

## MODUL II

### Instalasi OracleXE Dan SQL

#### I. Tujuan

1. Dapat menginstalasi aplikasi Oracle XE
2. Dapat mengeset user baru pada aplikasi Oracle XE
3. Dapat mengubah password username "hr" dan mengaktifkannya
4. Dapat menjalankan perintah-perintah SQL

#### II. Perlengkapan yang dibutuhkan:

1. Komputer Set (memory >= 1GB, CPU >= PIV)
2. Installer OracleXE

#### III. Dasar Teori

Dalam dunia database ada beberapa tool yang dapat digunakan untuk membantu kita dalam membangun aplikasi database. Tool yang digunakan dapat dibedakan berdasarkan pada tahapan-tahapan pengembangan database. Secara umum tahapan dalam pengembangan database adalah:

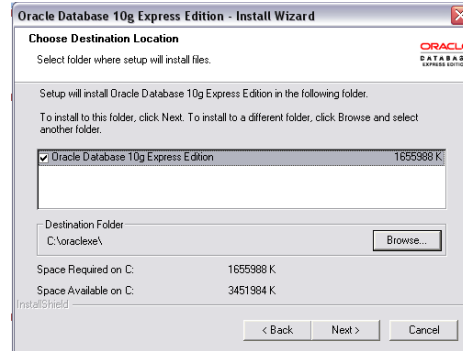
1. Disain database
2. Implementasi Database
3. Penyimpanan, Ekstrasi, Modifikasi dan Perawatan Database

Dari 3 tahapan tersebut kita bisa dibantu oleh 2 buah tool, yakni sebuah disain tool dan sebuah aplikasi DBMS. Untuk keperluan disain kita bisa dibantu oleh aplikasi Sybase Power Designer, meskipun ada banyak varian aplikasi lain yang dapat kita gunakan. Sedangkan untuk aplikasi database kita gunakan OracleXE

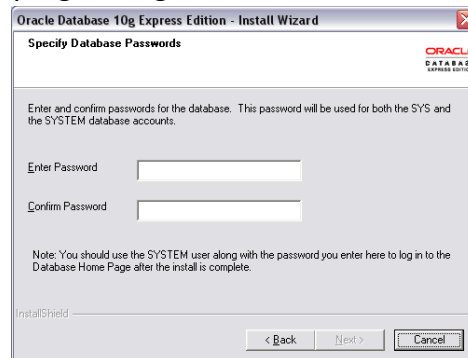
#### IV. Langkah – Langkah Praktikum

##### 1. Instalasi Oracle XE

- i. Jika aplikasi Oracle XE sudah ada, maka uninstall terlebih dahulu
- ii. Jalankan file "OracleXEUniv.exe"
- iii. Tentukan folder lokasi instalasi



- iv. Masukkan password untuk user name SYS dan SYSTEM, catat password yang anda gunakan.



- v. Lanjutkan instalasi dan tekan tombol instal

## 2. Membuat user baru pada aplikasi Oracle XE

- i. Buka halaman website <http://127.0.0.1:8080/apex>
- ii. Login menggunakan username system dan password yang sudah anda tentukan
- iii. Pilih menu Administrator → Database User → Create User
- iv. Buat User Baru (username gunakan nama depan, password gunakan nim)
- v. Logout
- vi. Kemudian Login menggunakan user baru

## 3. Mengubah password untuk user name "hr"

- i. Buka halaman website <http://127.0.0.1:8080/apex>
- ii. Login menggunakan username system dan password yang sudah anda tentukan
- iii. Pilih menu Administrator → Database User → Manage Users
- iv. Pilih User "hr", ketikkan password baru
- v. Aktifkan user "hr"
- vi. Logout
- vii. Kemudian Login menggunakan user "hr"

#### 4. Tablespace

Untuk menyimpan obyek-obyek database dibutuhkan suatu ruang memori yang disebut **TABLESPACE**. Untuk mengakses tablespace dibutuhkan sebuah *account* user. Berikut ini perintah SQL untuk membuat tablespace, user dan perintah untuk manipulasi tabel.

- i. Melihat tablespace

```
select * from user_free_space;
```

- ii. Membuat tablespace

```
SQL>Create tablespace nama-tablespace datafile 'nama-datafile.dat' size 1M  
autoextend on next 500K maxsize 2M;
```

- iii. Mengubah ukuran tablespace

```
SQL>alter database datafile 'nama-datafile.dat' resize 2M;
```

- iv. Menambah ukuran tablespace dengan cara menambahkan data file baru

```
SQL>alter tablespace dataku add datafile 'nama-datafile.dat' size 2M;
```

- v. Menghapus table space

```
SQL>drop tablespace nama-tablespace including contents;
```

#### 5. Membuat *Account*

Setelah Tablespace siap, maka langkah selanjutnya membuat *account* baru. Account ini akan anda gunakan selama praktikum. *Account* yang sudah ada ketika instalasi selesai adalah SYS, SYSTEM dan SCOTT. Untuk membuat *account* baru kita mempergunakan user SYS dan SYSTEM. Penggunaan *tablespace* yang tadi telah anda buat untuk menyimpan data anda selama praktikum.

