

Nama: Mochammad Khalish Mulyadi

NPM: 14118187

Kelas: 3KA01

ACT 2 Java for Intermediate

1. Membuat database dan tabel

```
XAMPP for Windows - mysql -u root
Khalish@ATTRIXIASDEVICE c:\xampp
# mysql -u root
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 108
Server version: 10.4.14-MariaDB mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> CREATE DATABASE sewa_ruang_khalish;
Query OK, 1 row affected (0.003 sec)

MariaDB [(none)]> USE sewa_ruang_khalish;
Database changed
MariaDB [sewa_ruang_khalish]> CREATE TABLE ruangan(
  -> id_ruang int(5) PRIMARY KEY,
  -> jenis_ruang VARCHAR(10) NOT NULL);
Query OK, 0 rows affected (0.292 sec)

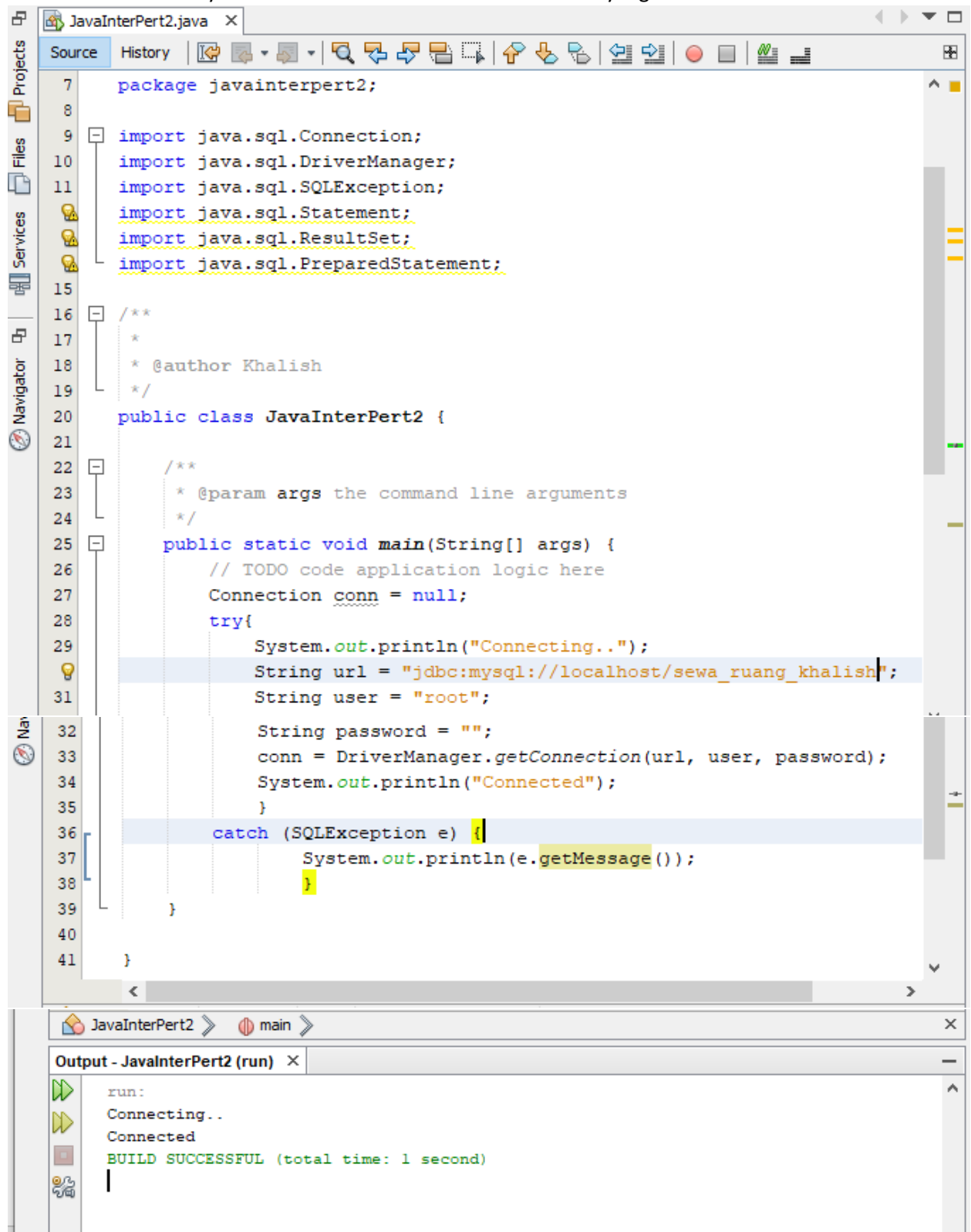
MariaDB [sewa_ruang_khalish]> DESC ruangan;
+-----+-----+-----+-----+-----+-----+
| Field      | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id_ruang   | int(5)    | NO   | PRI | NULL    |       |
| jenis_ruang | varchar(10) | NO   |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.029 sec)

MariaDB [sewa_ruang_khalish]> ALTER TABLE ruangan ADD harga_sewa FLOAT(5);
Query OK, 0 rows affected (0.133 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [sewa_ruang_khalish]> DESC ruangan;
+-----+-----+-----+-----+-----+-----+
| Field      | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id_ruang   | int(5)    | NO   | PRI | NULL    |       |
| jenis_ruang | varchar(10) | NO   |     | NULL    |       |
| harga_sewa | float     | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.005 sec)

MariaDB [sewa_ruang_khalish]>
```

