

# NESNE TABANLI PROGRAMLAMA I

Ders 6

Uzay Çetin

May 9, 2013

# İçindekiler

<b>1</b>	<b>Netbeans Palette yardımıyla Programlama</b>	<b>2</b>
1.1	Basit Hesap Makinesi . . . . .	3
1.1.1	Yeni bir proje açmak . . . . .	3
1.1.2	Netbeans ile JFrame Form” açmak . . . . .	7
1.1.3	Netbeans Paleti . . . . .	9
1.1.4	JPanel eklemek . . . . .	10
1.1.5	JButton Eklemek . . . . .	15
1.1.6	jLabel ve jTextField Eklemek . . . . .	21
1.2	Palette yardımıyla Olay algılamak . . . . .	27
1.2.1	Butona basıldığında olay algılamak . . . . .	27

# Bölüm 1

## Netbeans Palette yardımıyla Programlama

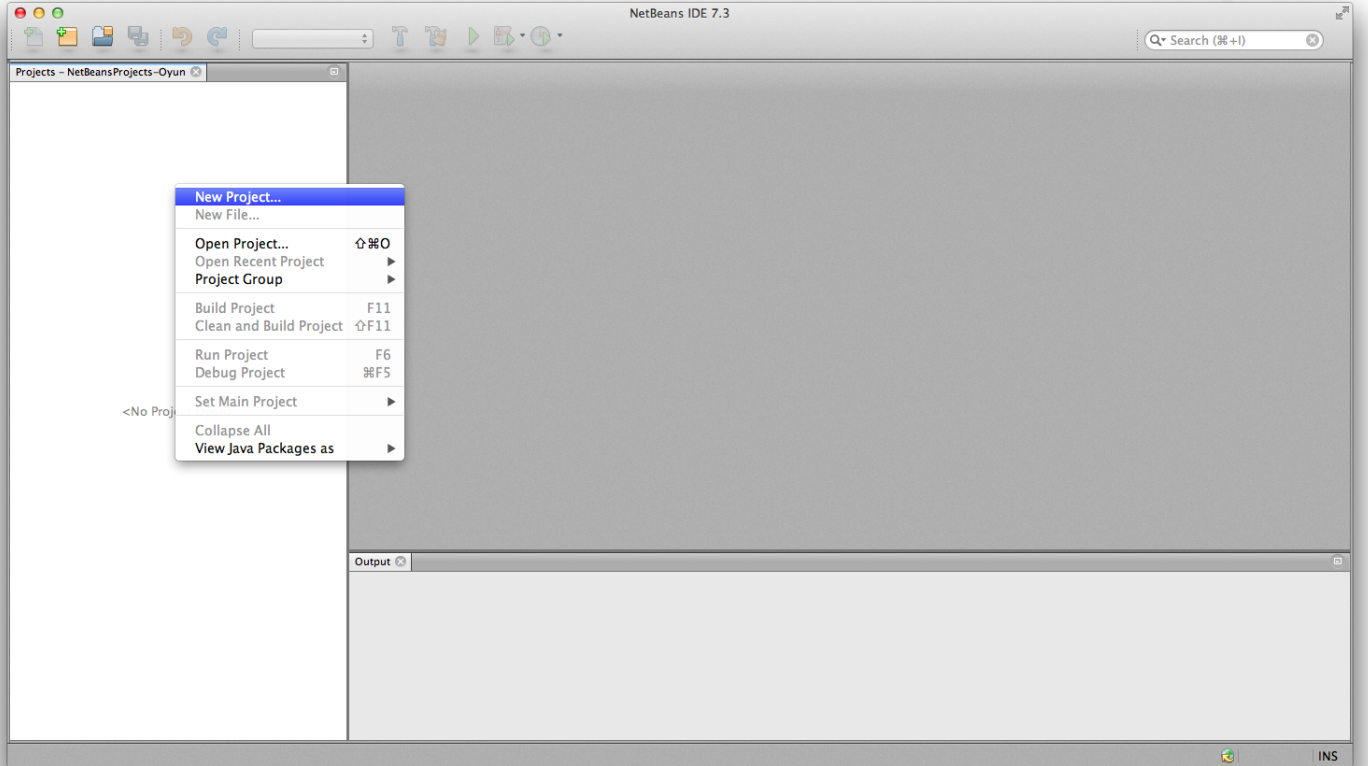
Grafiksel programlama diyerek, üzerinde buton, menü, yazı girdi alanı vb. görsel öğeler içeren programları kastediyoruz. Bu derste, bu tür bir programı, Netbeans ile kolayca yapmanın yolunu göreceğiz.

## 1.1 Basit Hesap Makinesi

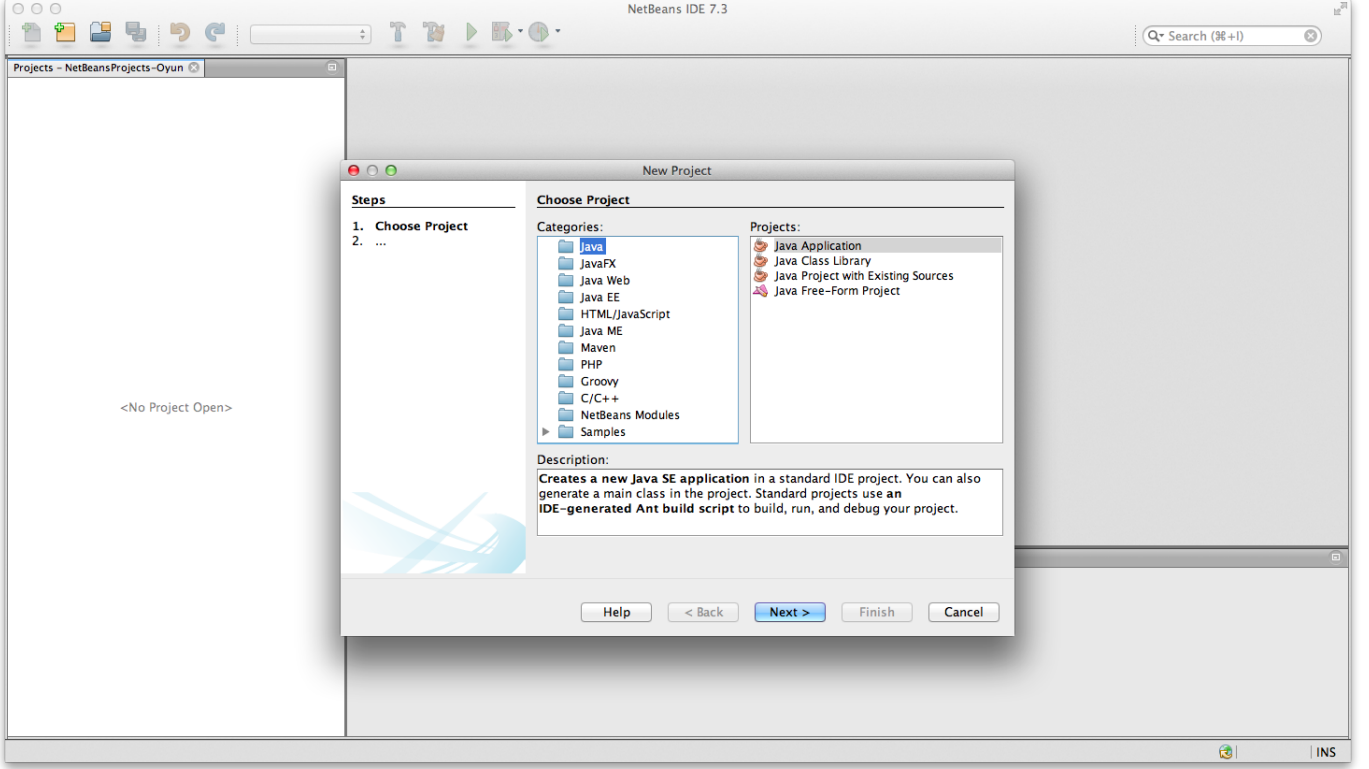
Örnek olarak, basit bir hesap makinesi yapalım.

