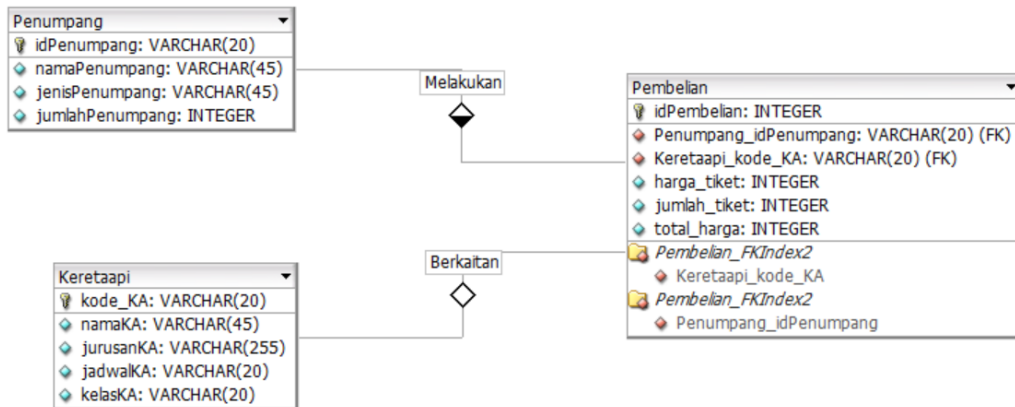


Nama : Nur Fitria Melani
NIM : L200180012
Kelas : A
Mata Kuliah : Praktikum Sistem Basis Data

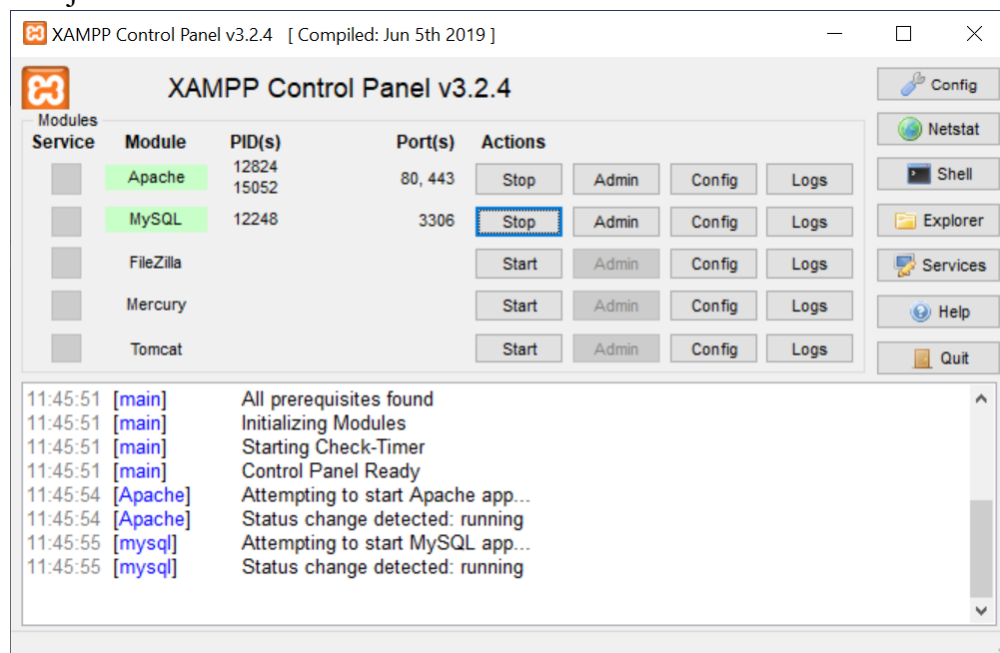
MODUL 12.
APLIKASI DENGAN DATABASE

1. Buatlah sebuah aplikasi(berbasis python atau berbasis java) pada salah satu diantara studi kasus berikut. Saya membuat sebuah aplikasi berbasis java pada studi kasus **Tiket Kereta Api**, dengan minimal entitas yang ada, antara lain:
 - **Kereta Api**
 - **Penumpang**
 - **Pembelian**
2. Penjelasan studi kasus serta ER Diagram
 - **Kereta Api**, mempunyai attribute:
 - kodeKA : kode kereta api
 - namaKA : nama kereta api
 - jurusanKA : jurusan kereta api
 - jadwalKA : jadwal keberangkatan kereta api
 - kelasKA : kelas kereta api
 - **Penumpang**, mempunyai attribute:
 - idPenumpang : id penumpang
 - namaPenumpang : nama penumpang
 - jenisPenumpang : jenis penumpang
 - jumlahPenumpang : jumlah penumpang
 - **Pembelian**, mempunyai attribute:
 - idpemb : id pembelian
 - hargatiket : harga tiket kereta
 - jmltiket : jumlah tiket yang dibeli
 - totalharga : total harga tiket yang dibeli

