

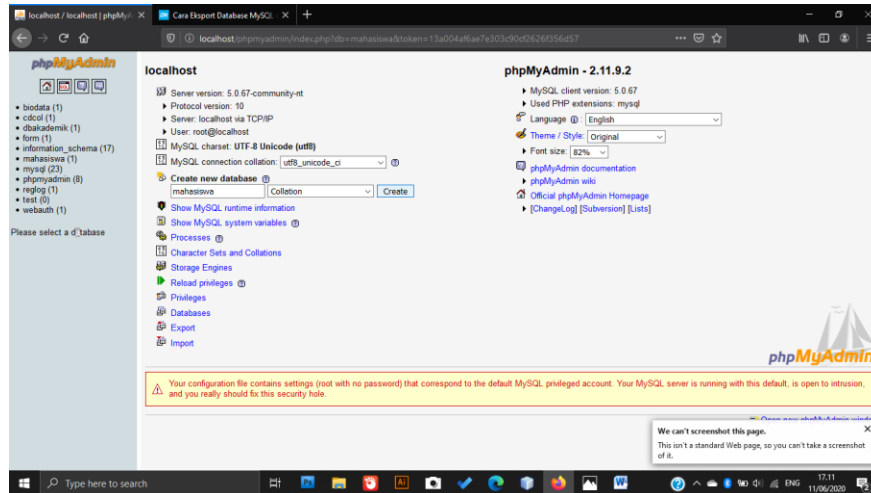
# Dokumentasi Pembuatan Form Data Mahasiswa

Nama : Amnah Kurnia

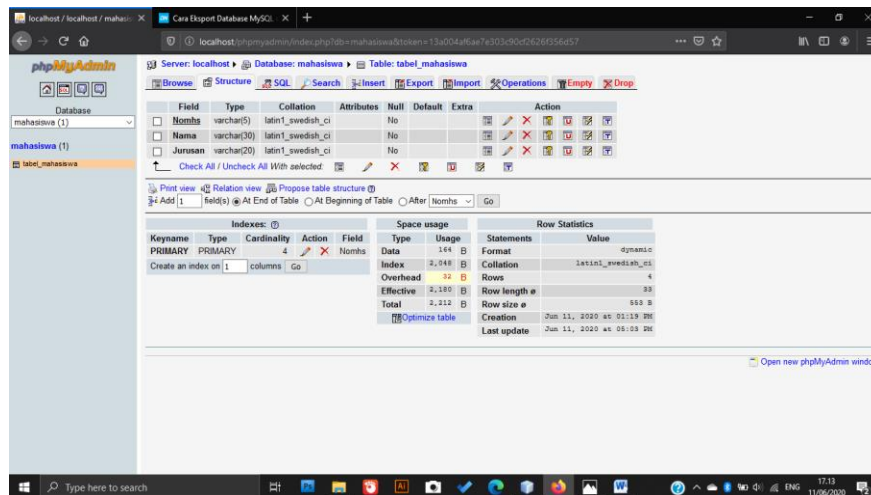
Stb : 1302010122

## 1. Pembuatan Database

### a. Database mahasiswa

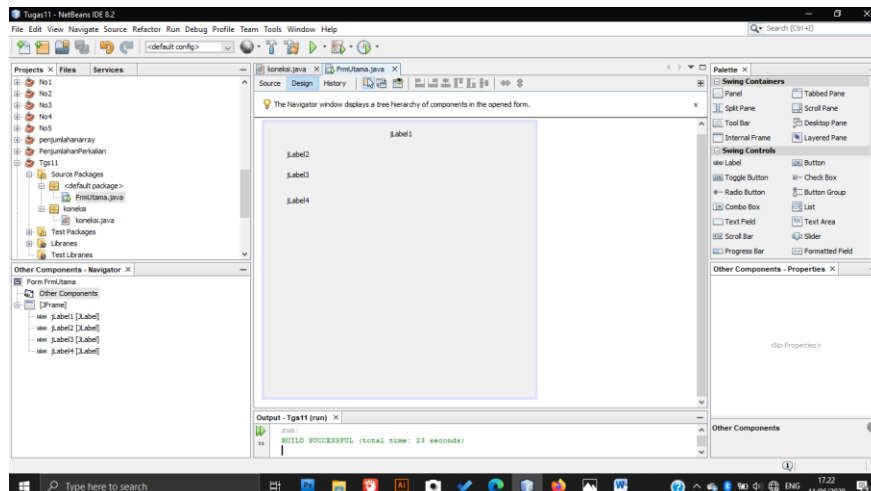


### b. Pembuatan tabel\_mahasiswa

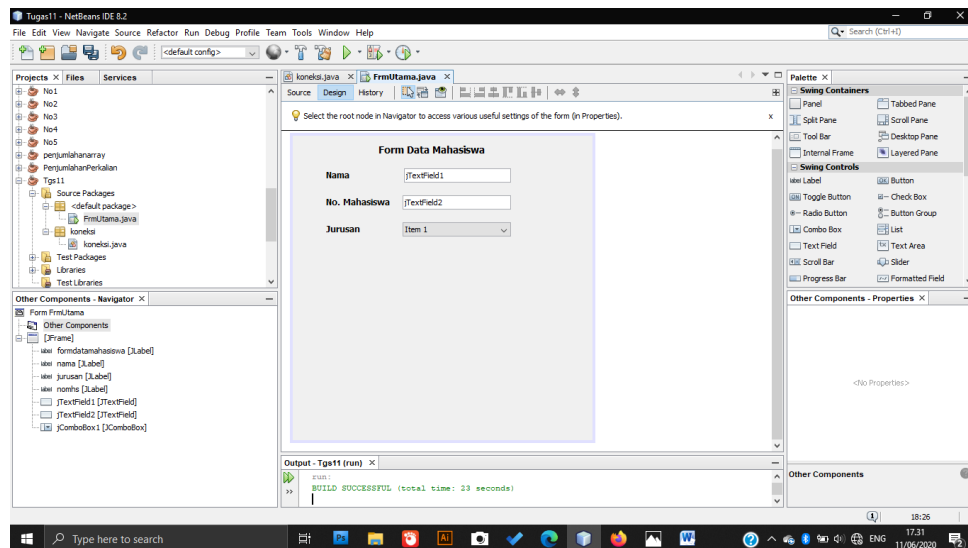
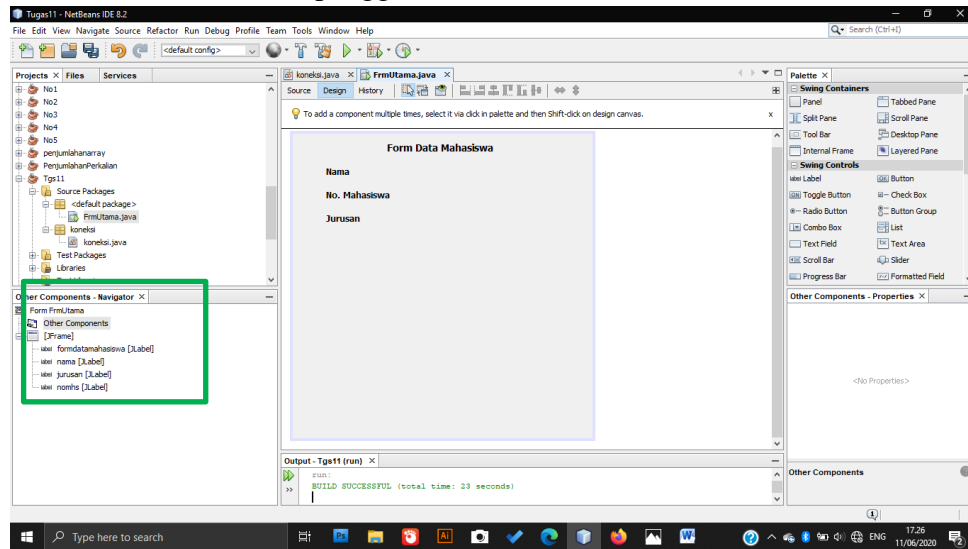