### 1.1.1 Yeni bir proje açmak

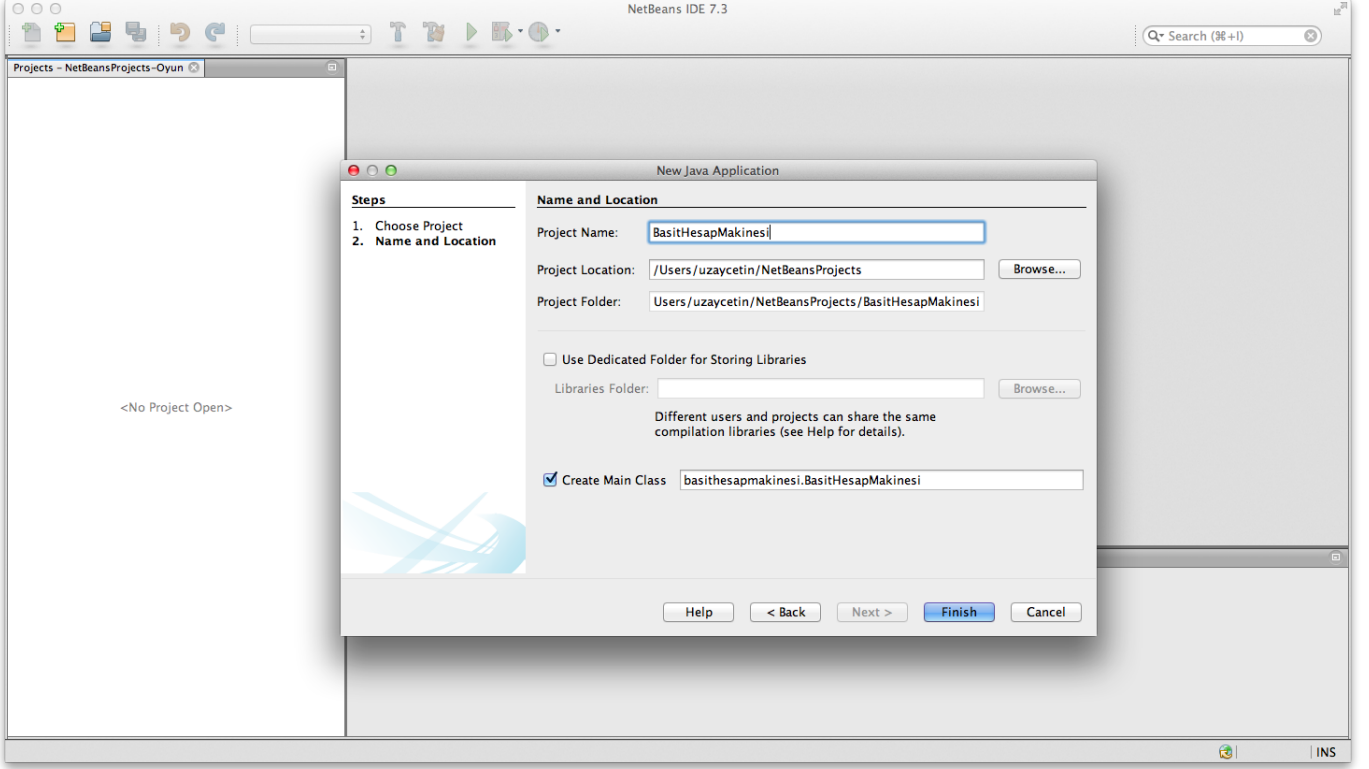
Öncelikle, yeni bir proje açmamız gerekli.



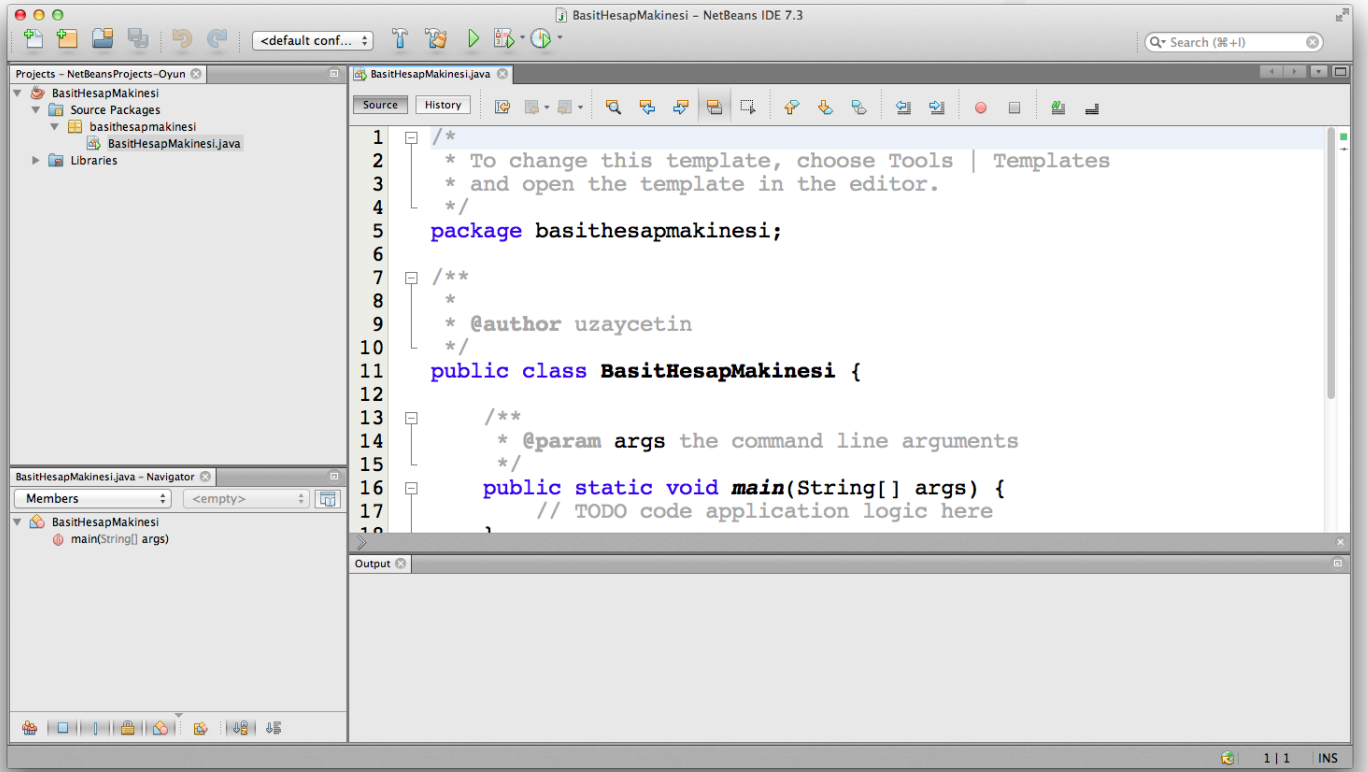
Şekil 1.1: Yeni bir proje açalım: Sol taraftaki project alanında, sağ tıklayarak, "New Project"i seçelim.



