

Nama : Rifqi Alwan Pramasandy

NIM : L200180008

Kelas A

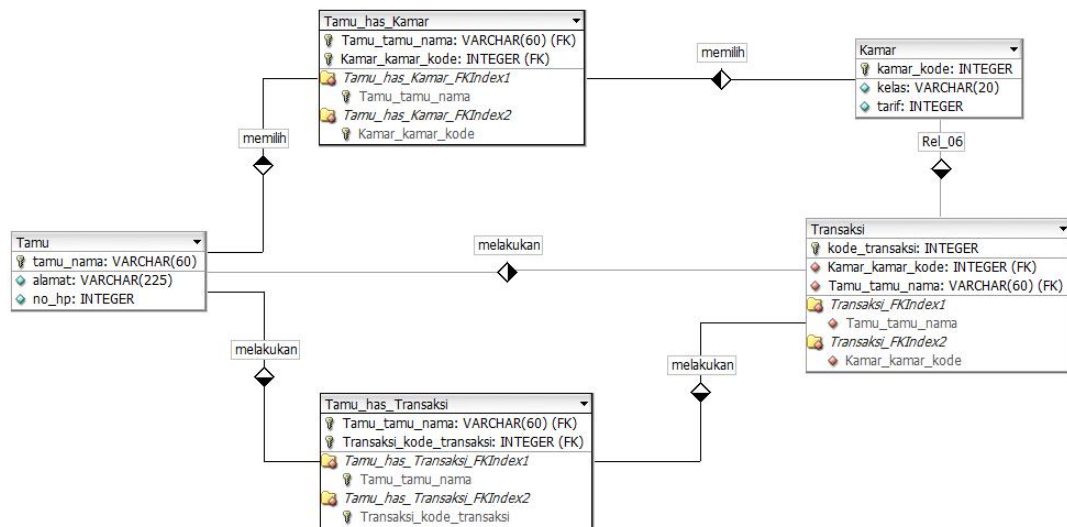
Modul 12

APLIKASI DENGAN DATABASE

1. Membuat aplikasi berbasis java pada salah satu studi kasus yaitu perhotelan, dengan entitas yang ada antara lain :
 - a. Tamu
 - b. Kamar
 - c. Transaksi
 - d. Tamu_has_transaksi
 - e. Tamu_has_kamar

2. Penjelasan dari studi kasus
 - a. Entitas Tamu memiliki attribute :
 - nama_tamu
 - alamat_tamu
 - no_hp
 - b. Entitas Kamar memiliki attribute :
 - kode_kamar
 - kelas
 - tarif
 - c. Entitas Transaksi memiliki attribute :
 - kode_transaksi
 - kode_kamar
 - nama_tamu
 - d. Entitas Tamu_has_transaksi memiliki attribute :
 - kode_transaksi
 - tamu_nama
 - e. Entitas Tamu_has_kamar memiliki attribute :
 - tamu_nama
 - kamar_kode