## 2. Tampilan Awal Form

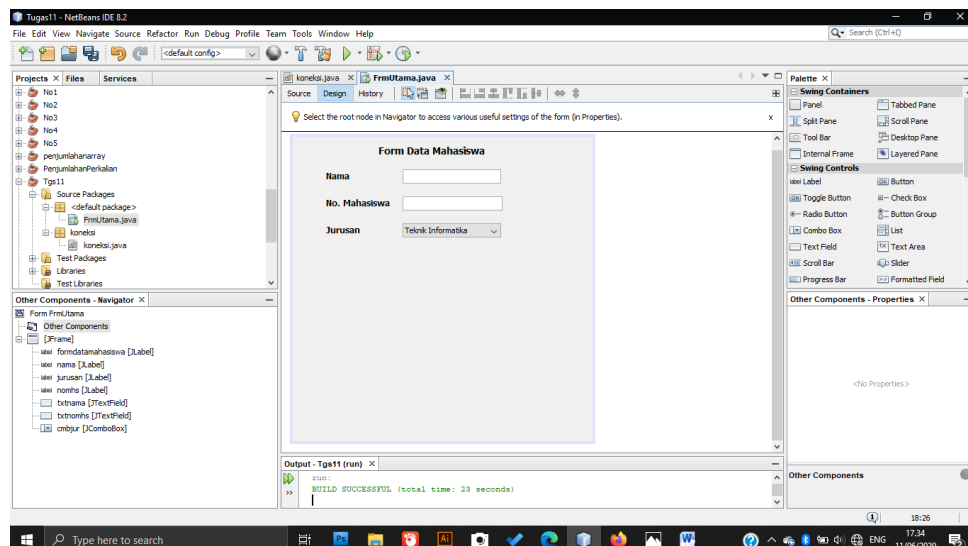
### a. Pembuatan judul untuk form data mahasiswa



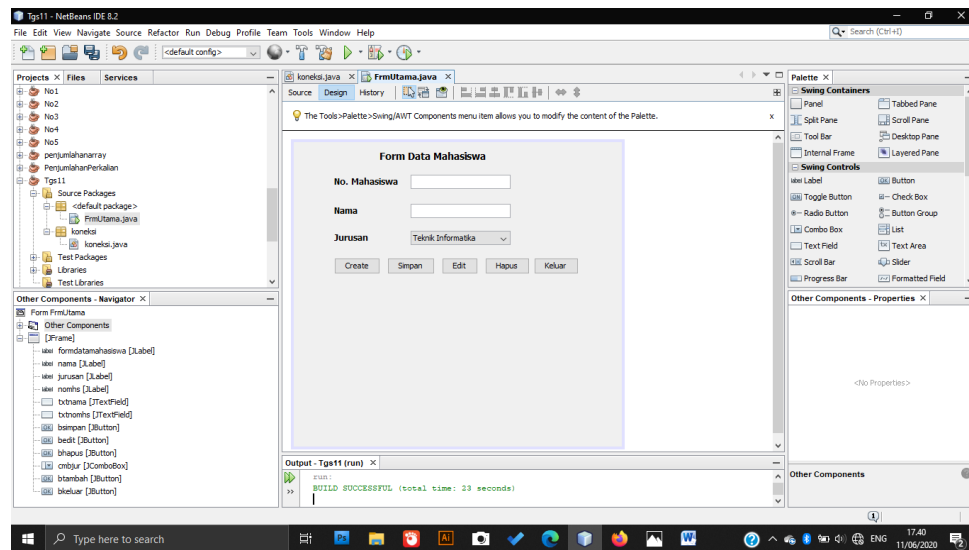
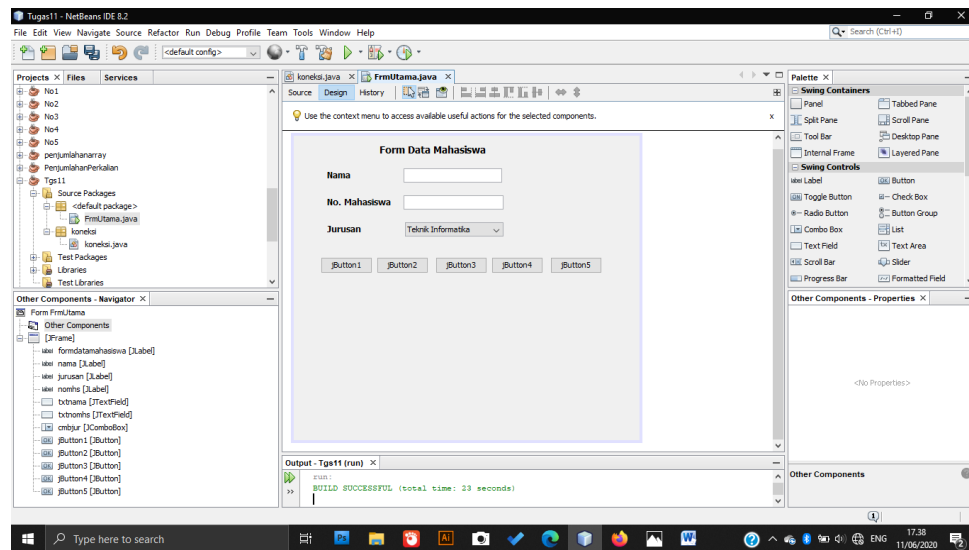
- Ganti nama sesuai data dari database yang dibuat dan jangan lupa uga untuk memberi nama di kotak hijau agar lebih mudah ketika di panggil.