Şekil 1.2: Java Application seçiliyken, next diyelim.



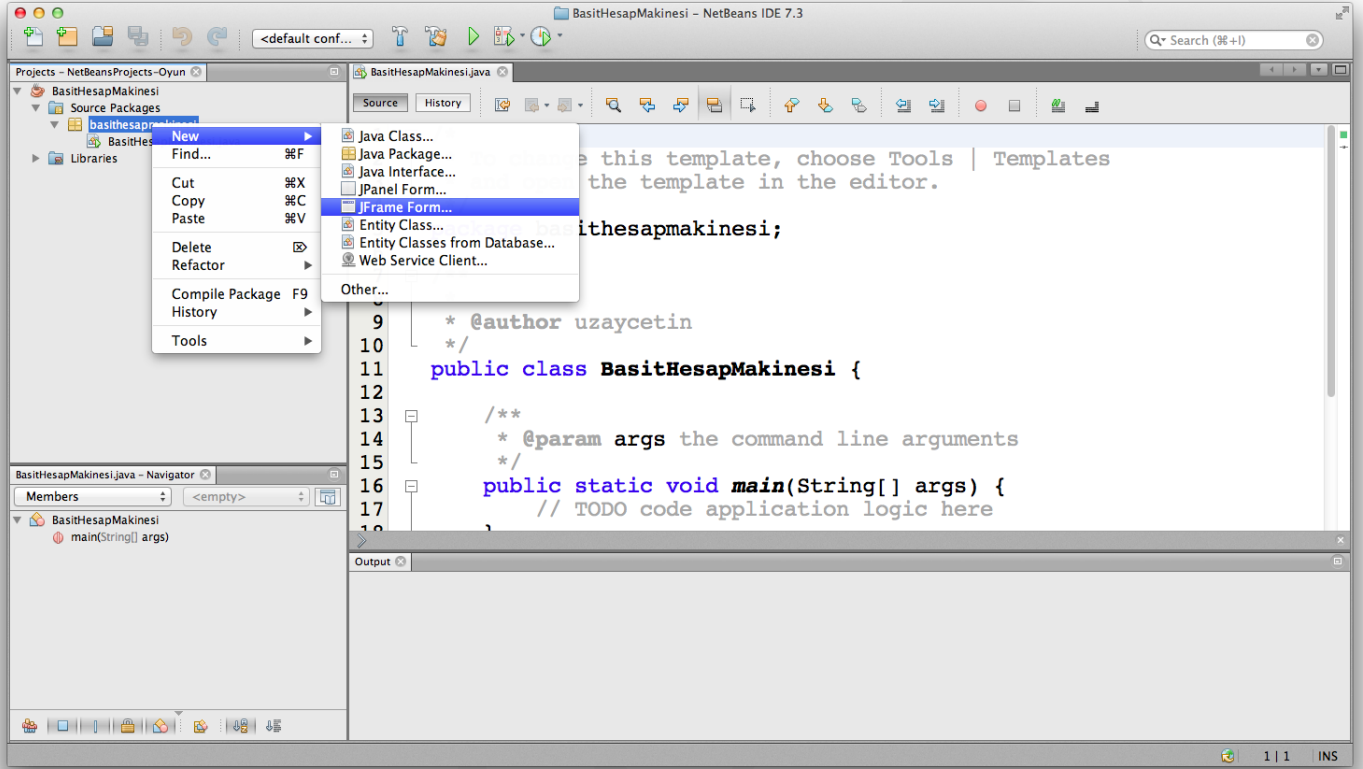
Şekil 1.3: Proje adını belirleyelim: Projemizin adı "BasitHesapMakinesi" olsun. Daha sonra "Finish" diyelim.



Şekil 1.4: "BasitHesapMakinesi" adındaki projemiz yaratıldı. Aynı isimle paket ve onun içinde de bir java sınıfı otomatik olarak yaratılmış oldu.

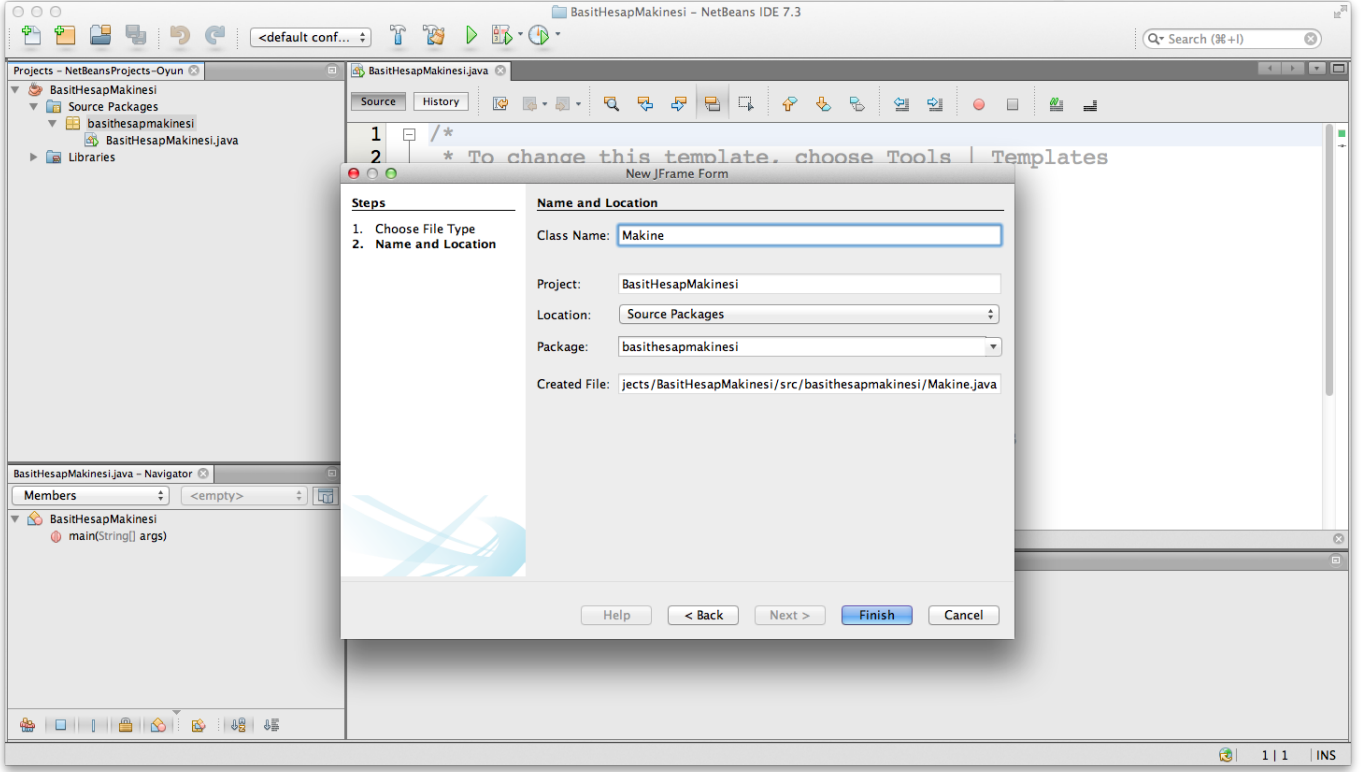
### 1.1.2 Netbeans ile JFrame Form” açmak