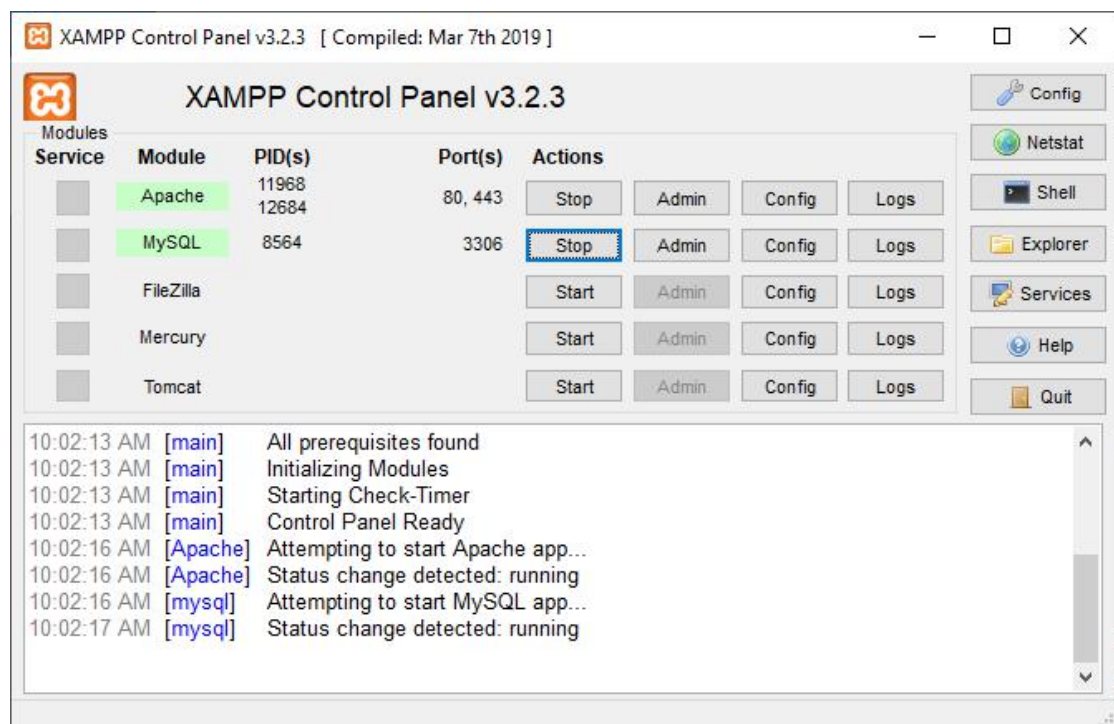
3. ER Diagram



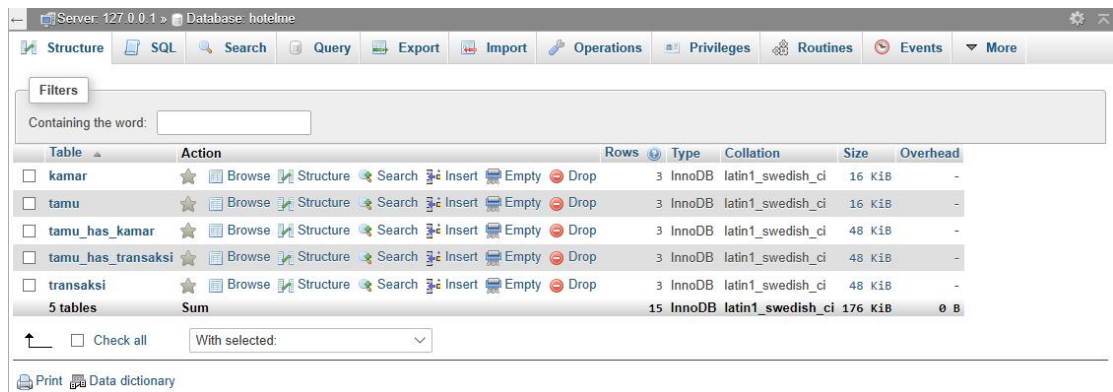
Dari ER Diagram tersebut, hubungan antara Tamu dan Kamar adalah n:n, Tamu dan Transaksi adalah n:n, sedangkan Kamar dan Transaksi adalah 1:n.

4. Membuat aplikasi menggunakan netbeans

a. Menjalankan xampp control panel



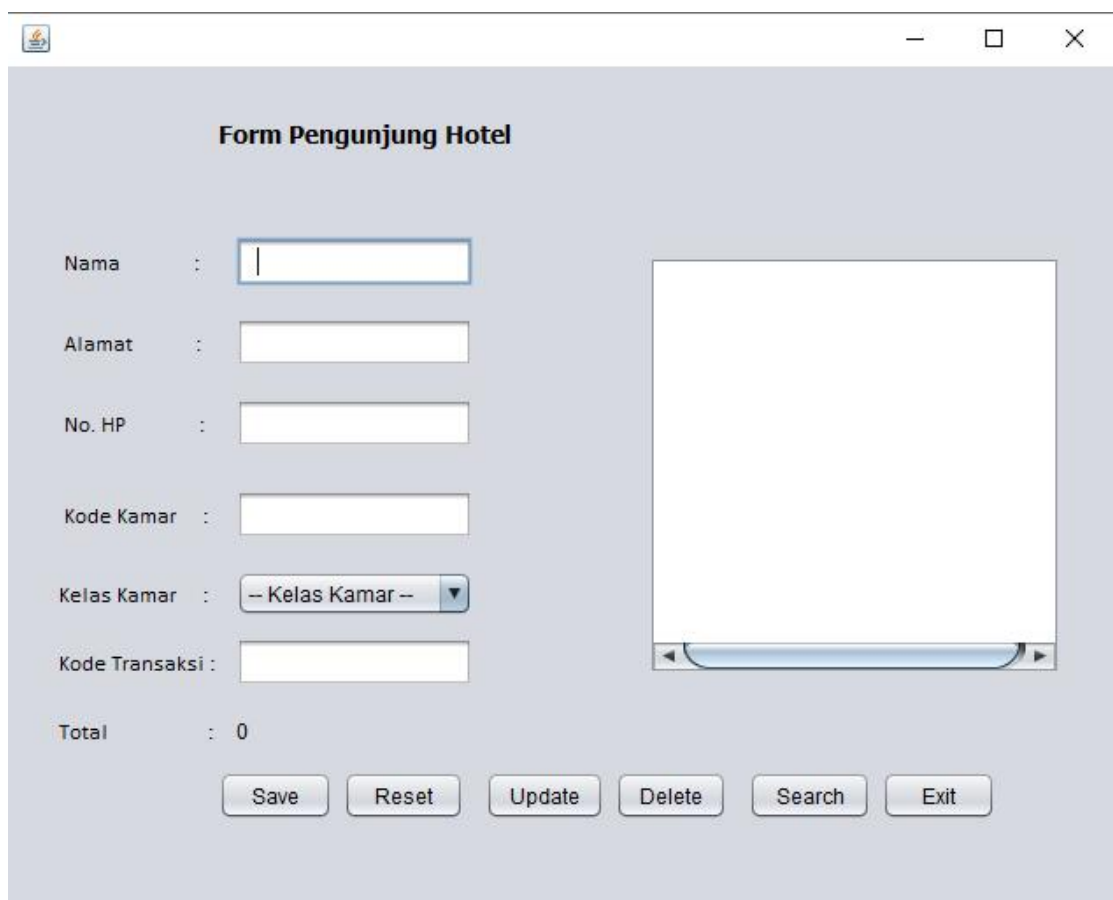
b. Membuat database beserta tabel yang dipakai