2. Test koneksi. Disini saya mencoba untuk tes koneksi ke database yang telah dibuat



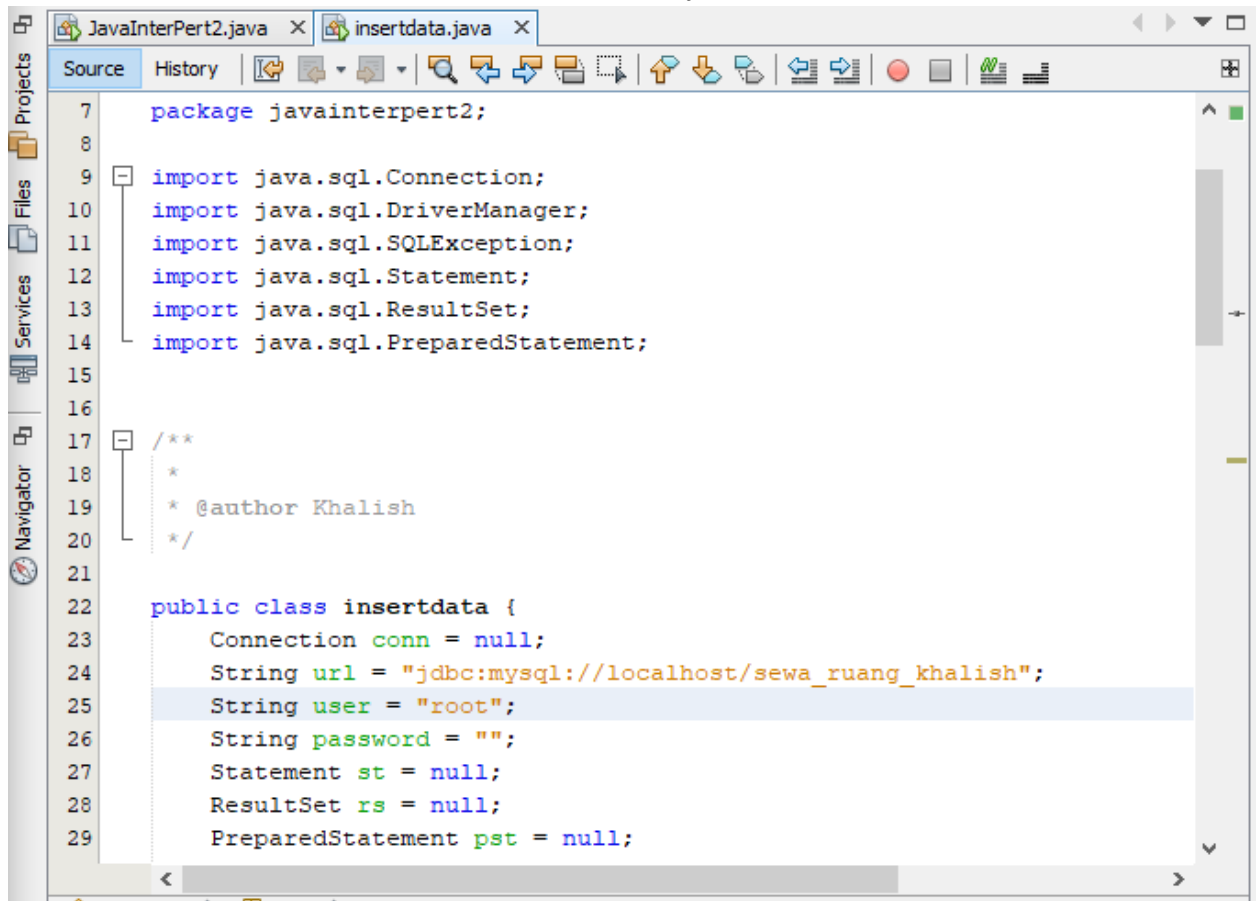
The screenshot displays an IDE window titled 'JavaInterPert2.java'. The code defines a package 'javainterpert2' and imports several JDBC classes: 'Connection', 'DriverManager', 'SQLException', 'Statement', 'ResultSet', and 'PreparedStatement'. A class 'JavaInterPert2' is defined with a 'main' method. The 'main' method attempts to establish a database connection using 'DriverManager.getConnection' with the URL 'jdbc:mysql://localhost/sewa\_ruang\_khalish', user 'root', and an empty password. It prints 'Connecting..' before the connection attempt and 'Connected' upon success. A 'catch' block for 'SQLException' prints the error message. The IDE's 'Output' window at the bottom shows the execution results: 'Connecting..', 'Connected', and 'BUILD SUCCESSFUL (total time: 1 second)'.

```
7 package javainterpert2;
8
9 import java.sql.Connection;
10 import java.sql.DriverManager;
11 import java.sql.SQLException;
12 import java.sql.Statement;
13 import java.sql.ResultSet;
14 import java.sql.PreparedStatement;
15
16 /**
17  *
18  * @author Khalish
19  */
20 public class JavaInterPert2 {
21
22     /**
23      * @param args the command line arguments
24      */
25     public static void main(String[] args) {
26         // TODO code application logic here
27         Connection conn = null;
28         try{
29             System.out.println("Connecting..");
30             String url = "jdbc:mysql://localhost/sewa_ruang_khalish";
31             String user = "root";
32
33             String password = "";
34             conn = DriverManager.getConnection(url, user, password);
35             System.out.println("Connected");
36         }
37         catch (SQLException e) {
38             System.out.println(e.getMessage());
39         }
40     }
41 }
```