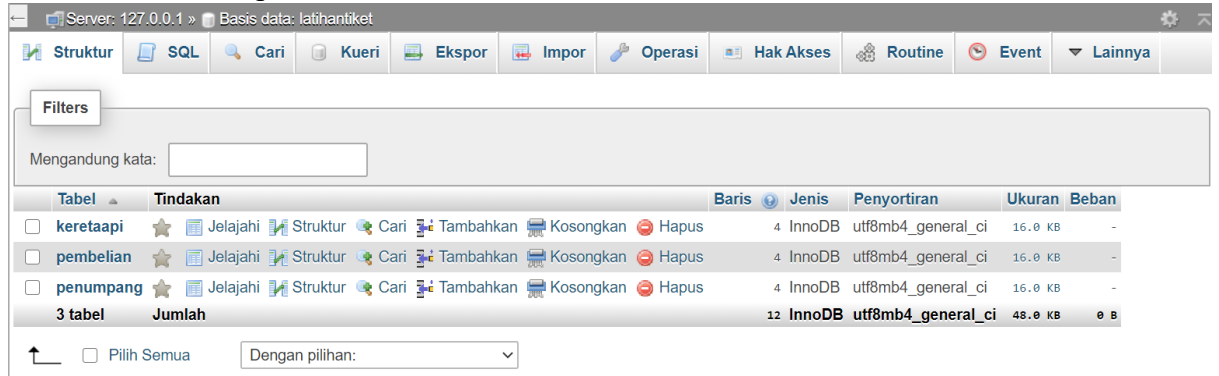


Penumpang dengan Pembelian memiliki hubungan 1:n(one to many), dimana penumpang bisa melakukan banyak pembelian. Sedangkan Kereta api dengan pembelian memiliki hubungan 1:1(one to one).

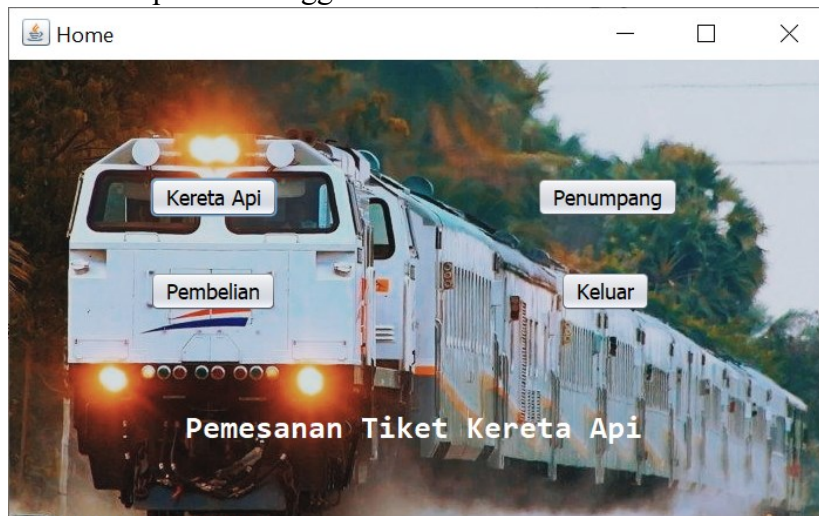
3. Menerapkan beberapa fungsi berikut(view, insert, update, delete) untuk semua entitas:
 - Menjalankan XAMPP Control Panel terlebih dahulu



- Membuat database pada localhost

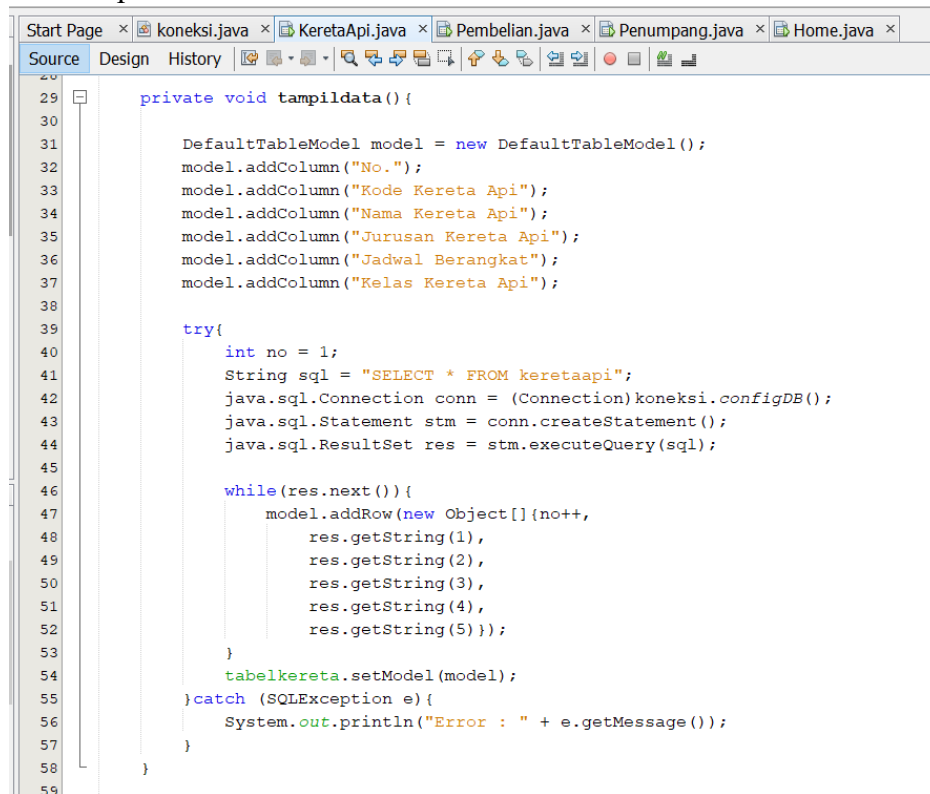


- Membuat aplikasi menggunakan netbeans



- a. Proses VIEW(Select)