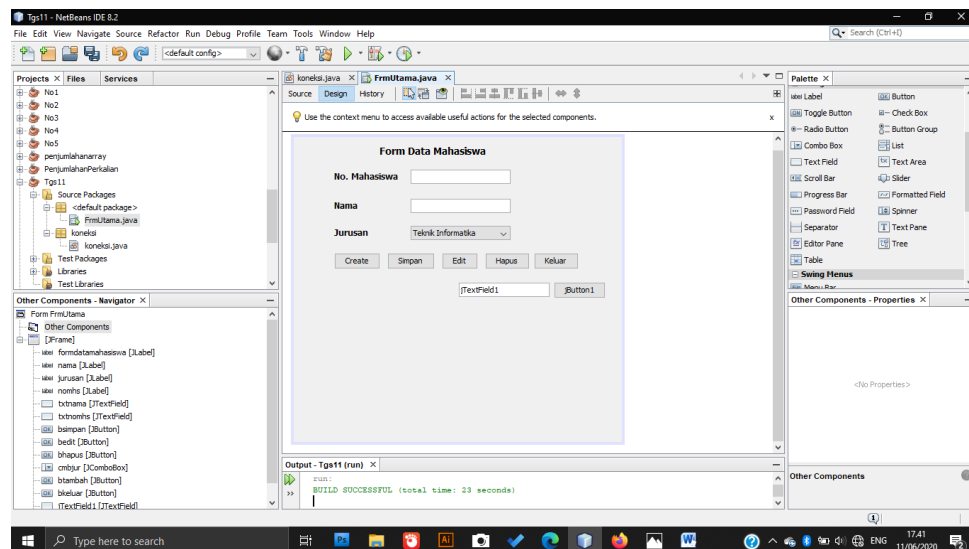
- Lakukan lagi seperti diatas, pada bagian jurusan saya membuat 2 jurusan yang dapat dipilih yaitu Teknik Informatika dan Sistem Inofrmasi.