The screenshot shows the MySQL Database Structure window for a database named 'hotelm'. The window has a menu bar with options: Structure, SQL, Search, Query, Export, Import, Operations, Privileges, Routines, Events, and More. Below the menu bar is a 'Filters' section with a text input field 'Containing the word:'. The main area displays a table of database tables with columns: Table, Action, Rows, Type, Collation, Size, and Overhead. The table lists five tables: kamar, tamu, tamu_has_kamar, tamu_has_transaksi, and transaksi. Each table has a 'Sum' row showing the total rows, type, collation, size, and overhead. At the bottom, there is a 'Check all' checkbox and a 'With selected:' dropdown menu. A 'Print' button and a 'Data dictionary' link are also visible.

Table	Action	Rows	Type	Collation	Size	Overhead
kamar	★ Browse Structure Search Insert Empty Drop	3	InnoDB	latin1_swedish_ci	16 KiB	-
tamu	★ Browse Structure Search Insert Empty Drop	3	InnoDB	latin1_swedish_ci	16 KiB	-
tamu_has_kamar	★ Browse Structure Search Insert Empty Drop	3	InnoDB	latin1_swedish_ci	48 KiB	-
tamu_has_transaksi	★ Browse Structure Search Insert Empty Drop	3	InnoDB	latin1_swedish_ci	48 KiB	-
transaksi	★ Browse Structure Search Insert Empty Drop	3	InnoDB	latin1_swedish_ci	48 KiB	-
5 tables	Sum	15	InnoDB	latin1_swedish_ci	176 KiB	0 B

c. Membuat aplikasi dengan netbeans



The screenshot shows a Java Swing application window titled 'Form Pengunjung Hotel'. The window has a light blue background and a title bar with standard Windows controls. The form contains several input fields and a large text area. The fields are labeled: Nama, Alamat, No. HP, Kode Kamar, Kelas Kamar, and Kode Transaksi. The 'Kelas Kamar' field is a dropdown menu with the text '-- Kelas Kamar --'. The 'Total' label shows the value '0'. At the bottom of the form, there are six buttons: Save, Reset, Update, Delete, Search, and Exit. A large, empty text area is positioned to the right of the input fields.

Form Pengunjung Hotel

Nama :

Alamat :

No. HP :

Kode Kamar :

Kelas Kamar :

Kode Transaksi :