Biz Netbeans’in Palet özeliğinden yararlanmak istiyoruz. Bunun için yeni bir ”JFrame Form” yaratmaya ihtiyacımız var.



Şekil 1.5: Faremizi paket isminin üzerine getirerek, önce ”New” daha sonra ”JFrame Form”u seçiyoruz. Böylece, kolayca tasarımı yapabileceğimiz bir java sınıf formu yaratmış olacağız.

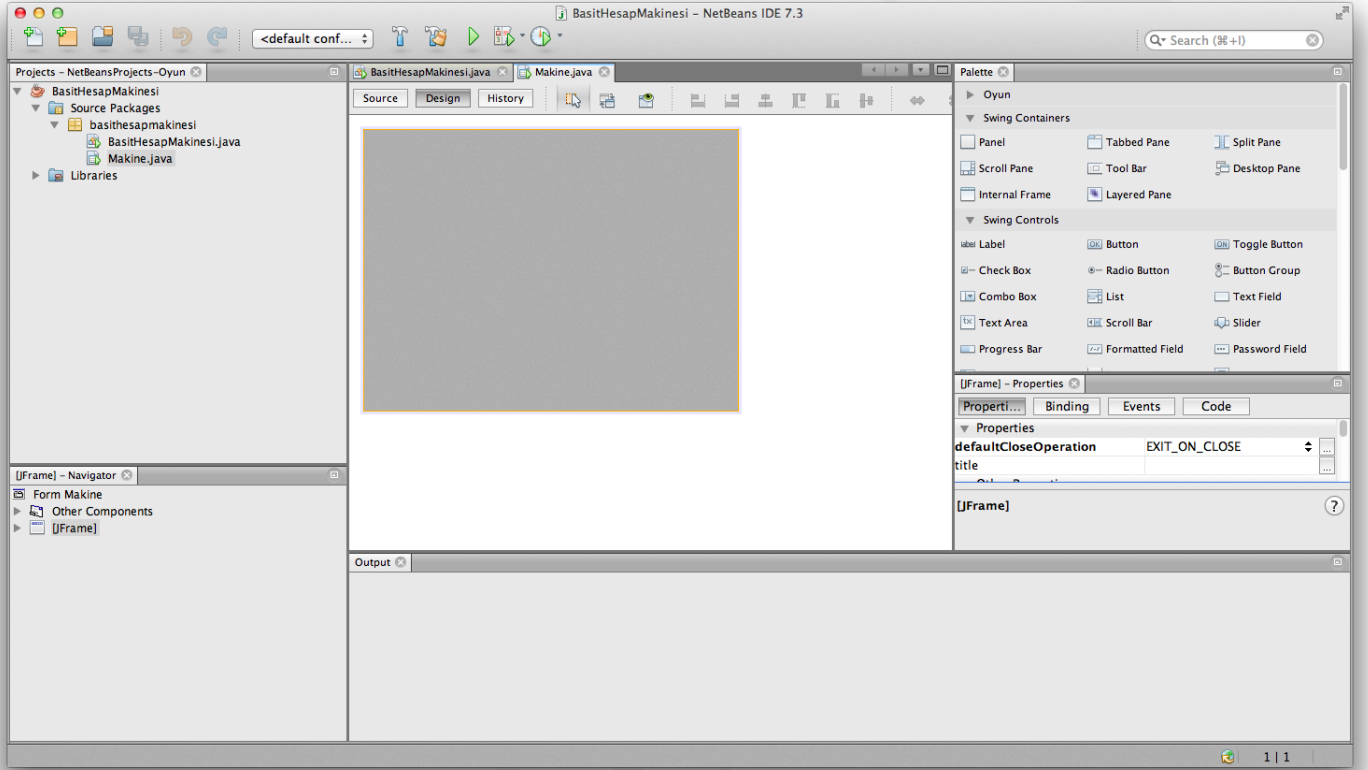




Şekil 1.6: "JFrame Form"un adını yazıyoruz: "Makine". Daha sonra Finish diyeceğiz.