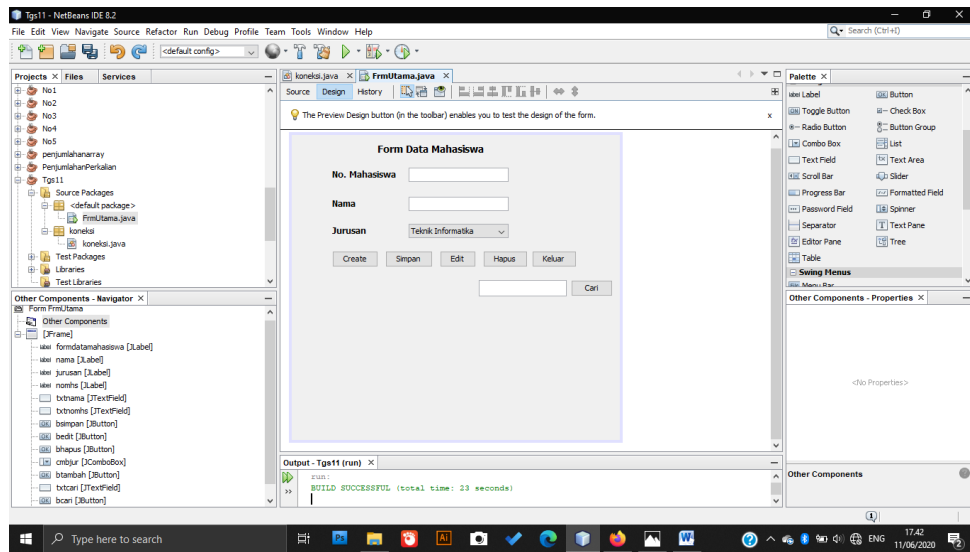


## b. Pembuatan Tombol untuk Form

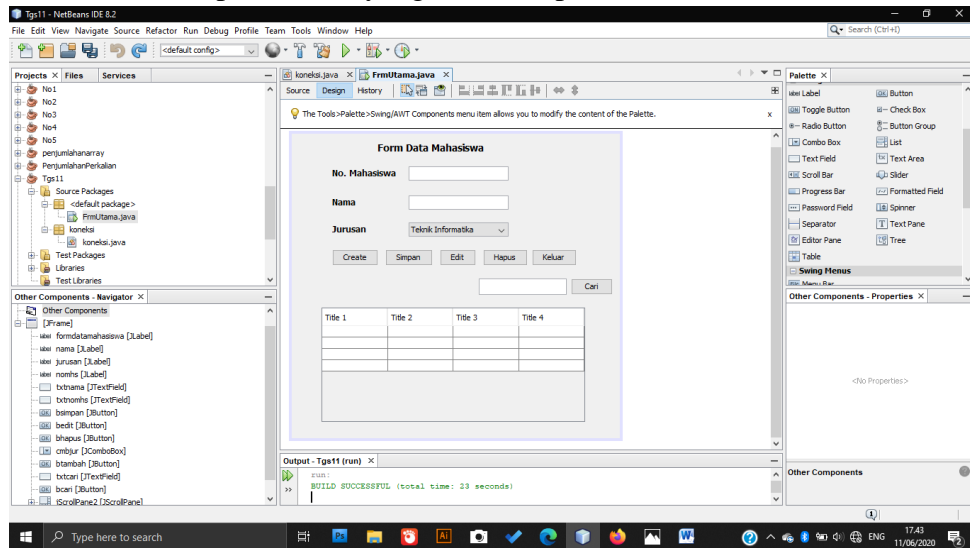


## c. Pembuatan Tombol Pencarian

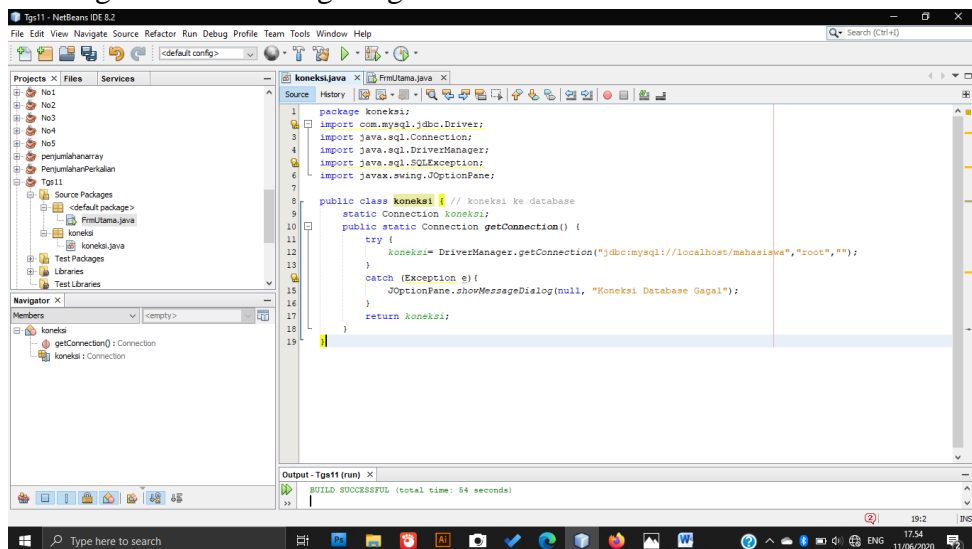




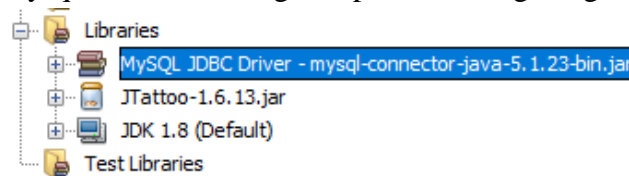
d. Pembuatan tabel untuk menampilkan data yang telah diinput di form



3. Pembuat File koneksi agar bisa terhubung dengan database



- Tambahkan juga bagian mysql JDBC driver agar dapat terhubung dengan database.



4. Pembuatan tambahan agar desain yang telah dibuat dapat berfungsi

- Bagian tombol create, berfungsi mengarahkan kita ke dalam teks yang mau kita isi di form. Tombol create saya namai dengan variable btambah.

```
private void btambahActionPerformed(java.awt.event.ActionEvent evt) {
    txtnomhs.setText("");
    txtnama.setText("");
    cmbjur.setSelectedItem("Teknik Informatika");
    txtnomhs.requestFocus();
}
```

- Bagian Tombol Simpan, berfungsi untuk menyimpan data yang kita input dan mengarahkannya ke tabel yang telah kita siapkan. Tombol simpan variabelnya bsimpan.

```
private void bsimpanActionPerformed(java.awt.event.ActionEvent evt) {
    String Nomhs = txtnomhs.getText();
    String Nama = txtnama.getText();
    String Jurusan = (String) cmbjur.getSelectedItem();
    try {
        Statement statement = (Statement) koneksi.getConnection().createStatement();
        statement.executeUpdate("insert into tabel_mahasiswa VALUES ('"+ Nomhs + "','"+ Nama + "','"+ Jurusan + "')");
        statement.close();
        JOptionPane.showMessageDialog(null, "Data Berhasil Disimpan");
    } catch (Exception e) {
        JOptionPane.showMessageDialog(null, "Data Gagal Disimpan");
    }
}
```

- Bagian Tombol Edit, berfungsi mengedit data yang telah diinputkan sebelumnya. Tombol edit variabelnya bedit.

```
private void beditActionPerformed(java.awt.event.ActionEvent evt) {
    if (bedit.getText().equals("bedit")){
        txtnomhs.requestFocus();
    } else {
        String Nomhs = txtnomhs.getText();
        String Nama = txtnama.getText();
        String Jurusan = (String) cmbjur.getSelectedItem();

        try {
            Statement statement = (Statement) koneksi.getConnection().createStatement();
            statement.executeUpdate("update tabel_mahasiswa SET Nomhs='"+Nomhs+"', "+ "Nama='"+
                +Nama+"', "+ "Jurusan='"+Jurusan+"' "+ "WHERE Nomhs='"+Nomhs+"' OR Nama='"+Nama+"'");
            statement.close();
            JOptionPane.showMessageDialog(null, "Data Berhasil Diubah");
        } catch (Exception e) {
            JOptionPane.showMessageDialog(null, "Data Gagal Diubah");
        }
        datatable();
    }
}
```

- d. Bagian Tombol Hapus, berfungsi untuk menghapus data yang telah diinputkan sebelumnya. Variablenya bhapus.

```
private void bhapusActionPerformed(java.awt.event.ActionEvent evt) {  
    String Nomhs = txtnomhs.getText();  
    String Nama = txtnama.getText();  
    try {  
        Statement statement = (Statement) koneksi.getConnection().createStatement();  
        statement.executeUpdate("delete from tabel_mahasiswa where Nomhs='"+Nomhs+"' OR Nama='"+Nama+"'");  
        JOptionPane.showMessageDialog(null, "Data Berhasil Dihapus");  
        txtnomhs.setText("");  
        txtnama.setText("");  
        cmbjur.setSelectedItem("Teknik Informatika");  
    } catch (Exception e) {  
        JOptionPane.showMessageDialog(null, "Data Gagal Dihapus");  
    }  
    datatable();  
}
```

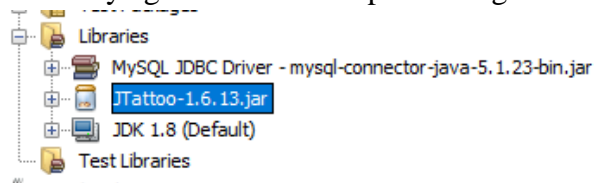
- e. Bagian Tombol Keluar, berfungsi untuk keluar dari program. Variabelnya bkeluar.

```
private void bkeluarActionPerformed(java.awt.event.ActionEvent evt) {  
    dispose();  
}
```

