



SMA NEGERI 3 MALANG

Jl. Sultan Agung Utara No.7 Telp (0341) 324768, Fax (0341) 341530

Website: <http://www.sman3malang.sch.id>

E - mail : humas@sman3malang.sch.id

Handout & Lembar Kerja Database – 2

Table and Practice 2 (Manage Table)



Nama:	Kelas / No: XI IPA-Aks /
-------------	--------------------------------

**Cambridge Syllabus Adaptation Section 4:
ICT Core Module (Standard Level)**
Data Manipulation (4.3): *Using database facilities to manipulate data to solve problems and represent data graphically.*

Objectives:

- *Identifying Principles of Database*
- *Database Management*

Performance Criteria:

- *Using database program & creating table*

Subjective:

- *Using database program*
- *Creating tables*

Copyright © 2011. Wibisono Sukmo Wardhono, ST, MT

Sebagian maupun seluruh isi dokumen ini dapat disadur, disalin dan disebarluaskan untuk keperluan komersial maupun non-komersial tanpa harus izin kepada penulis dengan tetap mencantumkan pernyataan copyleft dan lisensi ini. Ketentuan lisensi tersedia di <http://www.fsf.org/licensing/licenses/fdl.html>

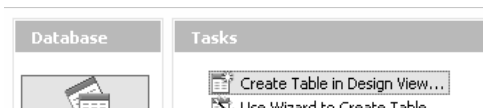
Perhatian! Langkah-langkah pada halaman 1 ini adalah materi yang sama persis dengan halaman 3 worksheet minggu sebelumnya dan dimuat kembali untuk mendukung kegiatan praktik pada hari ini. Materi UTS TIK kelas XI-Aksel sampai dengan materi praktik pada worksheet ini, ikuti baik2 petunjuk Guru yang akan mendampingi kalian selama kegiatan praktik.

BEKERJALAH SECARA BERKELOMPOK DENGAN ANGGOTA DUA ORANG, KEMUDIAN BUATLAH SEBUAH DATABASE MENGGUNAKAN OOo BASE DENGAN NAMA: **db01_xx_xx.odt** (xx = no. absen)

Membentuk Tabel pada OOo Base

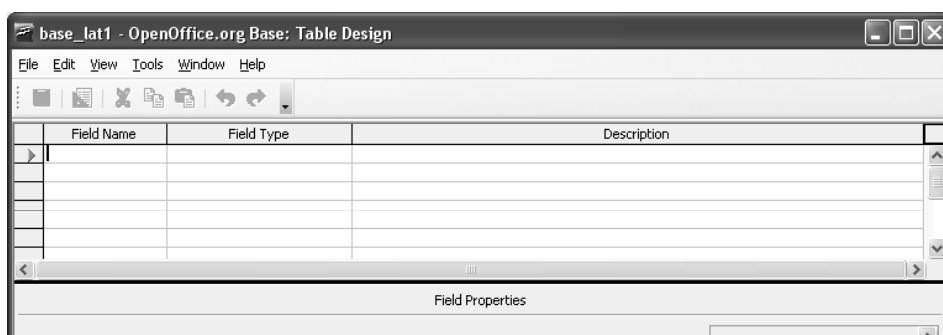
Langkah-langkah membuat tabel melalui "Design View"

- Klik "Tables" pada menu obyek, lalu klik "Create Table in Design View..."



Gambar 1.1. Memulai pembuatan tabel dengan menggunakan wizard

- Setelah itu akan ditampilkan jendela "Tables Design" seperti pada gambar 1.2



Gambar 1.2. Jendela "Table Design" pada OOo Base


- Ikutilah petunjuk guru kalian, isikan **Field Properties** dengan data-data di bawah ini:

Field Name	Field Type	Entry Required	Length
No	Number (NUMERIC)	No	2
Nama	Text (fix) (CHAR)	Yes	20
Alamat	Text (VARCHAR)	No	50
JumlahSaudara	Decimal (DECIMAL)	Yes	2