- i. Membuat user baru dan dengan tablespace yang dipakai misalnya dtpraktek.

```
SQL>create user nama_user identified by  
password-user default tablespace  
dtpraktek;
```

- ii. Parameter **nama\_user** adalah nama login, dan **password-user** adalah password user yang dipilih. Misalnya dipilih untuk **nama\_user** user01. Berikan hak akses user pada database. Misalnya hak akses **grant** dan **resource**.

```
SQL> grant connect to user01;  
SQL> grant resource to user01;
```

- iii. Langkah berikutnya, lakukan sambungan dengan server dengan mempergunakan **user\_name** yang sudah dibuat tadi.

```
SQL> connect user01
```

```
Enter password: ***  
Connected.
```

## 6. Membuat Tabel

- i. Saat ini sudah tersambung dengan server dengan mempergunakan user01. Buat tabel mahasiswa yang memiliki struktur nim varchar(5), nama varchar(25), alamat varchar(30).

```
SQL> create table mahasiswa (nim varchar(5), nama varchar(25),  
alamat varchar(30));
```

- ii. Untuk melihat tabel yang ada dalam tablespace mempergunakan perintah :

```
SQL> select * from tab;
```

- iii. Untuk melihat struktur tabel dapat dilakukan dengan mempergunakan perintah :

```
SQL> describe mahasiswa;
```

- iv. Perintah alter berguna untuk melakukan perubahan tabel diantaranya merubah tipe field, menambah field baru dan merubah nama tabel.

Merubah tipe field tabel :

```
SQL> alter table mahasiswa modify (nim number(5));
```

Menambah field tabel :

```
SQL> alter table mahasiswa add (usia number(2));
```

Mengganti nama tabel :

```
SQL>alter table mahasiswa rename to mhs;
```

- v. Untuk menambahkan data kedalam tabel, penggunaan perintah insert

```
SQL>insert into mahasiswa values(1,'Sumanto','Sleman');
```

- vi. Data yang sudah disimpan dalam tabel dapat diedit dengan mempergunakan perintah update.

```
SQL>update mahasiswa set nama = 'Sumanto' where nim=1;
```

- vii. Untuk keperluan penghapusan data, penggunaan perintah :

```
SQL>delete from mahasiswa where nim=1;
```

## 7. Constraint

Constraint (konstrain) adalah aturan yang diberikan pada suatu tabel supaya data yang dimasukkan lebih terjamin validitasnya.

- i. Memberi konstrain pada saat membuat tabel :

```
SQL>create table mahasiswa(nim char(5) constraint pnim primary key, nama char(20) constraint nnama not null, alamat char(30));
```

- ii. Memberi konstrain pada tabel yang sudah ada :

```
SQL>alter table mahasiswa add constraint pnim primary key (nim);
```

```
SQL>alter table mahasiswa modify (nama char(20) constraint nnama not null);
```

- iii. Menghapus konstrain :

```
SQL>alter table mahasiswa drop constraint nnama;
```

## 8. Order By

Klausa ORDER BY digunakan untuk mengurutkan data yang di-query. Klausa ORDER BY diikuti dengan nama kolom atau kolom alias kemudian diikuti klausa jenis urutan yang kita inginkan. Jenis urutan bisa ASC (ascending) atau DESC (descending). Sebagai ilustrasi, tampilkan semua data dengan cara diurutkan ascending berdasarkan nama mahasiswa.

```
SQL> select * from mahasiswa order by nama;
```

```
SQL> select * from mahasiswa order by nama desc;
```

## 9. Where

Klausula LIKE WHERE digunakan untuk membatasi query menurut kriteria tertentu. Kriteria ini didefinisikan menggunakan bentuk perbandingan. Sebagai ilustrasi, tampilkan data mahasiswa yang berasal dari Jogjakarta.

```
SQL> select * from mahasiswa where alamat = 'Jogjakarta';
```

## 10. Like

Klausula LIKE digunakan untuk membatasi query menurut nilai tertentu. Nilai ini merupakan sebagian dari nilai suatu field. Sebagai ilustrasi, akan ditampilkan data mahasiswa yang memiliki nama yang nama akhirnya mengandung huruf "ni".

```
SQL> select * from mahasiswa where nama like '%ni';
```

## TUGAS MINGGU INI

1. Tampilkan data mahasiswa yang tersimpan dalam tabel mahasiswa.
2. Tampilkan data mahasiswa yang berusia 20 tahun.
3. Tampilkan data mahasiswa yang berasal dari Jogjakarta.
4. Tampilkan data mahasiswa yang namanya mengandung nilai "ni" dibelakang (disesuaikan dengan yang telah dimasukkan) dan berasal dari Jakarta.
5. Tampilkan jumlah mahasiswa berdasarkan alamat.