### 1.1.3 Netbeans Paleti

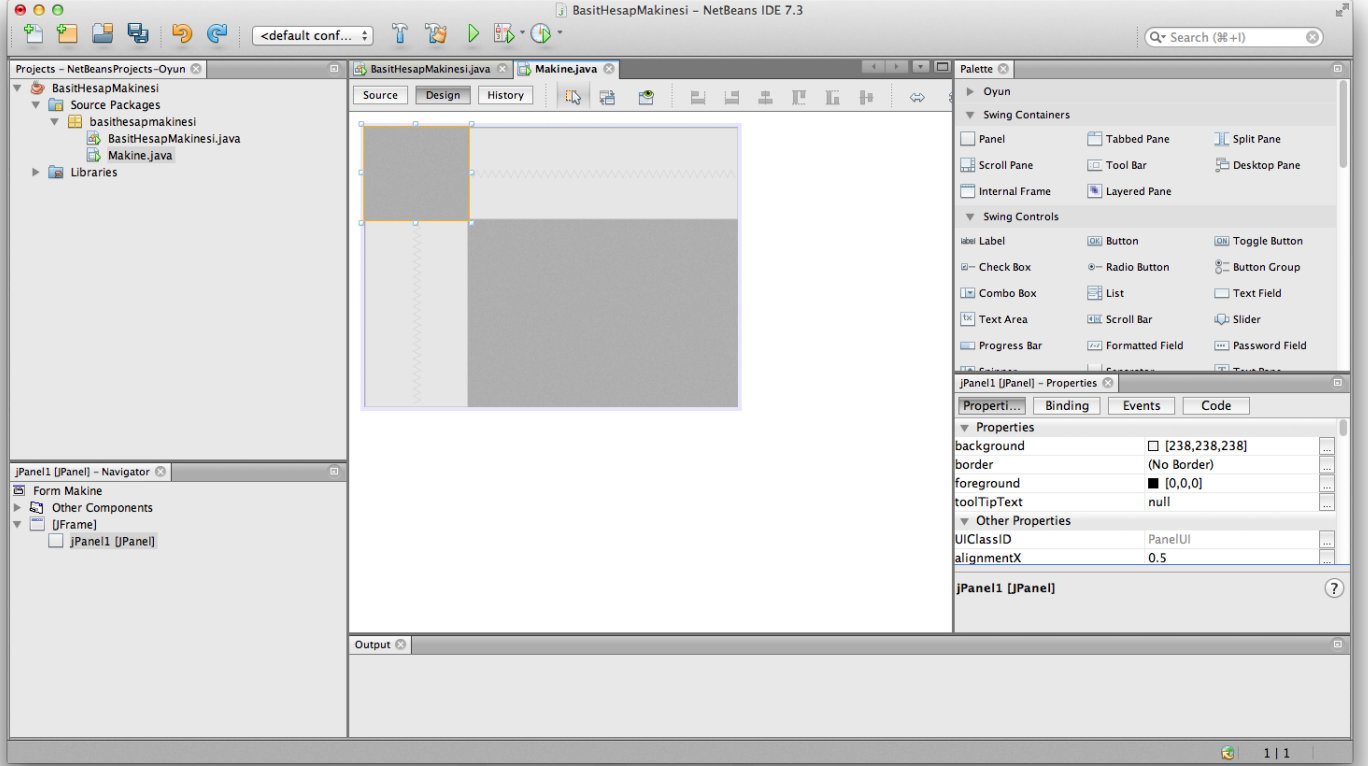
Netbeans Palet yardımıyla, bir çok java sınıfını, sürükle-bırak yöntemiyle kolayca, programımıza ekleyebiliriz. Saniyeler içinde çalışan görsel bir program yapmak Netbeans palet yardımıyla çok kolay.



Şekil 1.7: Böylece "Makine" adında içi boş bir "JFrame Form"u yarattık. Sağ tarafta palet penceresinin içinde bir çok buton, metin girdi alanı gibi java sınıflarını görmektedir.

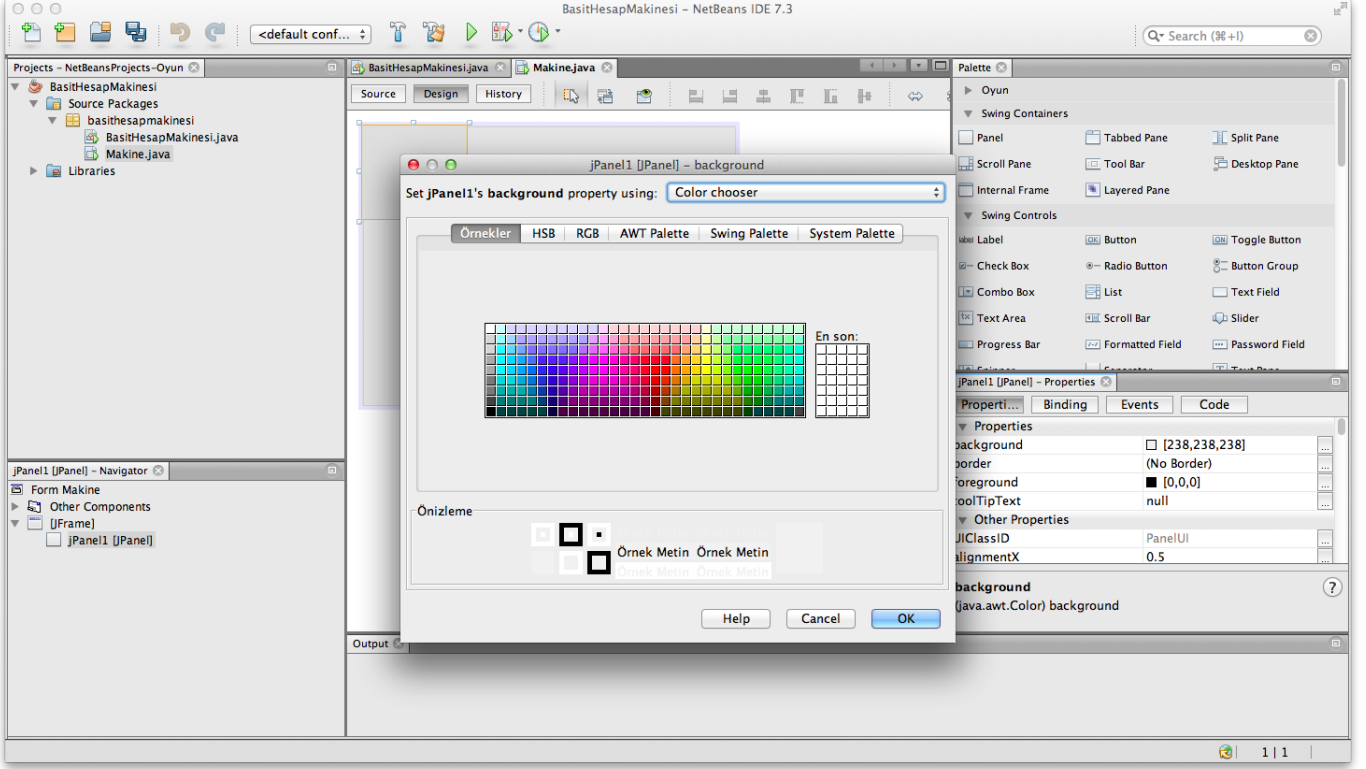
### 1.1.4 JPanel eklemek

Genelde içine resim çizmek, buton ya da diğer görsel öğeleri yerleştirmek için kullandığımız java kapsayıcı sınıfına verilen addır, JPanel.