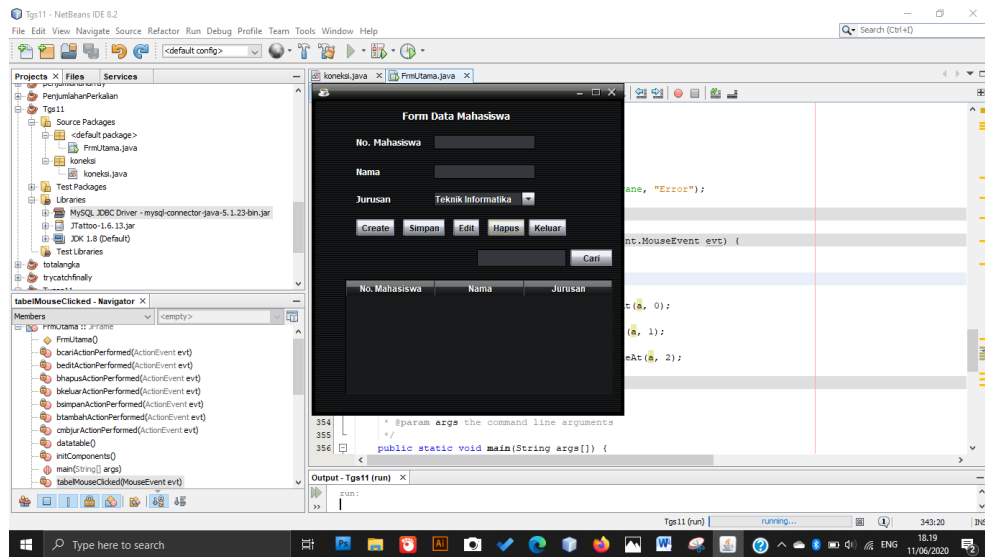
- f. Mengubungkan agar ketika mengklik data yang telah diinputkan pada tabel maka akan terhubung pada bagian form.

```
private void tabelMouseClicked(java.awt.event.MouseEvent evt) {  
    int a = tabel.getSelectedRow();  
    if (a == -1) {  
        return;  
    }  
    String Nomhs = (String) tabel.getValueAt(a, 0);  
    txtnomhs.setText(Nomhs);  
    String Nama = (String) tabel.getValueAt(a, 1);  
    txtnama.setText(Nama);  
    String Jurusan = (String) tabel.getValueAt(a, 2);  
    cmbjur.setSelectedItem(Jurusan);  
}
```

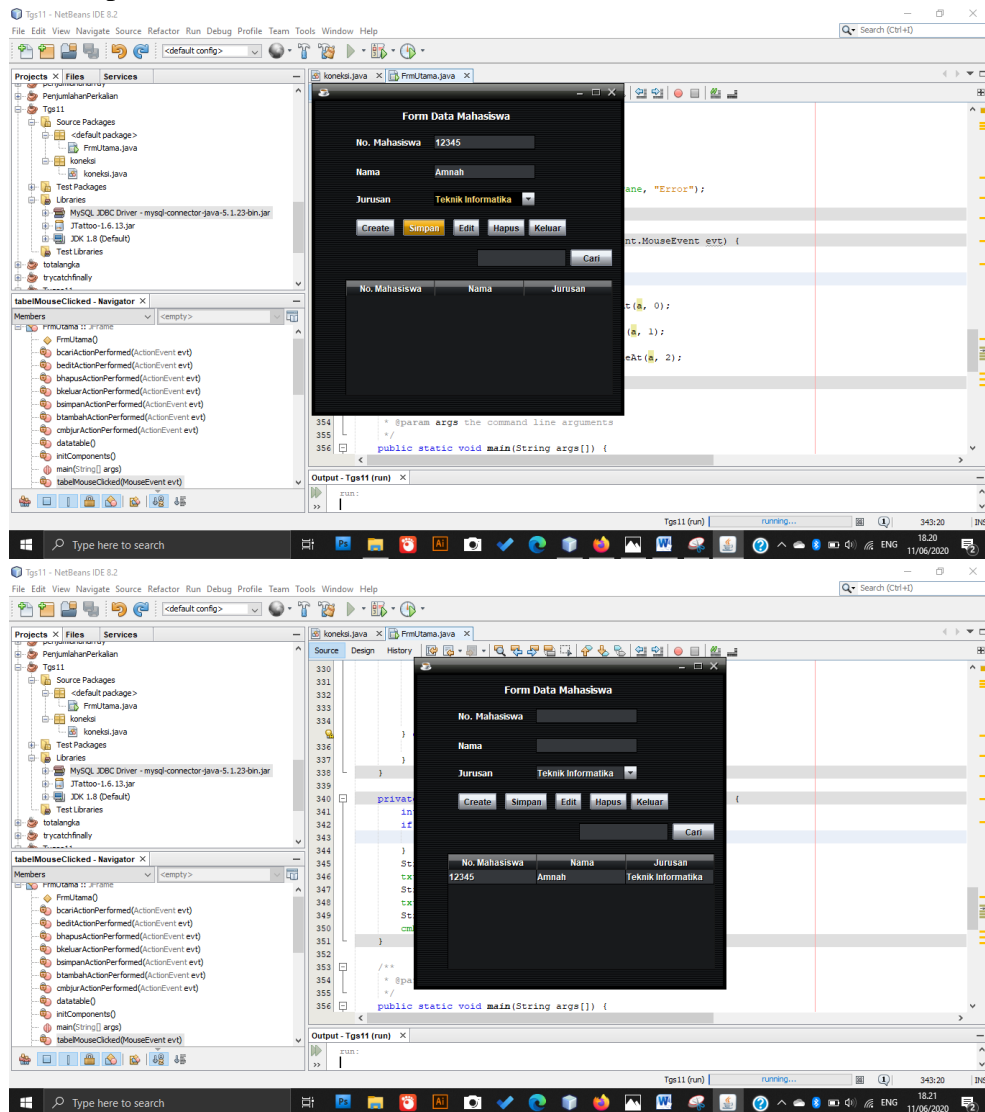
- g. Menambahkan JTatto dibagian library agar membuat tampilan design JFrame Lebih bagus