- KeretaApi

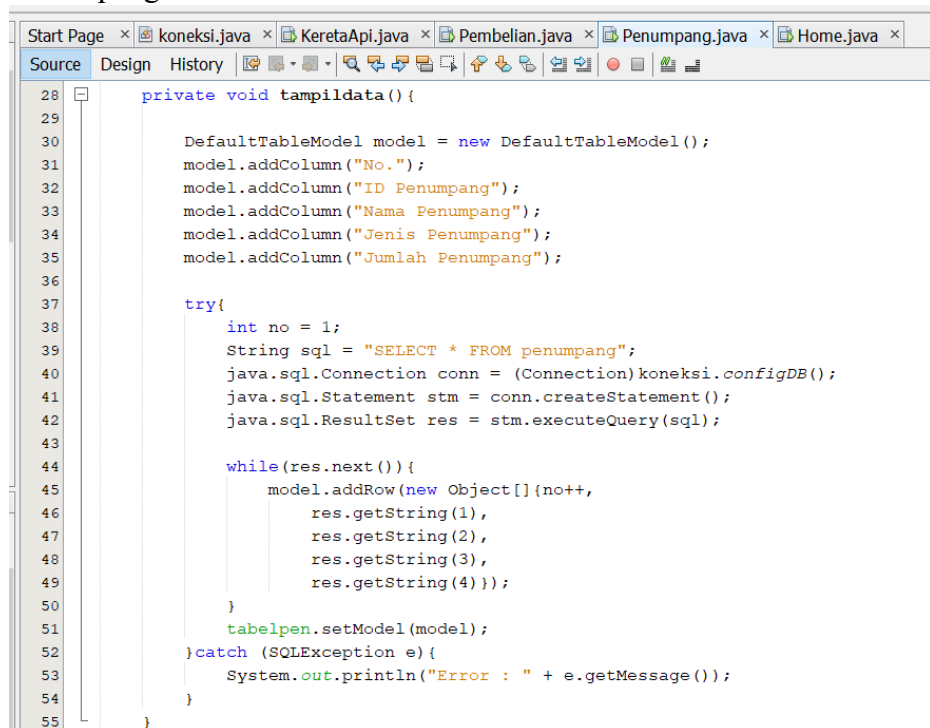


```

28
29     private void tampildata() {
30
31         DefaultTableModel model = new DefaultTableModel();
32         model.addColumn("No.");
33         model.addColumn("Kode Kereta Api");
34         model.addColumn("Nama Kereta Api");
35         model.addColumn("Jurusan Kereta Api");
36         model.addColumn("Jadwal Berangkat");
37         model.addColumn("Kelas Kereta Api");
38
39         try{
40             int no = 1;
41             String sql = "SELECT * FROM keretaapi";
42             java.sql.Connection conn = (Connection)koneksi.configDB();
43             java.sql.Statement stm = conn.createStatement();
44             java.sql.ResultSet res = stm.executeQuery(sql);
45
46             while(res.next()){
47                 model.addRow(new Object[]{no++,
48                     res.getString(1),
49                     res.getString(2),
50                     res.getString(3),
51                     res.getString(4),
52                     res.getString(5)});
53             }
54             tabelkereta.setModel(model);
55         }catch (SQLException e){
56             System.out.println("Error : " + e.getMessage());
57         }
58     }
59

```

- Penumpang



```

28
29     private void tampildata() {
30
31         DefaultTableModel model = new DefaultTableModel();
32         model.addColumn("ID Penumpang");
33         model.addColumn("Nama Penumpang");
34         model.addColumn("Jenis Penumpang");
35         model.addColumn("Jumlah Penumpang");
36
37         try{
38             int no = 1;
39             String sql = "SELECT * FROM penumpang";
40             java.sql.Connection conn = (Connection)koneksi.configDB();
41             java.sql.Statement stm = conn.createStatement();
42             java.sql.ResultSet res = stm.executeQuery(sql);
43
44             while(res.next()){
45                 model.addRow(new Object[]{no++,
46                     res.getString(1),
47                     res.getString(2),
48                     res.getString(3),
49                     res.getString(4)});
50             }
51             tabelpen.setModel(model);
52         }catch (SQLException e){
53             System.out.println("Error : " + e.getMessage());
54         }
55     }
56

```

- Pembelian

```

Start Page x koneksi.java x KeretaApi.java x Pembelian.java x Penumpang.java x Home.java x
Source Design History
59 private void tampildata() {
60
61     DefaultTableModel model = new DefaultTableModel();
62     model.addColumn("No.");
63     model.addColumn("ID Pembelian");
64     model.addColumn("Kode Kereta Api");
65     model.addColumn("ID Penumpang");
66     model.addColumn("Harga Tiket");
67     model.addColumn("Jumlah Tiket");
68     model.addColumn("Total Harga");
69
70     try {
71         int no = 1;
72         String sql = "SELECT * FROM pembelian";
73         java.sql.Connection conn = (Connection) koneksi.configDB();
74         java.sql.Statement stm = conn.createStatement();
75         java.sql.ResultSet res = stm.executeQuery(sql);
76
77         while(res.next()) {
78             model.addRow(new Object[] {no++,
79                 res.getString(1),
80                 res.getString(2),
81                 res.getString(3),
82                 res.getString(4),
83                 res.getString(5),
84                 res.getString(6)});
85         }
86         tabelpemb.setModel(model);
87     } catch (SQLException e) {
88         System.out.println("Error : " + e.getMessage());
89     }
90 }

```


b. INSERT(Memasukkan data)

- KeretaApi

```

Start Page x koneksi.java x KeretaApi.java x Pembelian.java x Penumpang.java x Home.java x
Source Design History
287 private void simpanActionPerformed(java.awt.event.ActionEvent evt) {
288     // TODO add your handling code here:
289     if(txtkodeka.getText().trim().equals("")) {
290         JOptionPane.showMessageDialog(null, "Maaf, kode kereta belum di isi!");
291         txtkodeka.requestFocus();
292     }
293     else {
294         try {
295             String sql = "INSERT INTO keretaapi VALUES ('"+txtkodeka.getText()
296                 + "','" + txtnamaka.getText()
297                 + "','" + cbjrsn.getSelectedItem()
298                 + "','" + txtjdw1.getText()
299                 + "','" + txtklska.getText() + "')";
300             java.sql.Connection conn = (Connection) koneksi.configDB();
301             java.sql.PreparedStatement pstmt = conn.prepareStatement(sql);
302             pstmt.execute();
303             JOptionPane.showMessageDialog(null, "Proses Simpan Data Berhasil!");
304             tampildata();
305             kosongkanform();
306         } catch (HeadlessException | SQLException e) {
307             JOptionPane.showMessageDialog(this, e.getMessage());
308         }
309     }
310 }