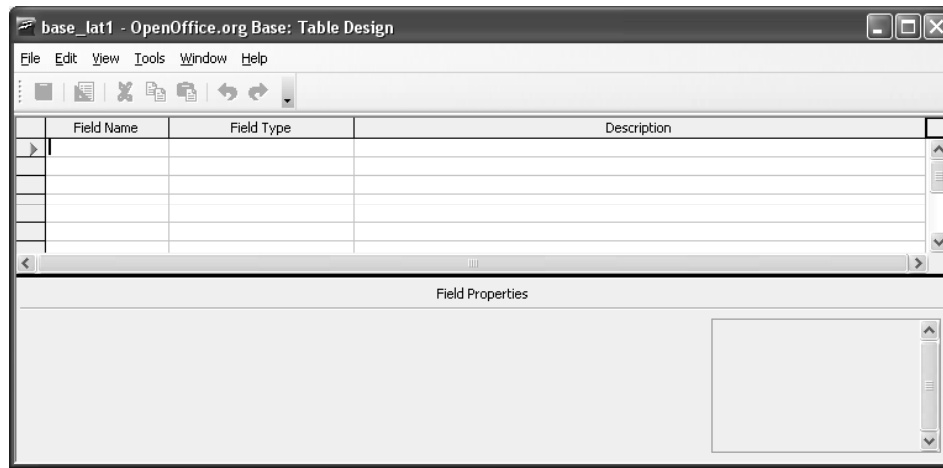
- Klik area sebelah kiri field "No", lalu klik kanan dan pilih *primary key*, ikuti petunjuk gambar 1.3. (a) dan (b). Field tersebut akan ditandai dengan ikon berbentuk kunci berwarna kuning.



Gambar 1.3. (a) Menentukan PK pada field "no" (b) Tanda PK pada field "No"

- Bila tabel selesai diolah, simpan dengan memilih menu File > Save atau klik tombol , lalu beri nama tabel tersebut, dalam contoh ini, kita beri nama "TableSiswa"
- Tutuplah Properti tabel dengan klik tombol [X] di bagian pojok kanan atas

Pengaturan tabel



Gambar 1.4. Jendela "Table Design" pada OOo Base

Pada jendela "Table Design" dapat ditentukan field-field dan tipe yang dibutuhkan. Bagian-bagian yang terdapat pada jendela "Table Design" adalah sebagai berikut:

- **Field Name.** Berisi kolom tempat membuat nama field yang diperlukan tabel, seperti "no_induk", "nama", "alamat" dll
- **Field Type.** Merupakan tipe data yang digunakan untuk masing-masing *field*. Contoh; tipe "Text [VARCHAR]" untuk *field* "nama"
- **Description.** Yaitu deskripsi singkat mengenai *field*. Sifatnya opsional, boleh dikosongkan
- **Field Properties.** Terdapat pada bagian bawah jendela "table design" dan digunakan untuk menentukan entri, panjang data, format data dan sebagainya.

Tipe Data pada OOo Base 3.2

Pembangunan sebuah basis-data harus mempertimbangkan efisiensi dalam penggunaan memori (ruang penyimpanan). Untuk itu harus ditentukan tipe data yang tepat untuk setiap *field*. Tipe data yang tersedia pada OOo Base adalah sebagai berikut:

Tipe Data	Kapasitas Memori	Jenis Data
Tiny Integer [TINYINT]	3 byte dinamis	Bilangan bulat (maks 3 digit)
BigInt [BIGINT]	19 byte dinamis	Bilangan bulat (maks 19 digit)
Image [LONGVARBINARY]	2147483647 byte dinamis	File gambar
Binary [VARBINARY]	2147483647 byte dinamis	Data biner
Binary (fix) [BINARY]	2147483647 byte dinamis	Data biner
Memo [LONGVARCHAR]	2147483647 byte dinamis	Karakter bebas, berupa teks dan angka yang tidak diperlukan dalam perhitungan matematis, biasanya digunakan untuk menuliskan keterangan
Text (fix) [CHAR]	Ditentukan pada length melalui properti field (0-2147483647 karakter)	Karakter bebas, berupa teks dan angka yang tidak diperlukan dalam perhitungan matematis
Number [NUMERIC]	Ditentukan pada length melalui properti field (0-646456993 digit)	angka yang diperlukan dalam perhitungan matematis berupa bilangan bulat
Decimal [DECIMAL]	Ditentukan pada length melalui properti field (0-646456993 digit)	angka yang diperlukan dalam perhitungan matematis, berupa bilangan desimal
Integer [INTEGER]	10 byte dinamis	Bilangan bulat (maks 10 digit)
Small Integer [SMALLINT]	5 byte dinamis	Bilangan bulat (maks 5 digit)
Float [FLOAT]	17 byte dinamis	Bilangan pecahan (maks 17 digit)