Total : 0

5. Menerapkan pad aplikasi yang dipilih beberapa fungsi berikut :

a. Insert (Memasukkan data)

```
404 private void saveActionPerformed(java.awt.event.ActionEvent evt) {  
405     String tamu = "INSERT INTO tamu(tamu_nama,alamat,no_hp) VALUE (?,?,?)";  
406     String kamar = "INSERT INTO kamar(kamar_kode,kelas,tarif) VALUE (?,?,?)";  
407     String thk = "INSERT INTO tamu_has_kamar(tamu_nama, kamar_kode) VALUE (?,?)";  
408     String transaksi = "INSERT INTO transaksi(kode_transaksi,kamar_kode,tamu_nama) VALUE (?,?,?)";  
409     String tht = "INSERT INTO tamu_has_transaksi(tamu_nama, kode_transaksi) VALUE (?,?)";  
410  
411     try{  
412         pst = conn.prepareStatement(tamu);  
413         pst.setString(1, nama.getText());  
414         pst.setString(2, alamat.getText());  
415         pst.setString(3, hp.getText());  
416         pst.execute();  
417  
418         pst = conn.prepareStatement(kamar);  
419         pst.setString(1, kode.getText());  
420         String value = kelas.getSelectedItem().toString();  
421         pst.setString(2, value);  
422         pst.setString(3, tarif.getText());  
423         pst.execute();  
424  
425         pst = conn.prepareStatement(thk);  
426         pst.setString(1, nama.getText());  
427         pst.setString(2, kode.getText());  
428         pst.execute();  
429  
430         pst = conn.prepareStatement(transaksi);  
431         pst.setString(1, kode_transaksi.getText());  
432         pst.setString(2, kode.getText());  
433         pst.setString(3, nama.getText());  
434         pst.execute();  
435  
436         pst = conn.prepareStatement(tht);  
437         pst.setString(1, nama.getText());  
438         pst.setString(2, kode_transaksi.getText());  
439         pst.execute();  
440  
441  
442         JOptionPane.showMessageDialog(null, "Berhasil disimpan");  
443     }catch(Exception e){  
444         JOptionPane.showMessageDialog(null, "Gagal menyimpan data");  
445     }viewData();  
446  
447  
448     }
```