```


— □ ×

Detail Kereta Api

Kode Kereta Api

Nama Kereta Api

Jurusan Kereta Api

Jadwal Kereta Api

Kelas Kereta Api

Simpan


Batal

Edit

Hapus

Keluar

No.	Kode Kereta Api	Nama Kereta Api	Jurusan Kereta Api	Jadwal Berangkat	Kelas Kereta Api
1	KA001	Sancaka	Solo Jebres-Ga...	2 Juni 2020, 15.45	Ekonomi
2	KA111	Brantas	Solo Balapan-Pa...	15 Juni 2020, 08....	Ekonomi
3	KA123	Malioboro Express	Solo Balapan-Yo...	21 Juni 2020, 07....	Eksekutif
4	KA192	Bengawan	Solo Balapan-Yo...	29 Juni 2020, 10....	Eksekutif


— □ ×

Detail Kereta Api

Kode Kereta Api

Nama Kereta Api

Jurusan Kereta Api

Jadwal Kereta Api

Kelas Kereta Api

Simpan


Batal

Edit

Hapus

Keluar

Message



Proses Simpan Data Berhasil!

OK

No.	Kode Kereta Api	Nama Kereta Api	Jurusan Kereta Api	Jadwal Berangkat	Kelas Kereta Api
1	KA001	Sancaka	Solo Jebres-Ga...	2 Juni 2020, 15.45	Ekonomi
2	KA111	Brantas	Solo Balapan-Pa...	15 Juni 2020, 08....	Ekonomi
3	KA123	Malioboro Express	Solo Balapan-Yo...	21 Juni 2020, 07....	Eksekutif
4	KA192	Bengawan	Solo Balapan-Yo...	29 Juni 2020, 10....	Eksekutif

—

□

×

Detail Kereta Api

Kode Kereta Api

KA034

Nama Kereta Api

Bengawan

Jurusan Kereta Api

Solo Jebres-Lempuyangan

Jadwal Kereta Api

11 Juni 2020, 14.15

Kelas Kereta Api

Ekonomi

Simpan

Batal

Edit

Hapus

Keluar

No.	Kode Kereta Api	Nama Kereta Api	Jurusan Kereta Api	Jadwal Berangkat	Kelas Kereta Api
1	KA001	Sancaka	Solo Jebres-Ga...	2 Juni 2020, 15.45	Ekonomi
2	KA034	Bengawan	Solo Jebres-Lem...	11 Juni 2020, 14...	Ekonomi
3	KA111	Brantas	Solo Balapan-Pa...	15 Juni 2020, 08...	Ekonomi
4	KA123	Malioboro Express	Solo Balapan-Yo...	21 Juni 2020, 07...	Eksekutif
5	KA192	Bengawan	Solo Balapan-Yo...	29 Juni 2020, 10...	Eksekutif

- Penumpang

Start Page

koneksi.java

KeretaApi.java

Pembelian.java

Penumpang.java

Home.java

Source

Design

History

```

268 private void simpanActionPerformed(java.awt.event.ActionEvent evt) {
269     // TODO add your handling code here:
270     if(txtIdpenump.getText().trim().equals("")){
271         JOptionPane.showMessageDialog(null, "Maaf, ID Penumpang belum di isi!");
272         txtIdpenump.requestFocus();
273     }
274     else if(txtnama.getText().trim().equals("")){
275         JOptionPane.showMessageDialog(null, "Maaf, Nama Penumpang belum di isi!");
276         txtnama.requestFocus();
277     }
278     else if(txtjmlpenump.getText().trim().equals("")){
279         JOptionPane.showMessageDialog(null, "Maaf, Jumlah Penumpang belum di isi!");
280         txtjmlpenump.requestFocus();
281     }
282     else{
283         try{
284             String sql = "INSERT INTO penumpang VALUES ('"+txtIdpenump.getText()
285                 + "','" +txtnama.getText()
286                 + "','" +cbjenis.getSelectedItem()
287                 + "','" +txtjmlpenump.getText()+"')";
288             java.sql.Connection conn = (Connection)koneksi.configDB();
289             java.sql.PreparedStatement pstmt = conn.prepareStatement(sql);
290             pstmt.execute();
291             JOptionPane.showMessageDialog(null, "Proses Simpan Data Berhasil!");
292             tampildata();
293             kosongkanform();
294         }catch (HeadlessException | SQLException e){
295             JOptionPane.showMessageDialog(this, e.getMessage());
296         }
297     }
298 }
299

```

Detail Penumpang

ID Penumpang

P005

Nama Penumpang

Sania Elvin

Jenis Penumpang

Dewasa

Jumlah Penumpang

1

Simpan

Batal

Edit

Hapus

Keluar

No.	ID Penumpang	Nama Penumpang	Jenis Penumpang	Jumlah Penumpang
1	P001	Naya Bilqis	Dewasa	1
2	P002	Jeffri Adam	Anak-anak	1
3	P003	Ilham Saputra	Dewasa	1
4	P004	Muhammad Tyo	Dewasa	1

Detail Penumpang

ID Penumpang

P005

Nama Penumpang

Sania Elvin

Jenis Penumpang

Dewasa

Jumlah Penumpang

1

Simpan

Batal

OK

Keluar

No.	ID Penumpang	Nama Penumpang	Jenis Penumpang	Jumlah Penumpang
1	P001	Naya Bilqis	Dewasa	1
2	P002	Jeffri Adam	Anak-anak	1
3	P003	Ilham Saputra	Dewasa	1
4	P004	Muhammad Tyo	Dewasa	1

Message

Proses Simpan Data Berhasil!