Output - JavaInterPert2 (run) x

```
run:
Connecting..
Connected
BUILD SUCCESSFUL (total time: 1 second)
```

3. Membuat DML(Data Manipulation Language) Disini saya menggunakan query dml yaitu insert into untuk memasukkan data ke dalam database melalui java.



```
7 package javainterpert2;
8
9 import java.sql.Connection;
10 import java.sql.DriverManager;
11 import java.sql.SQLException;
12 import java.sql.Statement;
13 import java.sql.ResultSet;
14 import java.sql.PreparedStatement;
15
16
17 /**
18  *
19  * @author Khalish
20  */
21
22 public class insertdata {
23     Connection conn = null;
24     String url = "jdbc:mysql://localhost/sewa_ruang_khalish";
25     String user = "root";
26     String password = "";
27     Statement st = null;
28     ResultSet rs = null;
29     PreparedStatement pst = null;
```

```
JavaInterPert2.java x insertdata.java x
Source History
31 public static void main(String[] args) {
32     insertdata app = new insertdata();
33
34     app.masukanData();
35     app.showData(); }
36 public void masukanData() {
37     System.out.println("***MASUKKAN DATA**");
38     try {
39         conn = DriverManager.getConnection(url, user, password);
40         String query = "INSERT INTO ruangan(id_ruang,jenis_ruang,ha
41         pst = conn.prepareStatement(query);
42         for (int i = 1; i <= 10; i++) {
43             pst.setInt(1, i);
44             pst.setString(2, "ruangan-" + i);
45             pst.setFloat(3, new Float(5.4 * i));
46             pst.executeUpdate(); }
47         System.out.println("=====");
48         System.out.println("***BERHASIL MASUKKAN DATA**");
49     } catch (SQLException e) { System.out.println(e.getMessage());
50     } finally {
51         System.out.println("=====");
52     }
```

```
JavaInterPert2.java x insertdata.java x
Source History
53     try {
54         if (pst != null) { pst.close(); }
55         if (conn != null) { conn.close(); }
56     } catch (SQLException e) { System.out.println(e.getMessage());
57     }
58
59 public void showData() {
60     System.out.println("***TAMPILKAN DATA**");
61     System.out.println("=====");
62     System.out.println("id_ruang"+"\\t"+"jenis_ruang"+"\\t"+"\\t"+"\\t"
63     try {
64         conn = DriverManager.getConnection(url, user, password);
65         pst = conn.prepareStatement("SELECT * FROM ruangan");
66         rs = pst.executeQuery();
67         while (rs.next()) {
68             System.out.print(rs.getInt("id_ruang"));
69             System.out.print("\\t"+"\\t");
70             System.out.print(rs.getString("jenis_ruang"));
71             System.out.print("\\t"+"\\t");
72             System.out.println(rs.getFloat("harga_sewa")); }
73     } catch (SQLException e) { System.out.println(e.getMessage());
74     } finally {
75         System.out.println("=====");
```

```
76         try {
77             if (rs != null) { rs.close(); }
78             if (pst != null) { pst.close(); }
79             if (conn != null) { conn.close(); }
80         } catch (SQLException e) { System.out.println(e.getMessage()); }
81     }
82 }
83 }
84
85
```

insertdata > user

Output - JavaInterPert2 (run) X

```
run:
**MASUKKAN DATA**
=====
**BERHASIL MASUKKAN DATA**
=====
**TAMPILKAN DATA**
=====
id_ruang      jenis_ruang      harga_sewa
1             ruangan-1        5.4
2             ruangan-2        10.8
3             ruangan-3        16.2
4             ruangan-4        21.6
5             ruangan-5        27.0
6             ruangan-6        32.4
7             ruangan-7        37.8
8             ruangan-8        43.2
9             ruangan-9        48.6
10            ruangan-10       54.0
=====
BUILD SUCCESSFUL (total time: 1 second)
```