Output :  
Tampilan Awal

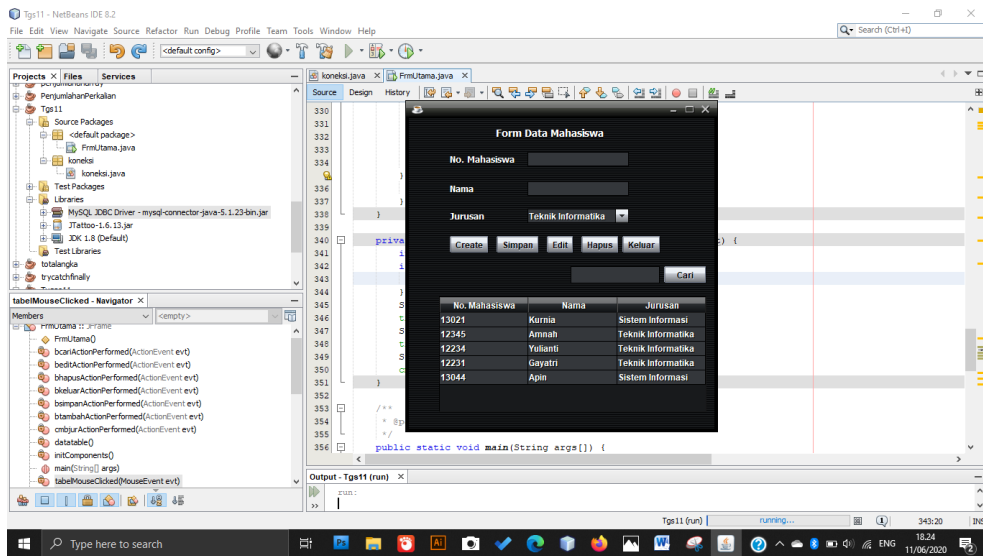


### a. Bagian Tombol Simpan

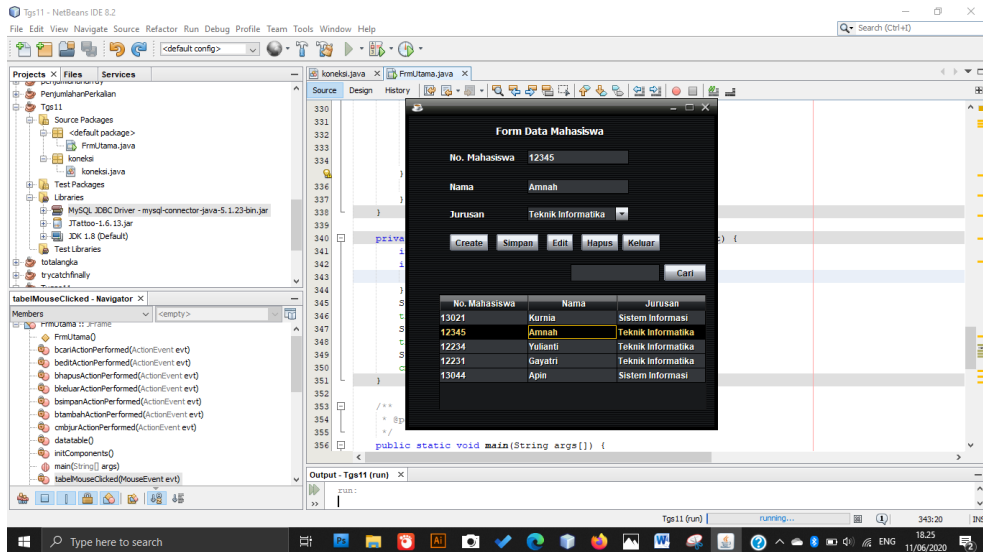


Saya akan membuat lagi sampai 5 data

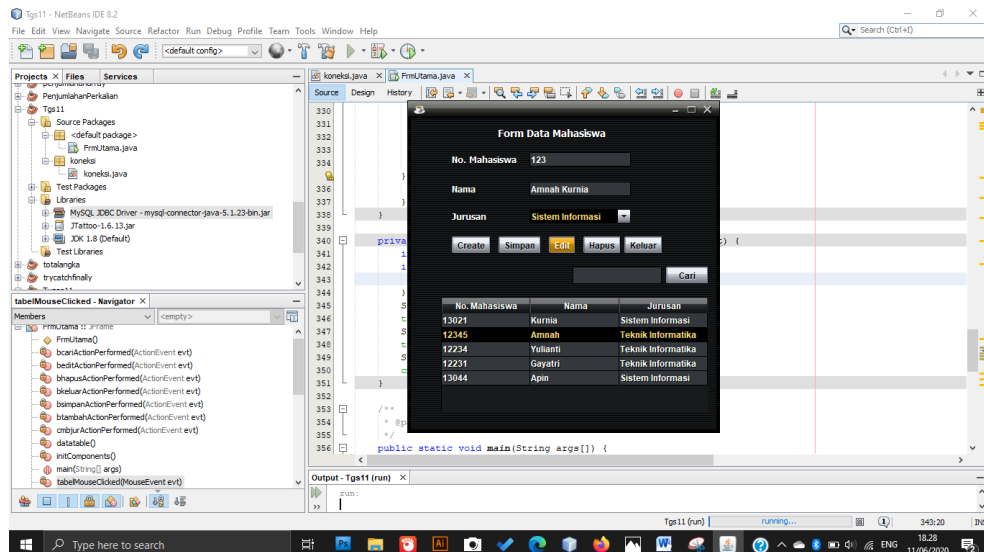




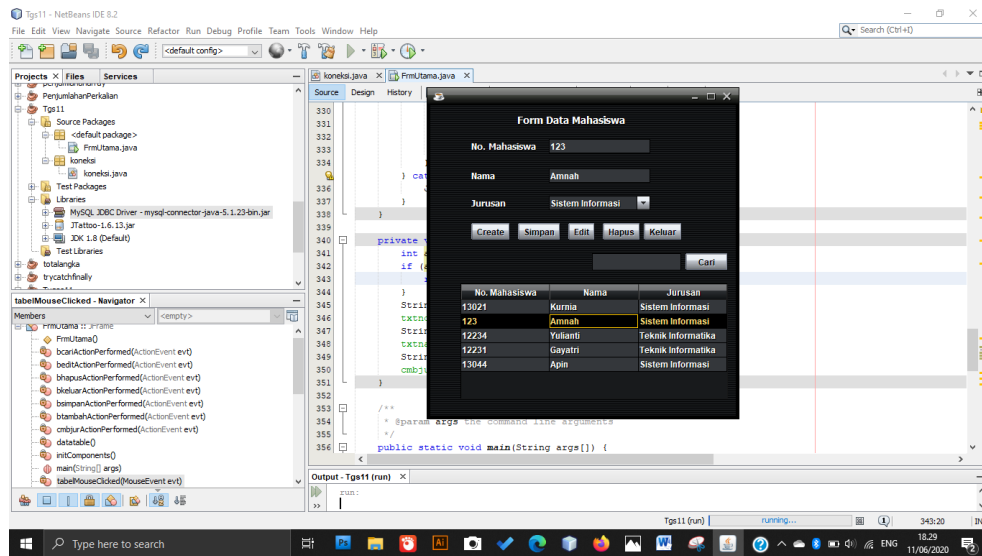
Ketika di klik salah satu data di tabel maka akan seperti ini