OK

Detail Penumpang

ID Penumpang

Nama Penumpang

Jenis Penumpang

--Pilih--

Jumlah Penumpang

Simpan

Batal

Edit

Hapus

Keluar

No.	ID Penumpang	Nama Penumpang	Jenis Penumpang	Jumlah Penumpang
1	P001	Naya Bilqis	Dewasa	1
2	P002	Jeffri Adam	Anak-anak	1
3	P003	Ilham Saputra	Dewasa	1
4	P004	Muhammad Tyo	Dewasa	1
5	P005	Sania Elvin	Dewasa	1

- Pembelian

```

249
250 private void simpanActionPerformed(java.awt.event.ActionEvent evt) {
251     // TODO add your handling code here:
252     if(txtidpemb.getText().trim().equals("")){
253         JOptionPane.showMessageDialog(null, "Maaf, ID Pembelian belum di isi!");
254         txtidpemb.requestFocus();
255     }
256     //else if(txtharga.getText().trim().equals("")){
257     //     JOptionPane.showMessageDialog(null, "Maaf, Harga Tiket belum di isi!");
258     //     txtharga.requestFocus();
259     //}
260     else{
261         try{
262             String sql = "INSERT INTO pembelian VALUES ('"+txtidpemb.getText()
263                 + "','" +cbkodeka.getSelectedItemAt()
264                 + "','" +cbidpenump.getSelectedItemAt()
265                 + "','" +txtharga.getText()
266                 + "','" +txtjml.getText()
267                 + "','" +txttotal.getText()+"')";
268             java.sql.Connection conn = (Connection)koneksi.configDB();
269             java.sql.PreparedStatement pstmt = conn.prepareStatement(sql);
270             pstmt.execute();
271             JOptionPane.showMessageDialog(null, "Proses Simpan Data Berhasil!");
272             tampildata();
273             kosongkanform();
274         }catch(HeadlessException | SQLException e){
275             JOptionPane.showMessageDialog(this, e.getMessage());
276         }
277     }
278 }

```

— □ ×

Detail Pembelian

ID Pembelian

Kode Kereta Api

KA111 ▾

ID Penumpang

P002 ▾

Harga Tiket

Jumlah Tiket

Total Harga

Simpan

Batal

Edit

Hapus

Keluar

No.	ID Pembelian	Kode Kereta Api	ID Penumpang	Harga Tiket	Jumlah Tiket	Total Harga
1	101	KA001	P001	110000	1	110000
2	102	KA111	P002	89000	1	89000
3	103	KA123	P003	91000	2	182000
4	104	KA192	P004	125000	1	125000

—

□

×

Detail Pembelian

ID Pembelian

105

Kode Kereta Api

KA111

ID Penumpang

P002

Harga Tiket

89000

Jumlah Tiket

2

Total Harga

178000

Simpan

Batal

Edit

Hapus

Keluar

No.	ID Pembelian	Kode Kereta Api	ID Penumpang	Harga Tiket	Jumlah Tiket	Total Harga
1	101	KA001	P001	110000	1	110000
2	102	KA111	P002	89000	1	89000
3	103	KA123	P003	91000	2	182000
4	104	KA192	P004	125000	1	125000

Message

Proses Simpan Data Berhasil!

OK

—

□

×

Detail Pembelian

ID Pembelian

Kode Kereta Api

--Pilih--

ID Penumpang

--Pilih--

Harga Tiket

Jumlah Tiket

Total Harga

Simpan

Batal

Edit

Hapus

Keluar

No.	ID Pembelian	Kode Kereta Api	ID Penumpang	Harga Tiket	Jumlah Tiket	Total Harga
1	101	KA001	P001	110000	1	110000
2	102	KA111	P002	89000	1	89000
3	103	KA123	P003	91000	2	182000
4	104	KA192	P004	125000	1	125000
5	105	KA111	P002	89000	2	178000

c. UPDATE(Mengubah data)

- KeretaApi

The screenshot displays a Java IDE with the `editActionPerformed` method in `KeretaApi.java`. The method handles an `ActionEvent` by constructing an SQL `UPDATE` statement for the `keretaapi` table. It uses text fields for `kodeKA`, `namaKA`, `jurusanKA`, `jadwalKA`, and `kelasKA`. The code connects to a database, executes the update, and shows a success message. Below the code, a Java Swing window titled "Detail Kereta Api" is shown. It contains a form with the same fields as the code, a table of train data, and buttons for `Simpan`, `Batal`, `Edit`, `Hapus`, and `Keluar`.

```
332 private void editActionPerformed(java.awt.event.ActionEvent evt) {
333     // TODO add your handling code here:
334     try{
335         String sql = "UPDATE keretaapi SET kodeKA='"+txtkodeka.getText()
336             +"',namaKA='"+txtnamaka.getText()
337             +"',jurusanKA='"+cbjrsn.getSelectedItem()
338             +"',jadwalKA='"+txtjdw1.getText()
339             +"',kelasKA='"+txtklska.getText()
340             +"' WHERE kodeKA='"+txtkodeka.getText()+"'";
341         java.sql.Connection conn = (Connection)koneksi.configDB();
342         java.sql.PreparedStatement pstmt = conn.prepareStatement(sql);
343         pstmt.execute();
344         JOptionPane.showMessageDialog(null, "Edit Data Berhasil!");
345     }catch(HeadlessException | SQLException e){
346         JOptionPane.showMessageDialog(this, e.getMessage());
347     }
348     tampildata();
349     kosongkanForm();
350 }
```