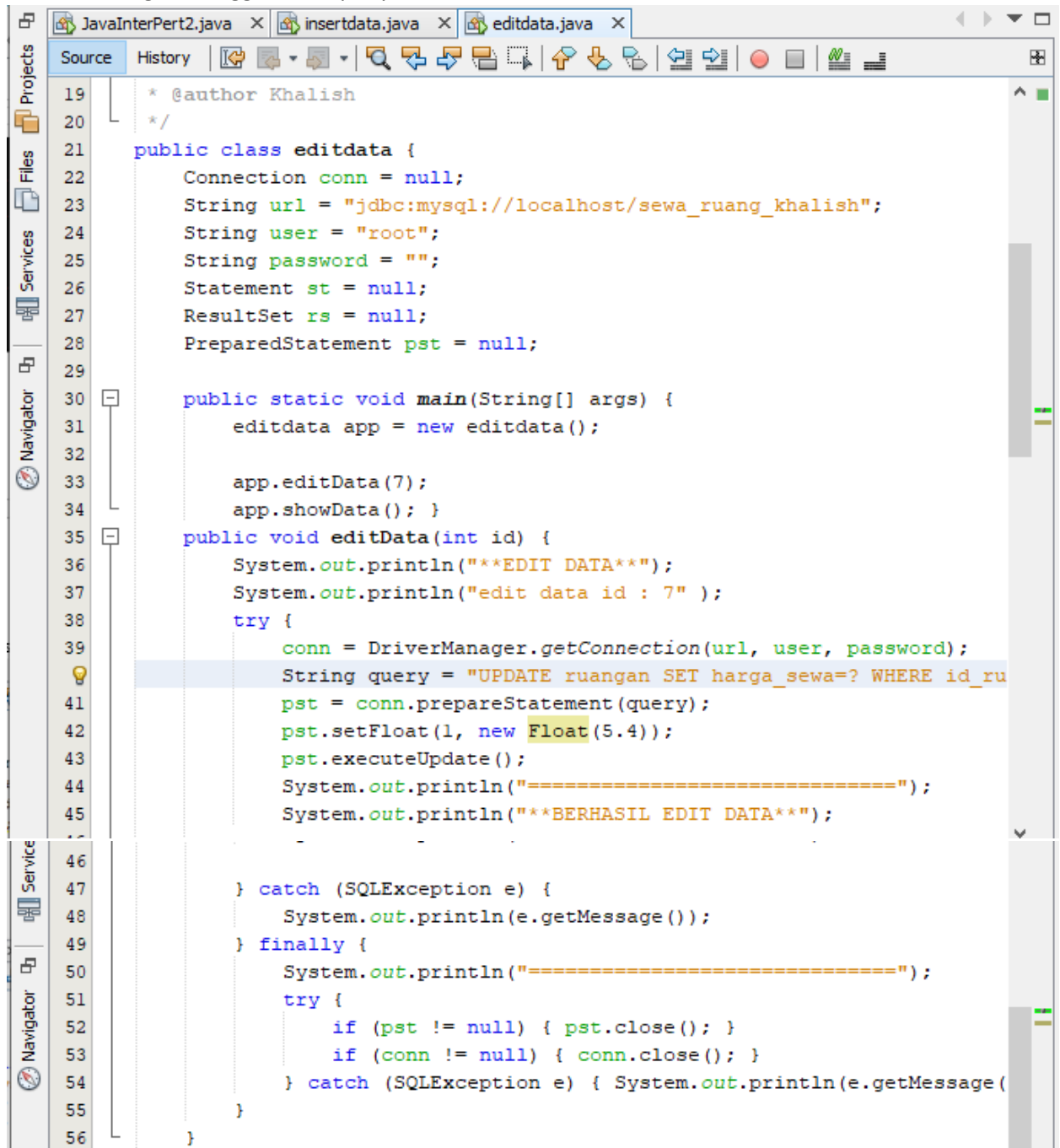
```
MariaDB [sewa_ruang_khalish]> SELECT * FROM ruangan;
```

```
+-----+-----+-----+
| id_ruang | jenis_ruang | harga_sewa |
+-----+-----+-----+
| 1        | ruangan-1   | 5.4        |
| 2        | ruangan-2   | 10.8       |
| 3        | ruangan-3   | 16.2       |
| 4        | ruangan-4   | 21.6       |
| 5        | ruangan-5   | 27         |
| 6        | ruangan-6   | 32.4       |
| 7        | ruangan-7   | 37.8       |
| 8        | ruangan-8   | 43.2       |
| 9        | ruangan-9   | 48.6       |
| 10       | ruangan-10  | 54         |
+-----+-----+-----+
```

```
10 rows in set (0.001 sec)
```

```
MariaDB [sewa_ruang_khalish]>
```

4. Edit Data. Disini saya menggunakan query update untuk melakukan update data yang ada pada database dengan menggunakan query UPDATE.



```
19  * @author Khalish
20  */
21  public class editdata {
22      Connection conn = null;
23      String url = "jdbc:mysql://localhost/sewa_ruang_khalish";
24      String user = "root";
25      String password = "";
26      Statement st = null;
27      ResultSet rs = null;
28      PreparedStatement pst = null;
29
30      public static void main(String[] args) {
31          editdata app = new editdata();
32
33          app.editData(7);
34          app.showData(); }
35
36      public void editData(int id) {
37          System.out.println("***EDIT DATA***");
38          System.out.println("edit data id : 7" );
39          try {
40              conn = DriverManager.getConnection(url, user, password);
41              String query = "UPDATE ruangan SET harga_sewa=? WHERE id_ru";
42              pst = conn.prepareStatement(query);
43              pst.setFloat(1, new Float(5.4));
44              pst.executeUpdate();
45              System.out.println("=====");
46              System.out.println("***BERHASIL EDIT DATA***");
47
48          } catch (SQLException e) {
49              System.out.println(e.getMessage());
50          } finally {
51              System.out.println("=====");
52              try {
53                  if (pst != null) { pst.close(); }
54                  if (conn != null) { conn.close(); }
55              } catch (SQLException e) { System.out.println(e.getMessage()); }
56          }
```