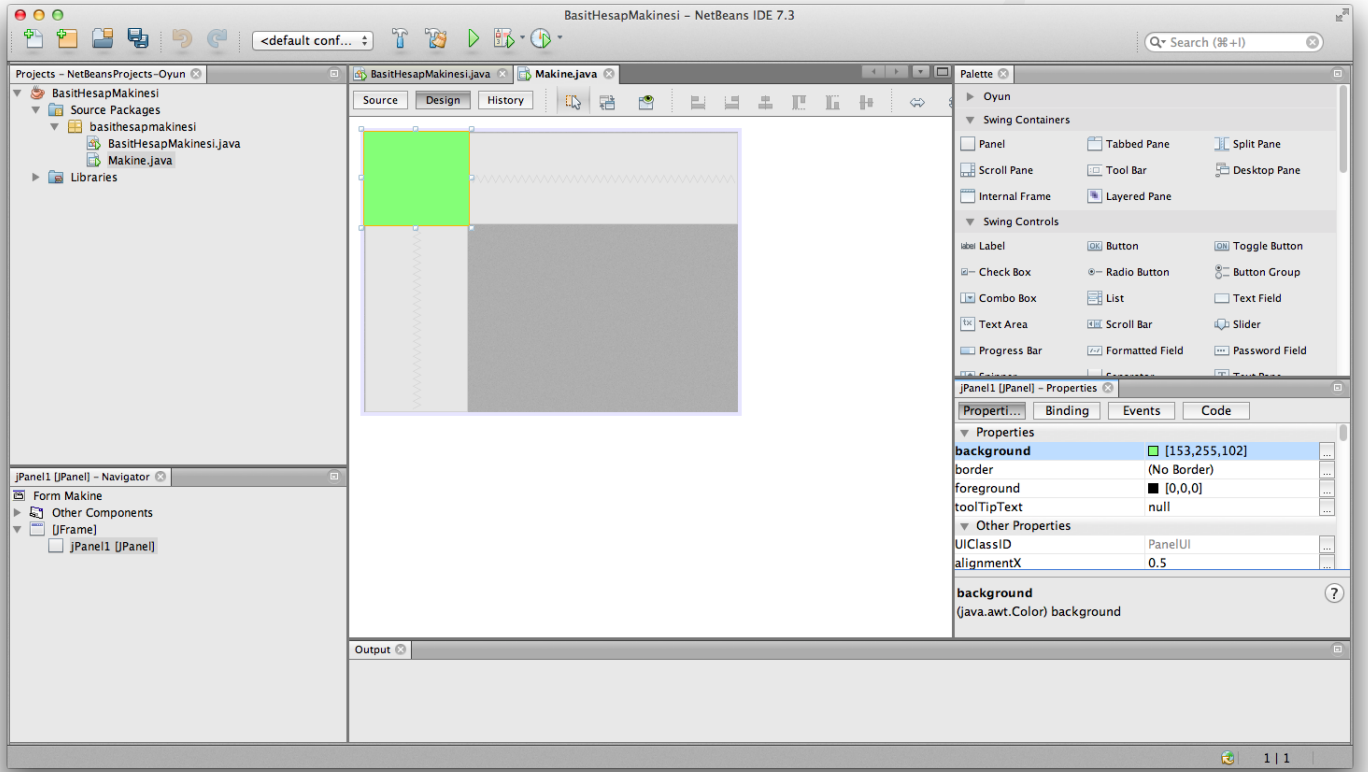


Şekil 1.8: Sağ taraftaki, "Palette" penceresinin altındaki "Swing Container" sekmesinde, yukarı-sol köşede bulunan JPanel'i sürükleyerek gri alana, yani sınıfımıza ekleyelim.

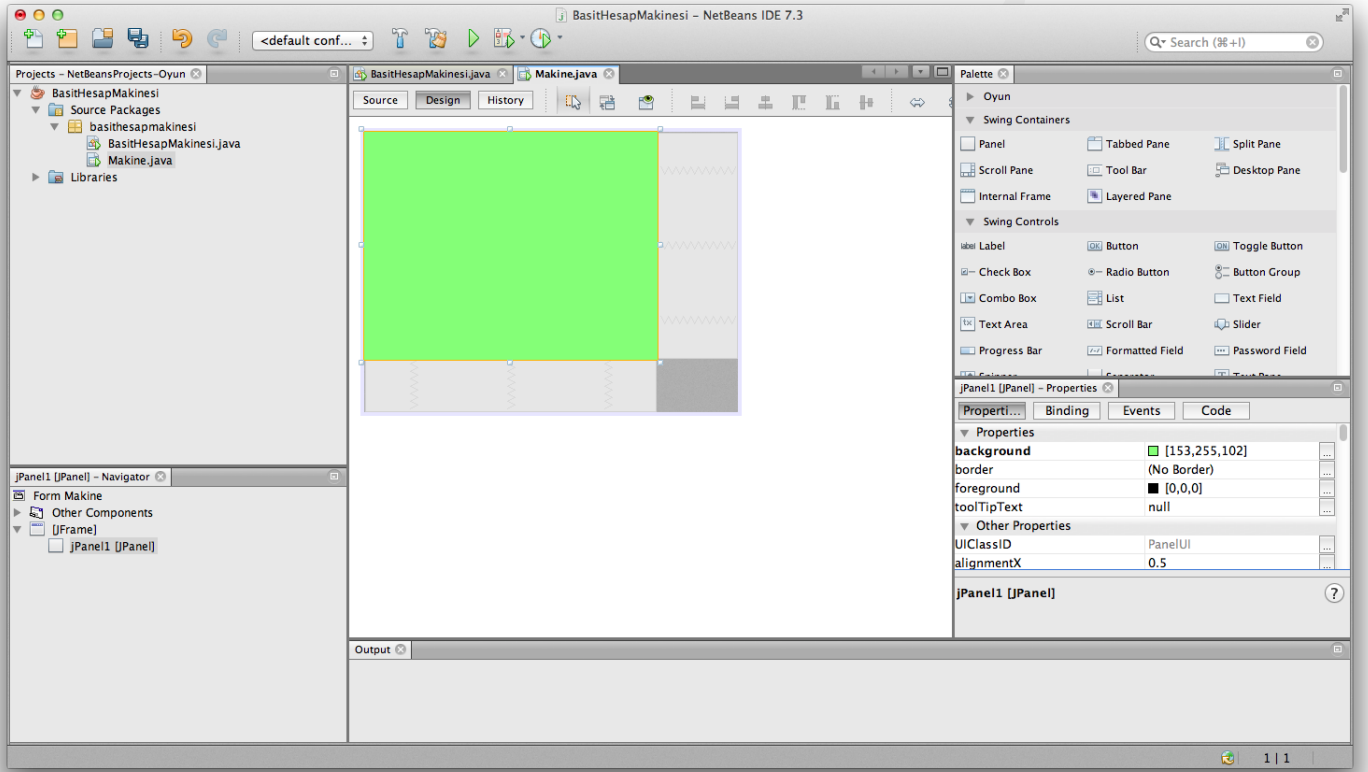
Şu an JPanel, resimde seçili görünüyor. Gri form alanında, turuncu kare çerçeve içinde görebilirsiniz. JPanel bu şekilde seçiliyken, sağ tarafta, altta Properties sekmesi ve onu altında "background" yazısını görebilirsiniz. Devamında, aynı satırda, rakamlar (238,238,238) ve onun da devamında bir kutucuk göreceksiniz. O kutucuğa tıklayarak, arka plan resmini değiştirebilirsiniz.