Detail Kereta Api

Kode Kereta Api: KA192

Nama Kereta Api: Prambanan Express

Jurusan Kereta Api: Solo Balapan-Yogyakarta

Jadwal Kereta Api: 29 Juni 2020, 10.00

Kelas Kereta Api: Eksekutif

Simpan Batal Edit Hapus Keluar

No.	Kode Kereta Api	Nama Kereta Api	Jurusan Kereta Api	Jadwal Berangkat	Kelas Kereta Api
1	KA001	Sancaka	Solo Jebres-Ga...	2 Juni 2020, 15.45	Ekonomi
2	KA034	Bengawan	Solo Jebres-Lem...	11 Juni 2020, 14....	Ekonomi
3	KA111	Brantas	Solo Balapan-Pa...	15 Juni 2020, 08....	Ekonomi
4	KA123	Malioboro Express	Solo Balapan-Yo...	21 Juni 2020, 07....	Eksekutif
5	KA192	Bengawan	Solo Balapan-Yo...	29 Juni 2020, 10....	Eksekutif

Detail Kereta Api

Kode Kereta Api

KA192

Nama Kereta Api

Prambanan Express

Jurusan Kereta Api

Solo Balapan-Yogyakarta

Jadwal Kereta Api

29 Juni 2020, 10:00

Kelas Kereta Api

Eksekutif

Simpan

Batal

Edit

Hapus

Keluar

Message

Edit Data Berhasil!

OK

No.	Kode Kereta Api	Nama Kereta Api	Jurusan Kereta Api	Jadwal Berangkat	Kelas Kereta Api
1	KA001	Sancaka	Solo Jebres-Ga...	2 Juni 2020, 15.45	Ekonomi
2	KA034	Bengawan	Solo Jebres-Lem...	11 Juni 2020, 14...	Ekonomi
3	KA111	Brantas	Solo Balapan-Pa...	15 Juni 2020, 08...	Ekonomi
4	KA123	Malioboro Express	Solo Balapan-Yo...	21 Juni 2020, 07...	Eksekutif
5	KA192	Bengawan	Solo Balapan-Yo...	29 Juni 2020, 10...	Eksekutif

Detail Kereta Api

Kode Kereta Api

Nama Kereta Api

Jurusan Kereta Api

--Pilih Jurusan--

Jadwal Kereta Api

Kelas Kereta Api

Simpan

Batal

Edit

Hapus

Keluar

No.	Kode Kereta Api	Nama Kereta Api	Jurusan Kereta Api	Jadwal Berangkat	Kelas Kereta Api
1	KA001	Sancaka	Solo Jebres-Ga...	2 Juni 2020, 15.45	Ekonomi
2	KA034	Bengawan	Solo Jebres-Lem...	11 Juni 2020, 14...	Ekonomi
3	KA111	Brantas	Solo Balapan-Pa...	15 Juni 2020, 08...	Ekonomi
4	KA123	Malioboro Express	Solo Balapan-Yo...	21 Juni 2020, 07...	Eksekutif
5	KA192	Prambanan Expr...	Solo Balapan-Yo...	29 Juni 2020, 10...	Eksekutif

- Penumpang

Start Page

koneksi.java

KeretaApi.java

Pembelian.java

Penumpang.java

Home.java

Source

Design

History

```

317 private void editActionPerformed(java.awt.event.ActionEvent evt) {
318     // TODO add your handling code here:
319     try{
320         String sql = "UPDATE penumpang SET idPenumpang='"+txtIdpenump.getText()
321             +"',namaPenumpang='"+txtnama.getText()
322             +"',jenisPenumpang='"+cbjenis.getSelectedItem()
323             +"',jumlahPenumpang='"+txtjmlpenump.getText()
324             +"' WHERE idPenumpang='"+txtIdpenump.getText()+"'";
325         java.sql.Connection conn = (Connection)koneksi.configDB();
326         java.sql.PreparedStatement pstmt = conn.prepareStatement(sql);
327         pstmt.execute();
328         JOptionPane.showMessageDialog(null, "Edit Data Berhasil!");
329     }catch(HeadlessException | SQLException e){
330         JOptionPane.showMessageDialog(this, e.getMessage());
331     }
332     tampildata();
333     kosongkanform();
334 }

```

Detail Penumpang

ID Penumpang

P005

Nama Penumpang

Sania Elvin

Jenis Penumpang

Dewasa

Jumlah Penumpang

2

Simpan

Batal

Edit

Hapus

Keluar

No.	ID Penumpang	Nama Penumpang	Jenis Penumpang	Jumlah Penumpang
1	P001	Naya Bilqis	Dewasa	1
2	P002	Jefri Adam	Anak-anak	1
3	P003	Ilham Saputra	Dewasa	1
4	P004	Muhammad Tyo	Dewasa	1
5	P005	Sania Elvin	Dewasa	1

Detail Penumpang

ID Penumpang

P005

Nama Penumpang

Sania Elvin

Jenis Penumpang

Dewasa

Jumlah Penumpang

2

Simpan

Batal

Keluar

Message

Edit Data Berhasil!

