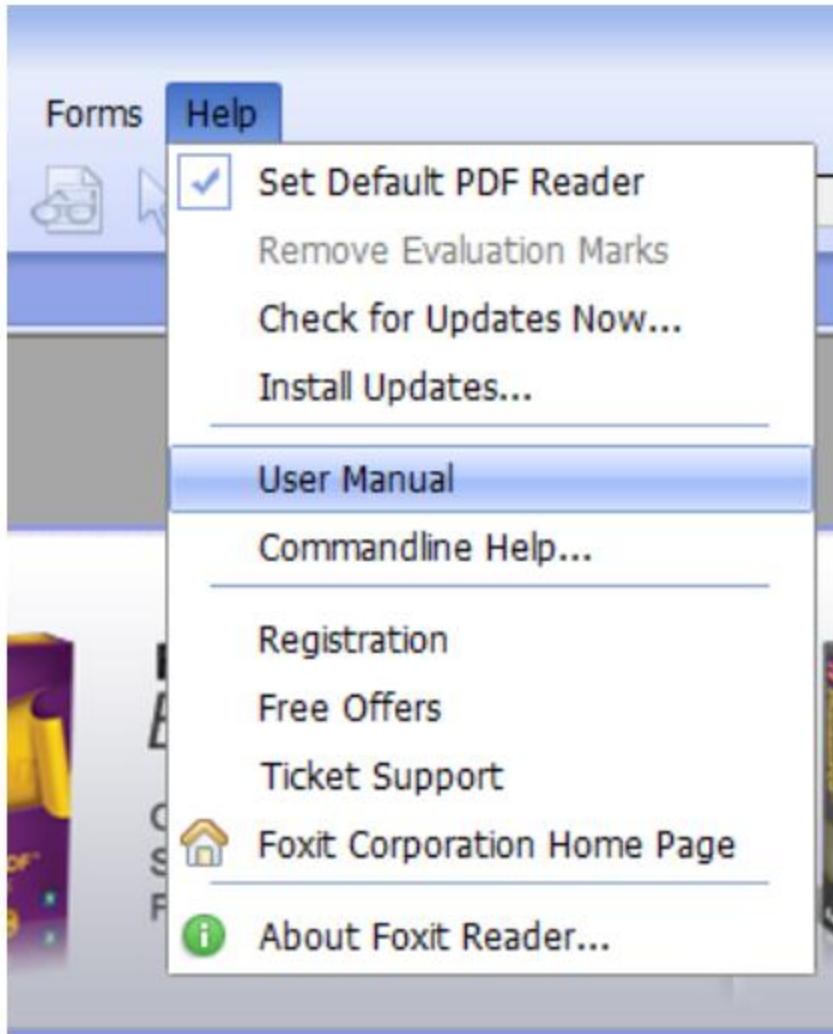
PERANCANGAN FASILITAS HELP & MANUAL

- Perancangan kertas manual sebaiknya memperhatikan hal-hal sebagai berikut:
 - Membuat informasi mudah ditemukan.
 - Informasi jelas dan mudah dimengerti.
 - Informasi lengkap tapi hanya informasi yang dibutuhkan dan informasi tersebut adalah informasi yang terbaru dan benar.
 - Konsisten.