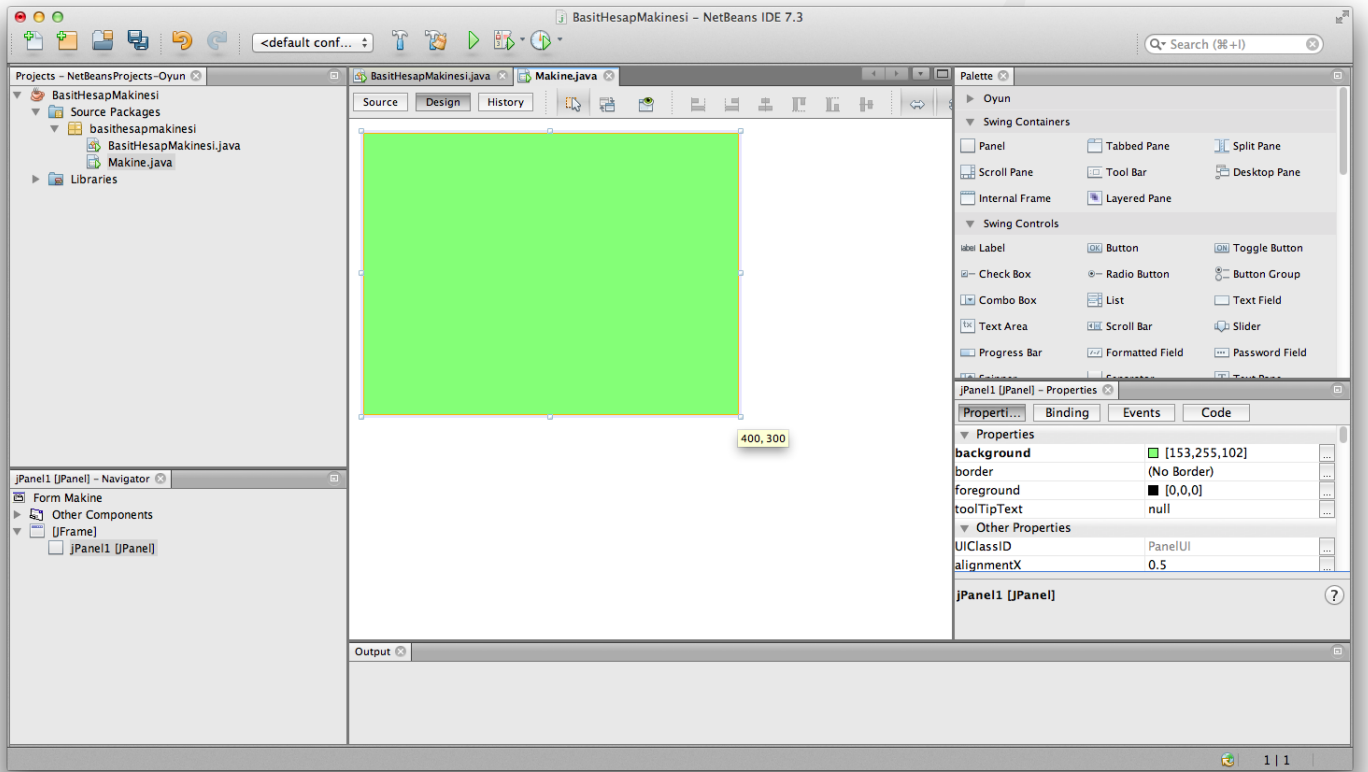
Şekil 1.9: Burdan herhangi bir rengi seçebilirsiniz. Dilerseniz RGB'sekmesine tıklayarak kendi renginizi de oluşturabilirsiniz.



Şekil 1.10: Biz yeşil rengi seçtik. JPanelimiz daha belirgin oldu. Şimdi de, JPanelin boyutunu, fareyle sağ-alt köşesinden tutarak uzatalım.



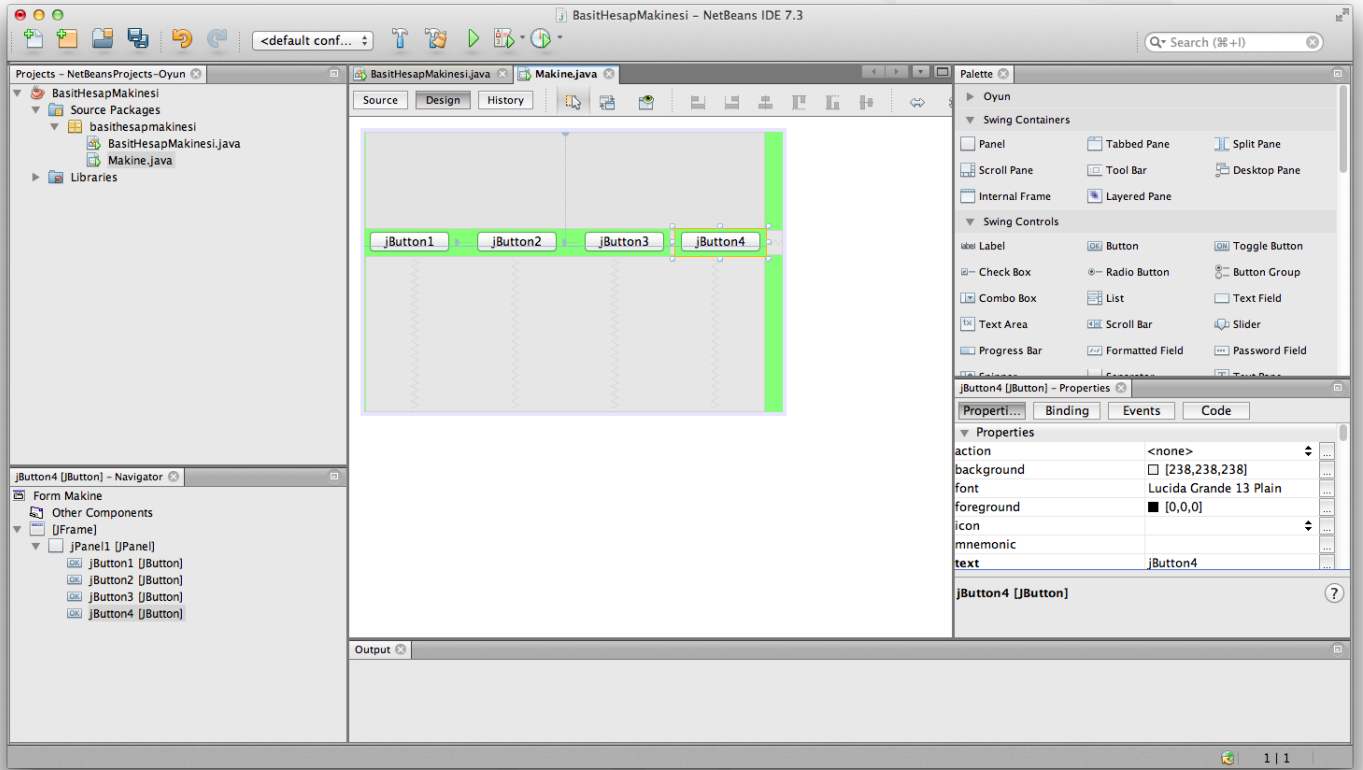
Şekil 1.11: JPanelin boyutunu, fareyle sağ-alt köşesinden tutarak uzatalım.



Şekil 1.12: Biraz daha uzatıp, JFrame Form alanını kapsayacak hale getirelim.