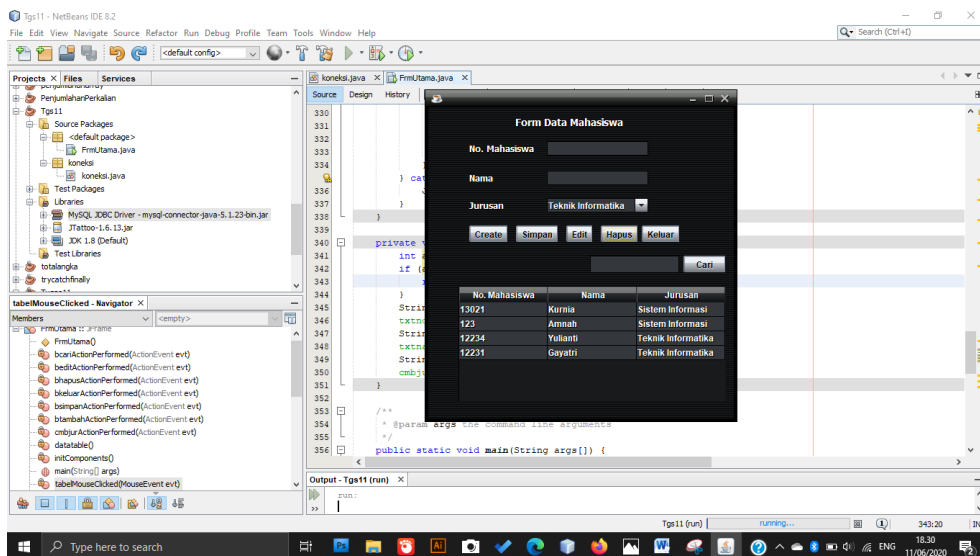
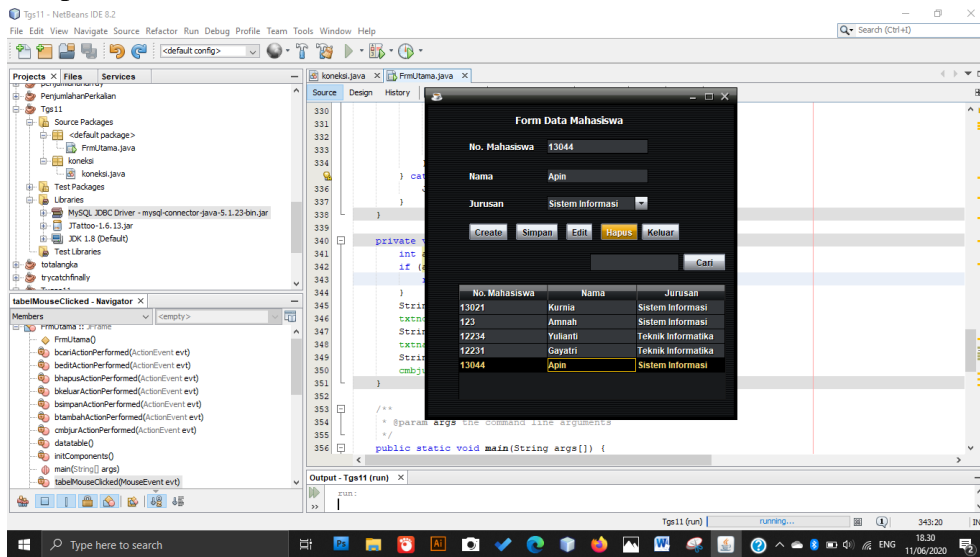
## b. Bagian Tombol Edit



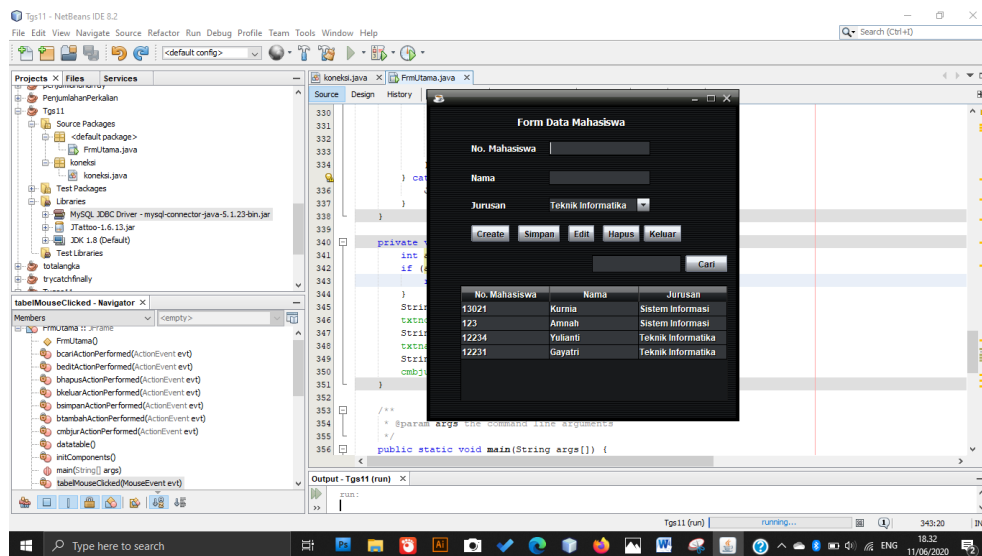
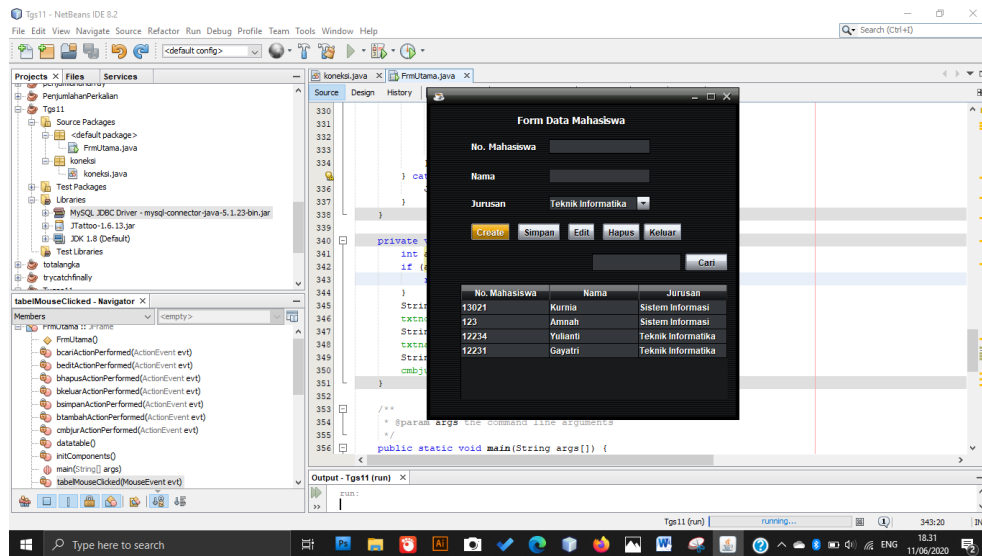