editdata > editData > try >

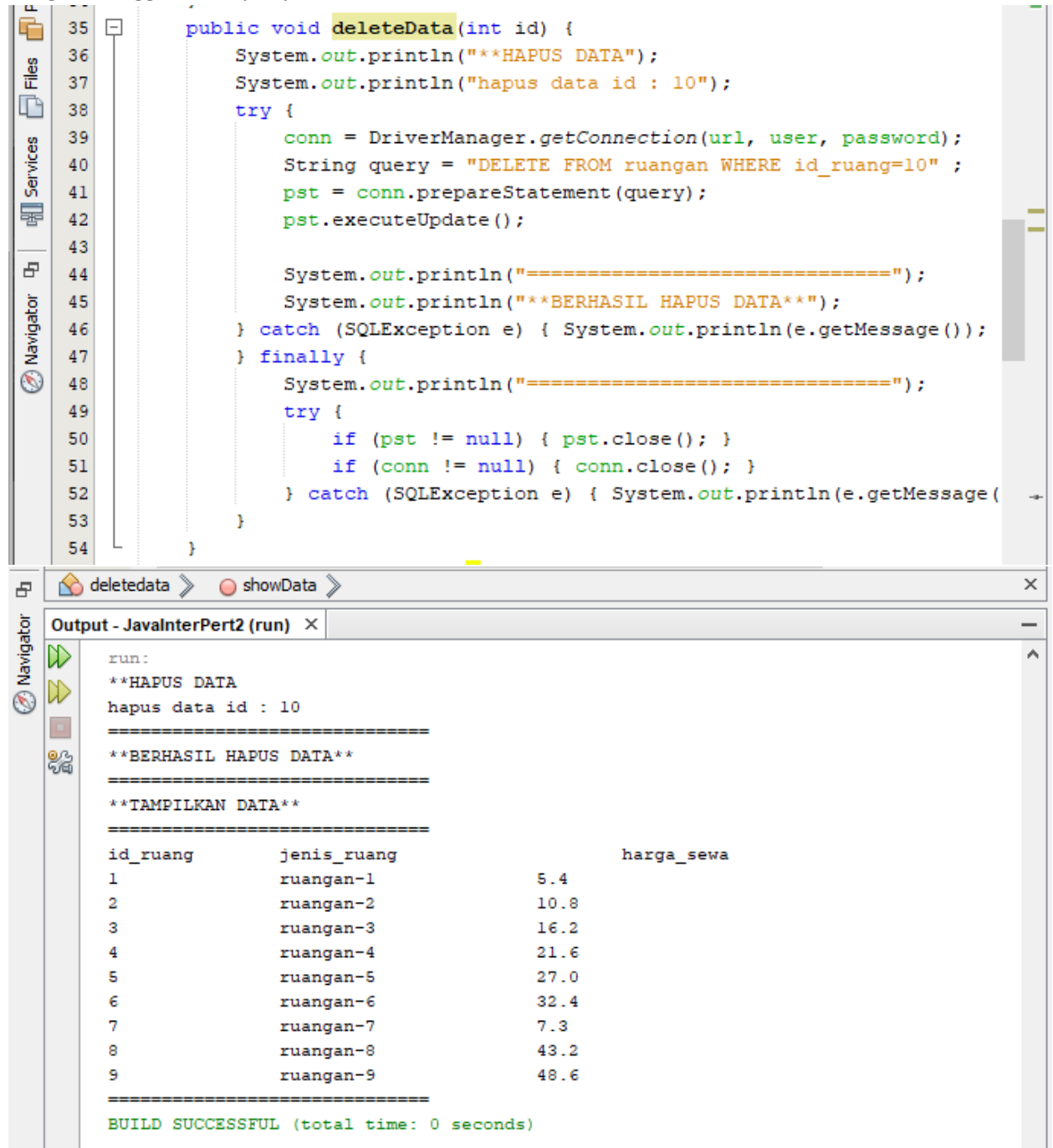
Output - JavaInterPert2 (run) x

```
run:
**EDIT DATA**
edit data id : 7
=====
**BERHASIL EDIT DATA**
=====
**TAMPILKAN DATA**
=====
id_ruang      jenis_ruang      harga_sewa
1             ruangan-1      5.4
2             ruangan-2      10.8
3             ruangan-3      16.2
4             ruangan-4      21.6
5             ruangan-5      27.0
6             ruangan-6      32.4
7             ruangan-7      7.3
8             ruangan-8      43.2
9             ruangan-9      48.6
10            ruangan-10     54.0
=====
BUILD SUCCESSFUL (total time: 0 seconds)
```

```
MariaDB [sewa_ruang_khalish]> SELECT * FROM ruangan;
+-----+-----+-----+
| id_ruang | jenis_ruang | harga_sewa |
+-----+-----+-----+
| 1        | ruangan-1   | 5.4        |
| 2        | ruangan-2   | 10.8       |
| 3        | ruangan-3   | 16.2       |
| 4        | ruangan-4   | 21.6       |
| 5        | ruangan-5   | 27         |
| 6        | ruangan-6   | 32.4       |
| 7        | ruangan-7   | 7.3        |
| 8        | ruangan-8   | 43.2       |
| 9        | ruangan-9   | 48.6       |
| 10       | ruangan-10  | 54         |
+-----+-----+-----+
10 rows in set (0.001 sec)

MariaDB [sewa_ruang_khalish]>
```