— □ ×

Form Pengunjung Hotel

Nama : Budi

Alamat : Bandung

No. HP : 08162777

Kode Kamar : 1012

Kelas Kamar : Double Bed

Kode Transaksi : 1103

Total : 400000

Save

Reset

Update

Delete

Search

Exit

Message

×

i

Berhasil disimpan

OK

— □ ×

Form Pengunjung Hotel

Nama : Budi

Alamat : Bandung

No. HP : 08162777

Kode Kamar : 1012

Kelas Kamar : Double Bed

Kode Transaksi : 1103

Total : 400000

Save

Reset

Update

Delete

Search

Exit

Nama Tamu : Budi

Alamat : Bandung

No. HP : 08162777

Kode Kamar : 1012

Kelas Kamar : Double Bed

Kode Transaksi : 1103

Total Bayar : 400000

b. Update

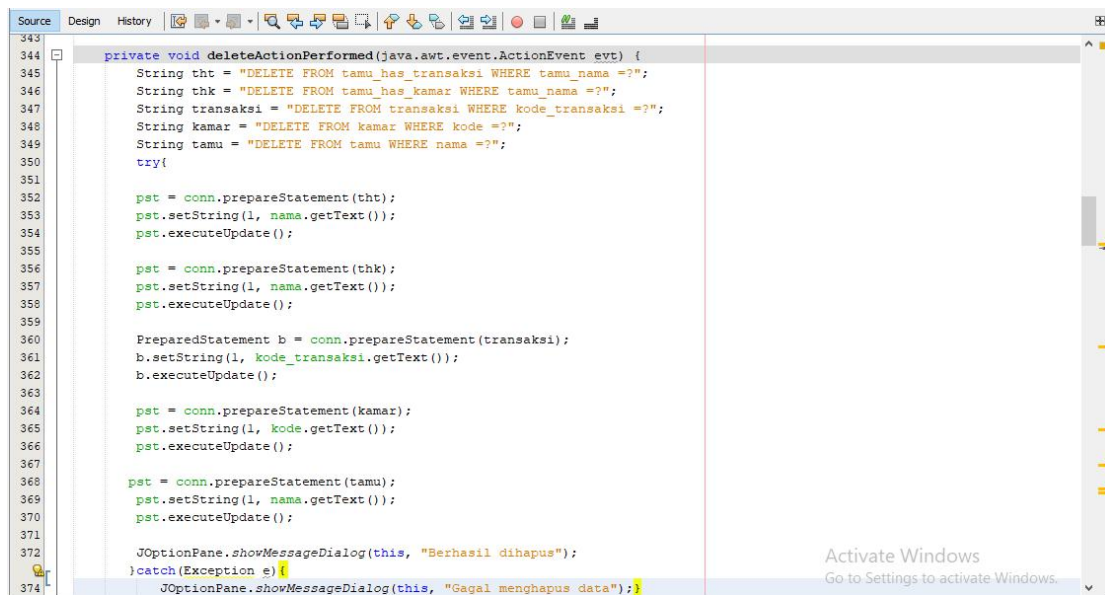
```
458 private void updateActionPerformed(java.awt.event.ActionEvent evt) {
459
460     try{
461         String tht = "UPDATE tamu_has_transaksi set kode_transaksi=? WHERE tamu_nama=?";
462         pst = conn.prepareStatement(tht);
463         pst.setString(1, kode_transaksi.getText());
464         pst.setString(2, nama.getText());
465         pst.executeUpdate();
466
467         String thk = "UPDATE tamu_has_kamar set kamar_kode=? WHERE tamu_nama=?";
468         pst = conn.prepareStatement(thk);
469         pst.setString(1, kode.getText());
470         pst.setString(2, nama.getText());
471         pst.executeUpdate();
472
473         String transaksi = "UPDATE transaksi set kamar_kode=? WHERE kode_transaksi=?";
474         pst = conn.prepareStatement(transaksi);
475         pst.setString(1, kode_transaksi.getText());
476         pst.setString(2, nama.getText());
477         pst.executeUpdate();
478
479         String kamar = "UPDATE kamar set kelas=?, tarif=? WHERE kamar_kode=?";
480         pst = conn.prepareStatement(kamar);
481         pst.setString(1, kode.getText());
482         String v = kelas.getSelectedItem().toString();
483         pst.setString(2, v);
484         pst.setString(3, tarif.getText());
485         pst.executeUpdate();
486
487         String tamu = "UPDATE tamu set alamat=?, no_hp=? WHERE tamu_nama=?";
488         pst = conn.prepareStatement(tamu);
489
490         String tamu = "UPDATE tamu set alamat=?, no_hp=? WHERE tamu_nama=?";
491         pst = conn.prepareStatement(tamu);
492         pst.setString(1, nama.getText());
493         pst.setString(2, alamat.getText());
494         pst.setString(3, hp.getText());
495         pst.executeUpdate();
496
497         JOptionPane.showMessageDialog(this, "Berhasil update");
498     }catch(Exception e){
499         JOptionPane.showMessageDialog(this, "Gagal update data");
500     }viewData();
501 }
```

Activate Windows
Go to Settings to activate Windows.

c. Select

```
501 private void selectActionPerformed(java.awt.event.ActionEvent evt) {
502
503     try{
504         String tamu = "select * from tamu where nama =?";
505         pst = conn.prepareStatement(tamu);
506         pst.setString(1, nama.getText());
507         pst.execute();
508
509         String kamar = "select * from kamar where kode=?";
510         pst = conn.prepareStatement(kamar);
511         pst.setString(1, kode.getText());
512         pst.execute();
513
514         String transaksi = "select * from transaksi where kode_transaksi=?";
515         pst = conn.prepareStatement(transaksi);
516         pst.setString(1, kode_transaksi.getText());
517         pst.execute();
518         JOptionPane.showMessageDialog(this, "Data ditemukan");
519
520     }catch(Exception e){
521         JOptionPane.showMessageDialog(this, "Data tidak tersedia");
522     }
523 }
```

d. Delete



```
343
344 private void deleteActionPerformed(java.awt.event.ActionEvent evt) {
345     String tht = "DELETE FROM tamu_has_transaksi WHERE tamu_nama =?";
346     String thk = "DELETE FROM tamu_has_kamar WHERE tamu_nama =?";
347     String transaksi = "DELETE FROM transaksi WHERE kode_transaksi =?";
348     String kamar = "DELETE FROM kamar WHERE kode =?";
349     String tamu = "DELETE FROM tamu WHERE nama =?";
350     try{
351
352         pst = conn.prepareStatement(tht);
353         pst.setString(1, nama.getText());
354         pst.executeUpdate();
355
356         pst = conn.prepareStatement(thk);
357         pst.setString(1, nama.getText());
358         pst.executeUpdate();
359
360         PreparedStatement b = conn.prepareStatement(transaksi);
361         b.setString(1, kode_transaksi.getText());
362         b.executeUpdate();
363
364         pst = conn.prepareStatement(kamar);
365         pst.setString(1, kode.getText());
366         pst.executeUpdate();
367
368         pst = conn.prepareStatement(tamu);
369         pst.setString(1, nama.getText());
370         pst.executeUpdate();
371
372         JOptionPane.showMessageDialog(this, "Berhasil dihapus");
373     } catch (Exception e) {
374         JOptionPane.showMessageDialog(this, "Gagal menghapus data");
375     }
```

Activate Windows
Go to Settings to activate Windows.