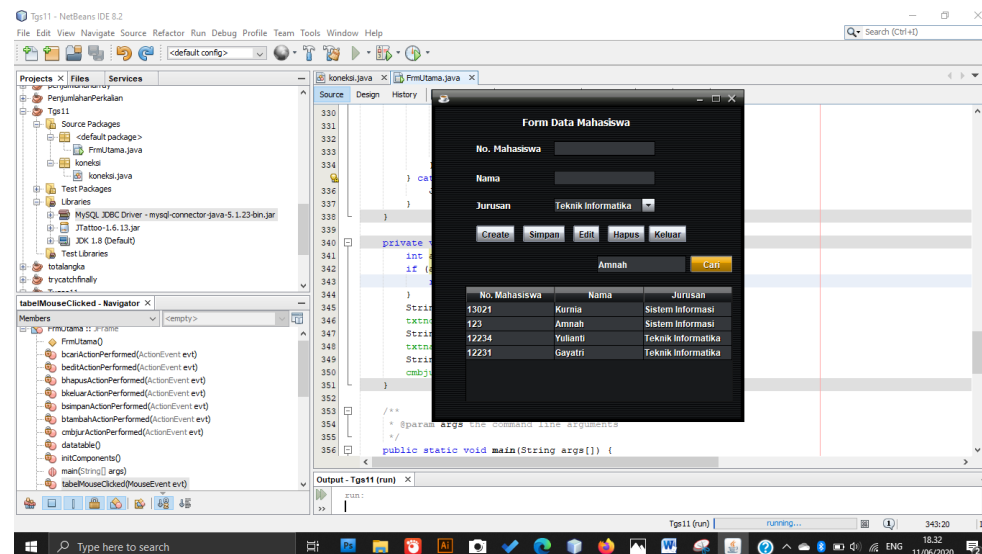
### c. Bagian Tombol Hapus

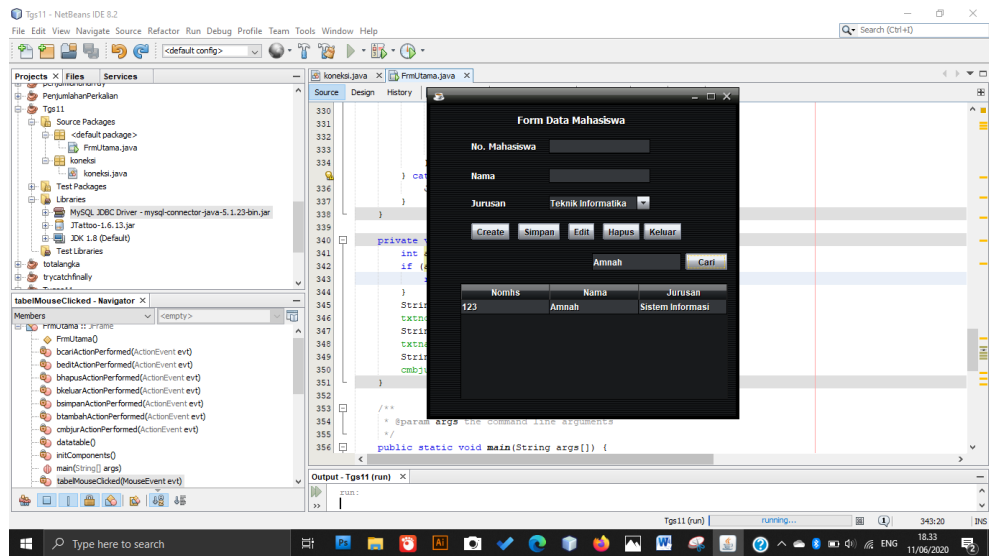


#### d. Bagian Tombol Create

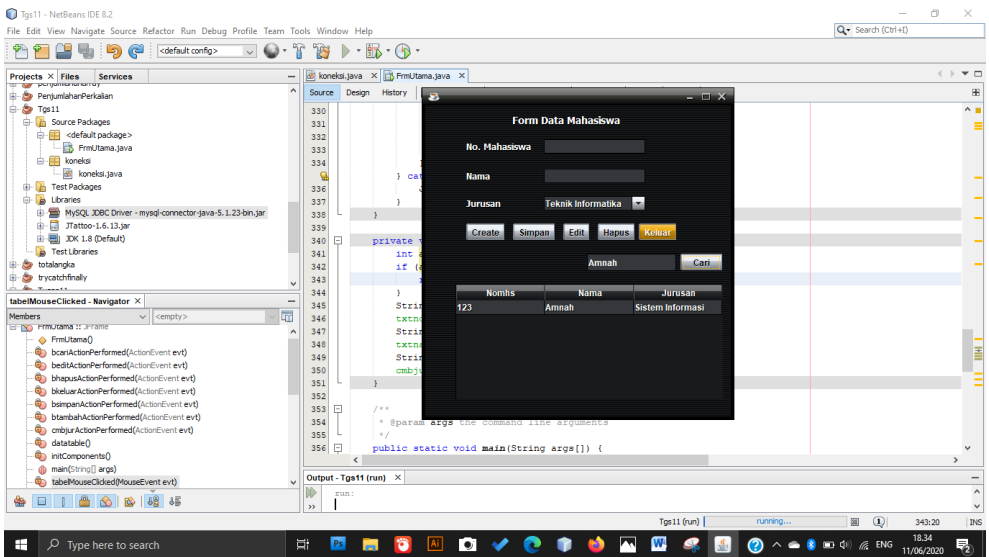


#### e. Bagian Tombol Cari





## f. Bagian Tombol Keluar



## Program telah keluar