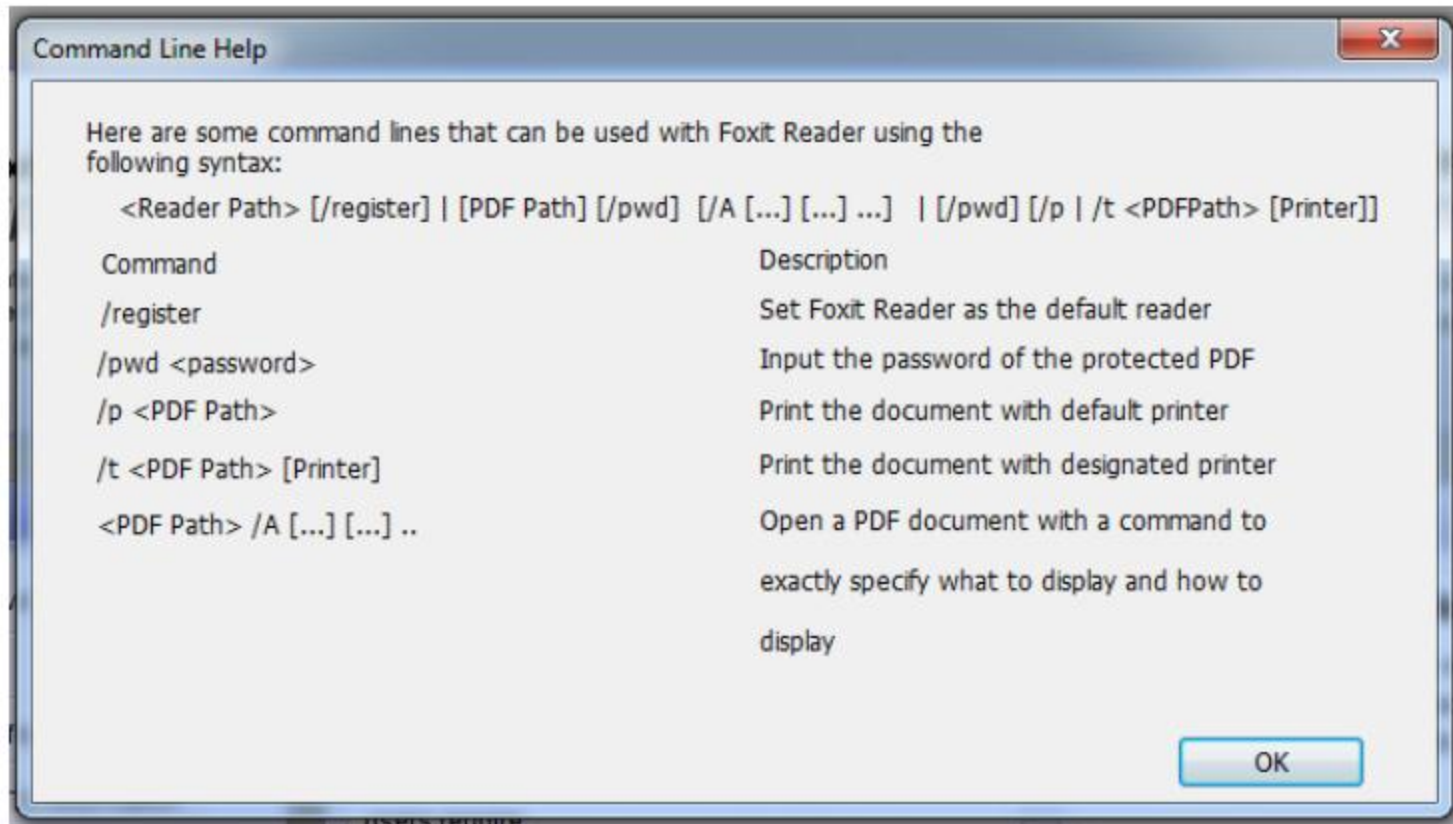
PERANCANGAN FASILITAS HELP & MANUAL

- Perancangan fasilitas bantuan langsung: terintegrasi dengan sistem atau merupakan sistem tersendiri dengan memperhatikan:
 - Kontrol penggunaan oleh pemakai (cara pemanggilan dan keluar dari fasilitas tersebut).
 - Penyajian tidak mengganggu pekerjaan (ada jendela khusus).
 - Pemanggilan bisa setiap saat.
 - Singkat, jelas, konsisten dan mudah dimengerti.

CONTOH



CONTOH



CONTOH

- Contoh help dengan menekan tombol F1