OK

No.	ID Penumpang	Nama Penumpang	Jenis Penumpang	Jumlah Penumpang
1	P001	Naya Bilqis	Dewasa	1
2	P002	Jeffri Adam	Anak-anak	1
3	P003	Ilham Saputra	Dewasa	1
4	P004	Muhammad Tyo	Dewasa	1
5	P005	Sania Elvin	Dewasa	1

Detail Penumpang

ID Penumpang

Nama Penumpang

Jenis Penumpang

--Pilih--

Jumlah Penumpang

Simpan

Batal

Edit

Hapus

Keluar

No.	ID Penumpang	Nama Penumpang	Jenis Penumpang	Jumlah Penumpang
1	P001	Naya Bilqis	Dewasa	1
2	P002	Jeffri Adam	Anak-anak	1
3	P003	Ilham Saputra	Dewasa	1
4	P004	Muhammad Tyo	Dewasa	1
5	P005	Sania Elvin	Dewasa	2

- Pembelian

```

284
285 private void editActionPerformed(java.awt.event.ActionEvent evt) {
286     // TODO add your handling code here:
287     try{
288         String sql = "UPDATE pembelian SET idPemb='"+txtidpemb.getText()
289             +"',kodeKA='"+cbkodeka.getSelectedItem()
290             +"',idPenumpang='"+cbidpenump.getSelectedItem()
291             +"',hargatiket='"+txtharga.getText()
292             +"',jmltiket='"+txtjml.getText()
293             +"',totalharga='"+txttotal.getText()
294             +"' WHERE idPemb='"+txtidpemb.getText()+"'";
295         java.sql.Connection conn = (Connection)koneksi.configDB();
296         java.sql.PreparedStatement pstm = conn.prepareStatement(sql);
297         pstm.execute();
298         JOptionPane.showMessageDialog(null, "Edit Data Berhasil!");
299     }catch(HeadlessException | SQLException e){
300         JOptionPane.showMessageDialog(this, e.getMessage());
301     }
302     tampildata();
303     kosongkanform();
304 }

```

— □ ×

Detail Pembelian

ID Pembelian

Kode Kereta Api

KA111 ▾

ID Penumpang

P002 ▾

Harga Tiket

Jumlah Tiket

Total Harga

Simpan

Batal

Edit

Hapus

Keluar

No.	ID Pembelian	Kode Kereta Api	ID Penumpang	Harga Tiket	Jumlah Tiket	Total Harga
1	101	KA001	P001	110000	1	110000
2	102	KA111	P002	89000	1	89000
3	103	KA123	P003	91000	2	182000
4	104	KA192	P004	125000	1	125000
5	105	KA111	P002	86000	2	172000

Detail Pembelian

ID Pembelian

105

Kode Kereta Api

KA111

ID Penumpang

P002

Harga Tiket

86000

Jumlah Tiket

2

Total Harga

172000

Message

Edit Data Berhasil!

OK

Simpan

Batal

Edit

Hapus

Keluar

No.	ID Pembelian	Kode Kereta Api	ID Penumpang	Harga Tiket	Jumlah Tiket	Total Harga
1	101	KA001	P001	110000	1	110000
2	102	KA111	P002	89000	1	89000
3	103	KA123	P003	91000	2	182000
4	104	KA192	P004	125000	1	125000
5	105	KA111	P002	89000	2	178000

Detail Pembelian

ID Pembelian

Kode Kereta Api

--Pilih--

ID Penumpang

--Pilih--

Harga Tiket

Jumlah Tiket

Total Harga

Simpan

Batal

Edit

Hapus

Keluar

No.	ID Pembelian	Kode Kereta Api	ID Penumpang	Harga Tiket	Jumlah Tiket	Total Harga
1	101	KA001	P001	110000	1	110000
2	102	KA111	P002	89000	1	89000
3	103	KA123	P003	91000	2	182000
4	104	KA192	P004	125000	1	125000
5	105	KA111	P002	86000	2	172000

d. DELETE(Menghapus data)

- KeretaApi

The top part of the image shows a Java IDE with the file `KeretaApi.java` open. The code defines the `deleteActionPerformed` method, which executes a SQL DELETE statement to remove a train record based on its code. It also displays a success message and refreshes the data table.

```
private void hapusActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    try{  
        String sql = "DELETE FROM keretaapi WHERE kodeKA='"+txtkodeka.getText()+"'";  
        java.sql.Connection conn = (Connection)koneksi.configDB();  
        java.sql.PreparedStatement pstm = conn.prepareStatement(sql);  
        pstm.execute();  
        JOptionPane.showMessageDialog(null, "Hapus Data Berhasil!");  
    }catch(HeadlessException | SQLException e){  
        JOptionPane.showMessageDialog(this, e.getMessage());  
    }  
    tampilkanData();  
    kosongkanForm();  
}
```

The bottom part of the image shows the 'Detail Kereta Api' application window. It contains a form with fields for Kode Kereta Api, Nama Kereta Api, Jurusan Kereta Api, Jadwal Kereta Api, and Kelas Kereta Api. Below the form are buttons for Simpan, Batal, Edit, Hapus, and Keluar. At the bottom, there is a table displaying the list of train data.

No.	Kode Kereta Api	Nama Kereta Api	Jurusan Kereta Api	Jadwal Berangkat	Kelas Kereta Api
1	KA001	Sancaka	Solo Jebres-Ga...	2 Juni 2020, 15.45	Ekonomi
2	KA034	Bengawan	Solo Jebres-Lem...	11 Juni 2020, 14...	Ekonomi
3	KA111	Brantas	Solo Balapan-Pa...	15 Juni 2020, 08...	Ekonomi
4	KA123	Maliboro Express	Solo Balapan-Yo...	21 Juni 2020, 07...	Eksekutif
5	KA192	Prambanan Expr...	Solo Balapan-Yo...	29 Juni 2020, 10...	Eksekutif

Detail Kereta Api

Kode Kereta Api:

Nama Kereta Api:

Jurusan Kereta Api:

Jadwal Kereta Api:

Kelas Kereta Api:

No.	Kode Kereta Api	Nama Kereta Api	Jurusan Kereta Api	Jadwal Berangkat	Kelas Kereta Api
1	KA001	Sancaka	Solo Jebres-Ga...	2 Juni 2020, 15.45	Ekonomi
2	KA034	Bengawan	Solo Jebres-Lempuyangan	11 Juni 2020, 14.15	Ekonomi
3	KA111	Brantas	Solo Balapan-Pa...	15 Juni 2020, 08.00	Ekonomi
4	KA123	Malioboro Express	Solo Balapan-Yo...	21 Juni 2020, 07.00	Eksekutif
5	KA192	Prambanan Expr...	Solo Balapan-Yo...	29 Juni 2020, 10.00	Eksekutif

Message

Hapus Data Berhasil!!