5. Menghapus data. Disini saya menghapus data dari dalam tabel yaitu dengan id\_ruang = 10 dengan menggunakan query DELETE.



The screenshot displays an IDE with a Java file named `deletedata`. The code defines a `deleteData` method that connects to a database, executes a `DELETE` query for `id_ruang = 10`, and prints the results. The output window shows the execution of the program, including the deletion message and a table of remaining data.

```
35 public void deleteData(int id) {
36     System.out.println("**HAPUS DATA");
37     System.out.println("hapus data id : 10");
38     try {
39         conn = DriverManager.getConnection(url, user, password);
40         String query = "DELETE FROM ruangan WHERE id_ruang=10" ;
41         pst = conn.prepareStatement(query);
42         pst.executeUpdate();
43
44         System.out.println("=====");
45         System.out.println("**BERHASIL HAPUS DATA**");
46     } catch (SQLException e) { System.out.println(e.getMessage()); }
47     finally {
48         System.out.println("=====");
49         try {
50             if (pst != null) { pst.close(); }
51             if (conn != null) { conn.close(); }
52         } catch (SQLException e) { System.out.println(e.getMessage()); }
53     }
54 }
```

Output - JavaInterPert2 (run) X

```
run:
**HAPUS DATA
hapus data id : 10
=====
**BERHASIL HAPUS DATA**
=====
**TAMPILKAN DATA**
=====
id_ruang    jenis_ruang    harga_sewa
1           ruangan-1      5.4
2           ruangan-2      10.8
3           ruangan-3      16.2
4           ruangan-4      21.6
5           ruangan-5      27.0
6           ruangan-6      32.4
7           ruangan-7      7.3
8           ruangan-8      43.2
9           ruangan-9      48.6
=====
BUILD SUCCESSFUL (total time: 0 seconds)
```

id_ruang	jenis_ruang	harga_sewa
1	ruangan-1	5.4
2	ruangan-2	10.8
3	ruangan-3	16.2
4	ruangan-4	21.6
5	ruangan-5	27.0
6	ruangan-6	32.4
7	ruangan-7	7.3
8	ruangan-8	43.2
9	ruangan-9	48.6



```
MariaDB [sewa_ruang_khalish]> SELECT * FROM ruangan;
```

id_ruang	jenis_ruang	harga_sewa
1	ruangan-1	5.4
2	ruangan-2	10.8
3	ruangan-3	16.2
4	ruangan-4	21.6
5	ruangan-5	27
6	ruangan-6	32.4
7	ruangan-7	7.3
8	ruangan-8	43.2
9	ruangan-9	48.6

```
9 rows in set (0.001 sec)
```

```
MariaDB [sewa_ruang_khalish]>
```