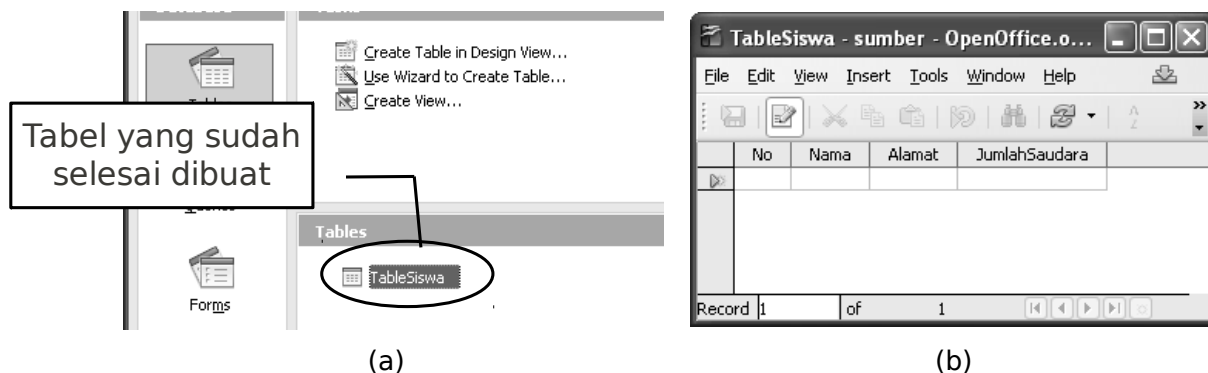
Real [REAL]	17 byte dinamis	Bilangan real (maks 17 digit)
Double [DOUBLE]	17 byte dinamis	Bilangan real (maks 17 digit)
Text [VARCHAR]	Ditentukan pada length melalui properti field (default 17 byte)	Karakter bebas, berupa teks dan angka yang tidak diperlukan dalam perhitungan matematis dengan memperhatikan huruf besar-kecil
Text [VARCHAR_IGNORECASE]	Ditentukan pada length melalui properti field (default 17 byte)	Karakter bebas, berupa teks dan angka yang tidak diperlukan dalam perhitungan matematis tanpa memperhatikan huruf besar-kecil
Yes/No [BOOLEAN]	1 bit statis	Pernyataan logika kebenaran dua kondisi, True atau False, 1 atau 0
Date [DATE]	8 byte statis	Data berupa tanggal atau jam
Time [TIME]	8 byte statis	Data berupa tanggal atau jam
OTHER [OTHER]	2147483647 byte dinamis	Data berupa nilai lain yang tidak terdapat pada tipe data yang lain

Penentuan kapasitas data untuk Efisiensi Memori

Pada pembangunan sebuah basis-data, harus diperhatikan kapasitas data yang akan disimpan. Tipe data yang akan diberikan untuk setiap *field* harus dipertimbangkan sejak awal. Setiap kapasitas tipe data yang ditentukan pada *field* akan secara otomatis dialokasikan oleh sistem ke dalam memori untuk setiap *record* data. Misal, bila ditentukan sebuah *field* "nama" dengan tipe data "text" dan kapasitas 25 karakter, maka maksimal record data yang dimasukkan adalah 25 karakter, kapasitas tersebut dialokasikan oleh sistem tiap kali dilakukan *record* data baru pada *field* "nama".

Memasukkan Record pada tabel

Untuk mengisikan record, klik-gandalah tabel yang telah kalian buat seperti ditunjukkan pada gambar 1.5 (a) dan perhatikan tabel pada gambar 1.5 (b)



Gambar 1.5. Posisi tabel yang sudah selesai dibuat dan tabel yang siap diisi record

Kemudian isikan *record* seperti yang diminta pada tabel di bawah ini, bila terdapat *error*, tuliskan pesan *error*-nya dan analisis *error* tersebut pada kolom kesimpulan.

No	Nama	Alamat	Jumlah Saudara	Ada Error	Pesan Error	Kesimpulan
1	Wibisono	Tlogomas	0			
2	Wibisono Sukmo Wardhono	Merjosari	0			

3	(kosongi)	Blimbing	3			
7	Rahadian	(kosongi)	3			
9	Noorman	Singosari	(kosongi)			
7	Adhi	Blimbing	4			

Perhatikan penjelasan guru kalian, lalu jawablah pertanyaan-pertanyaan di bawah ini:

1. Tuliskan Perbedaan penggunaan tipe "Number" dengan "Decimal"
.....
.....
2. Tuliskan Perbedaan penggunaan tipe "Text (fix) (CHAR)" dengan "Text (VARCHAR)"
.....
.....
3. Jelaskan mengenai <i>setting</i> "Entry Required"!
.....
.....
4. a. Jelaskan mengenai <i>setting</i> "Length"! b. Berapa angka terbesar yang bisa dimasukkan pada <i>field</i> "No"?
.....
.....
.....
5. Tuliskan karakter-karakter field yang memiliki sifat "Primary Key"
.....
.....