Detail Kereta Api

Kode Kereta Api:

Nama Kereta Api:

Jurusan Kereta Api:

Jadwal Kereta Api:

Kelas Kereta Api:

No.	Kode Kereta Api	Nama Kereta Api	Jurusan Kereta Api	Jadwal Berangkat	Kelas Kereta Api
1	KA001	Sancaka	Solo Jebres-Ga...	2 Juni 2020, 15.45	Ekonomi
2	KA111	Brantas	Solo Balapan-Pa...	15 Juni 2020, 08.00	Ekonomi
3	KA123	Malioboro Express	Solo Balapan-Yo...	21 Juni 2020, 07.00	Eksekutif
4	KA192	Prambanan Expr...	Solo Balapan-Yo...	29 Juni 2020, 10.00	Eksekutif

- Penumpang

Start Page

koneksi.java

KeretaApi.java

Pembelian.java

Penumpang.java

Home.java

SourceDesignHistory

```

336 private void hapusActionPerformed(java.awt.event.ActionEvent evt) {
337     // TODO add your handling code here:
338     try{
339         String sql = "DELETE FROM penumpang WHERE idPenumpang='"+txtIdpenump.getText()+"'";
340         java.sql.Connection conn = (Connection)koneksi.configDB();
341         java.sql.PreparedStatement pstmt = conn.prepareStatement(sql);
342         pstmt.execute();
343         JOptionPane.showMessageDialog(null, "Hapus Data Berhasil!");
344     }catch(HeadlessException | SQLException e){
345         JOptionPane.showMessageDialog(this, e.getMessage());
346     }
347     tampildata();
348     kosongkanform();
349 }

```

Detail Penumpang

ID Penumpang

P005

Nama Penumpang

Sania Elvin

Jenis Penumpang

Dewasa

Jumlah Penumpang

2

Simpan


Batal

Edit

Hapus

Keluar

No.	ID Penumpang	Nama Penumpang	Jenis Penumpang	Jumlah Penumpang
1	P001	Naya Bilqis	Dewasa	1
2	P002	Jefri Adam	Anak-anak	1
3	P003	Ilham Saputra	Dewasa	1
4	P004	Muhammad Tyo	Dewasa	1
5	P005	Sania Elvin	Dewasa	2



—

□

✕

Detail Penumpang

ID Penumpang

Nama Penumpang

Jenis Penumpang

--Pilih--

Jumlah Penumpang

Simpan

Batal

Edit

Hapus

Keluar

No.	ID Penumpang	Nama Penumpang	Jenis Penumpang	Jumlah Penumpang
1	P001	Naya Bilqis	Dewasa	1
2	P002	Jefri Adam	Anak-anak	1
3	P003	Ilham Saputra	Dewasa	1
4	P004	Muhammad Tyo	Dewasa	1

- Pembelian

Start Page

koneksi.java

KeretaApi.java

Pembelian.java

Penumpang.java

Home.java

Source

Design

History

```

306     private void hapusActionPerformed(java.awt.event.ActionEvent evt) {
307         // TODO add your handling code here:
308         try{
309             String sql = "DELETE FROM pembelian WHERE idpemb='"+txtidpemb.getText()+"'";
310             java.sql.Connection conn = (Connection)koneksi.configDB();
311             java.sql.PreparedStatement pstmt = conn.prepareStatement(sql);
312             pstmt.execute();
313             JOptionPane.showMessageDialog(null, "Hapus Data Berhasil!");
314         }catch(HeadlessException | SQLException e){
315             JOptionPane.showMessageDialog(this, e.getMessage());
316         }
317         tampilkan();
318         kosongkanform();
319     }

```

Detail Pembelian

ID Pembelian

105

Kode Kereta Api

KA111

ID Penumpang

P002

Harga Tiket

86000

Jumlah Tiket

2

Total Harga

172000

Simpan


Batal

Edit

Hapus

Keluar

No.	ID Pembelian	Kode Kereta Api	ID Penumpang	Harga Tiket	Jumlah Tiket	Total Harga
1	101	KA001	P001	110000	1	110000
2	102	KA111	P002	89000	1	89000
3	103	KA123	P003	91000	2	182000
4	104	KA192	P004	125000	1	125000
5	105	KA111	P002	86000	2	172000



Detail Pembelian

ID Pembelian

105

Kode Kereta Api

KA111

ID Penumpang

P001

Harga Tiket

86000

Jumlah Tiket

2

Total Harga

172000

Simpan


Batal

Edit

Hapus

Keluar

No.	ID Pembelian	Kode Kereta Api	ID Penumpang	Harga Tiket	Jumlah Tiket	Total Harga
1	101	KA001	P001	110000	1	110000
2	102	KA111	P002	89000	1	89000
3	103	KA123	P003	91000	2	182000
4	104	KA192	P004	125000	1	125000
5	105	KA111	P002	86000	2	172000


— □ ×

Detail Pembelian

ID Pembelian

Kode Kereta Api

--Pilih-- ▼

ID Penumpang

--Pilih-- ▼

Harga Tiket

Jumlah Tiket

Total Harga

Simpan

Batal

Edit

Hapus

Keluar

No.	ID Pembelian	Kode Kereta Api	ID Penumpang	Harga Tiket	Jumlah Tiket	Total Harga
1	101	KA001	P001	110000	1	110000
2	102	KA111	P002	89000	1	89000
3	103	KA123	P003	91000	2	182000
4	104	KA192	P004	125000	1	125000