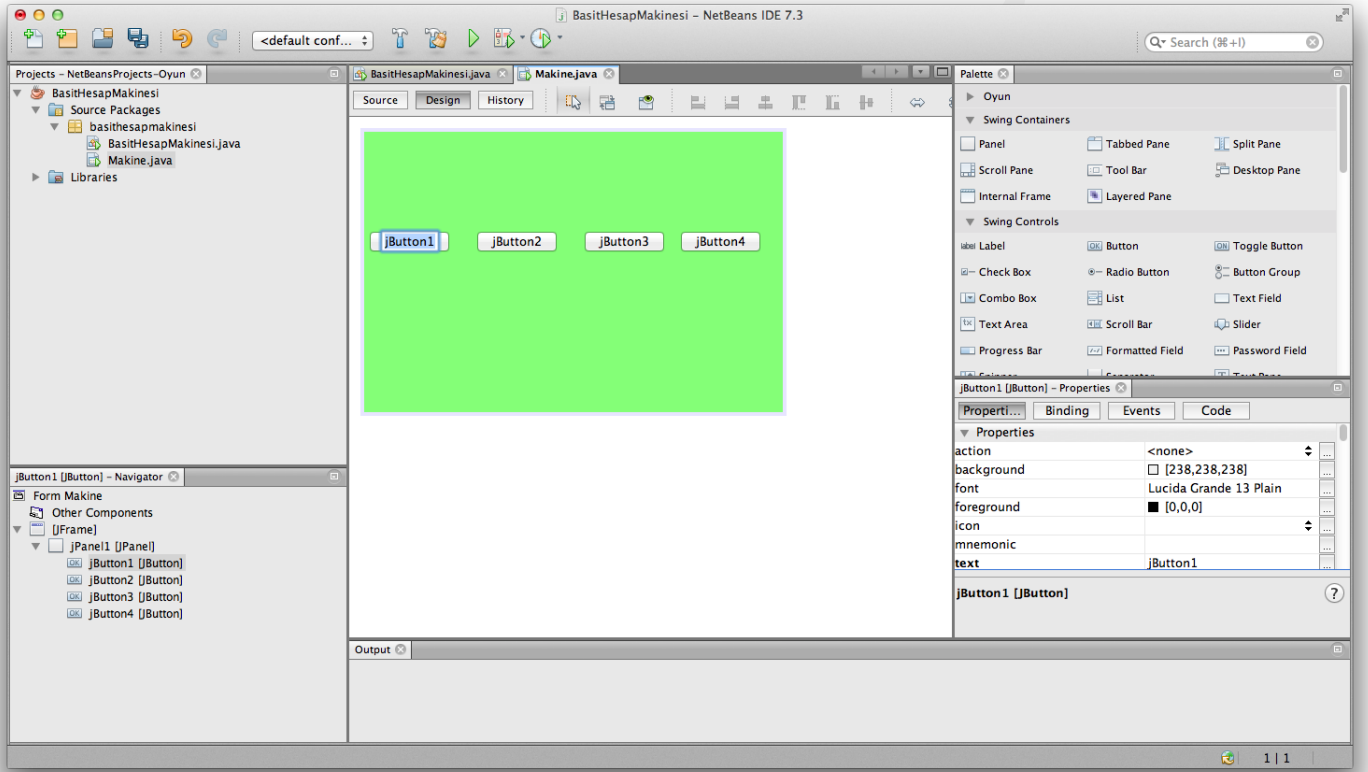
### 1.1.5 JButton Ekleme

Sağ tarafta, Palette penceresinin içinde bulunan, Swing Controls sekmesinde, ilk satırda ikinci sırada, Buton bulunmaktadır. Fareyle Buton'u tutup, Form alanına sürükleyelim. Arka arkaya 4 buton ekleyip, adlarını +, -, \*, / yapalım.

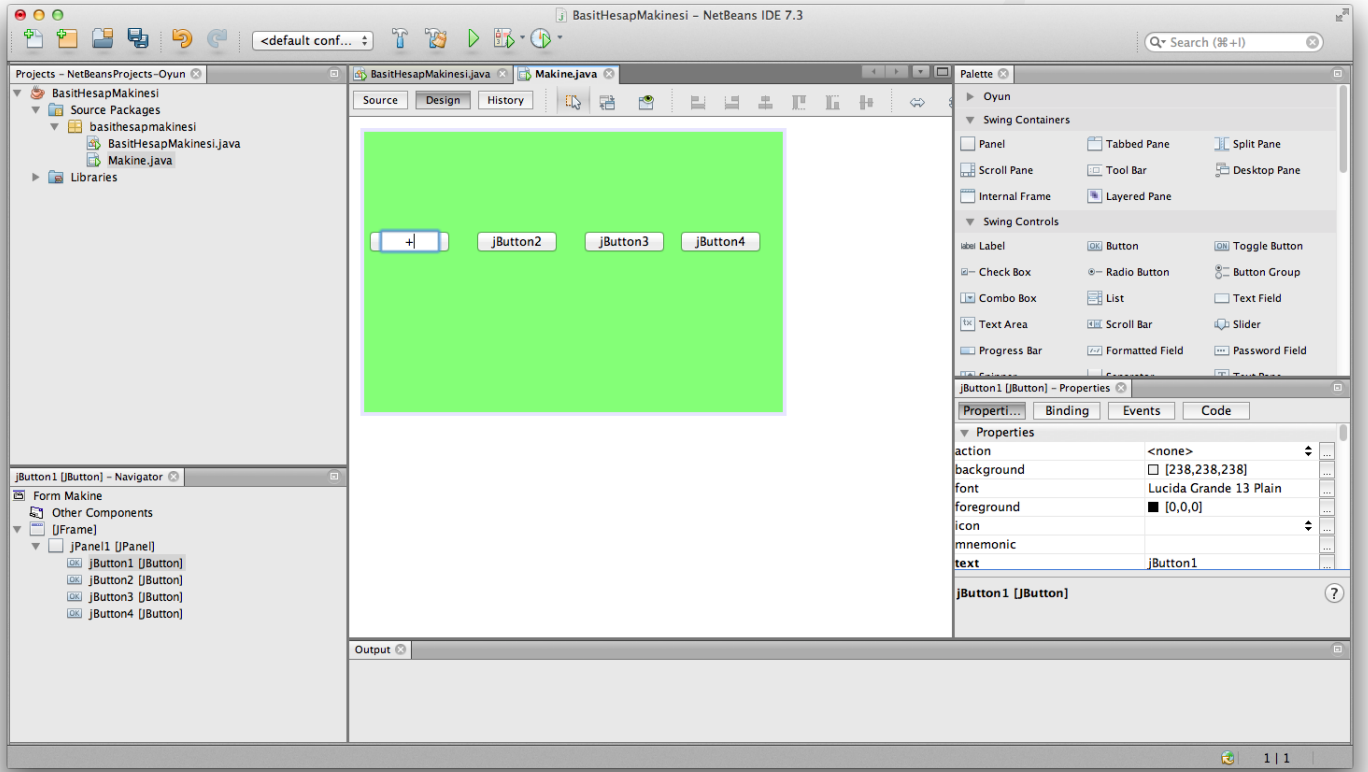


Şekil 1.13: Eklenen butonların üzerinde yazan jButton1, jButton2, jButton3 ve jButton4 yazılarını tek tek değiştirip sırasıyla +, -, \*, / yapalım.

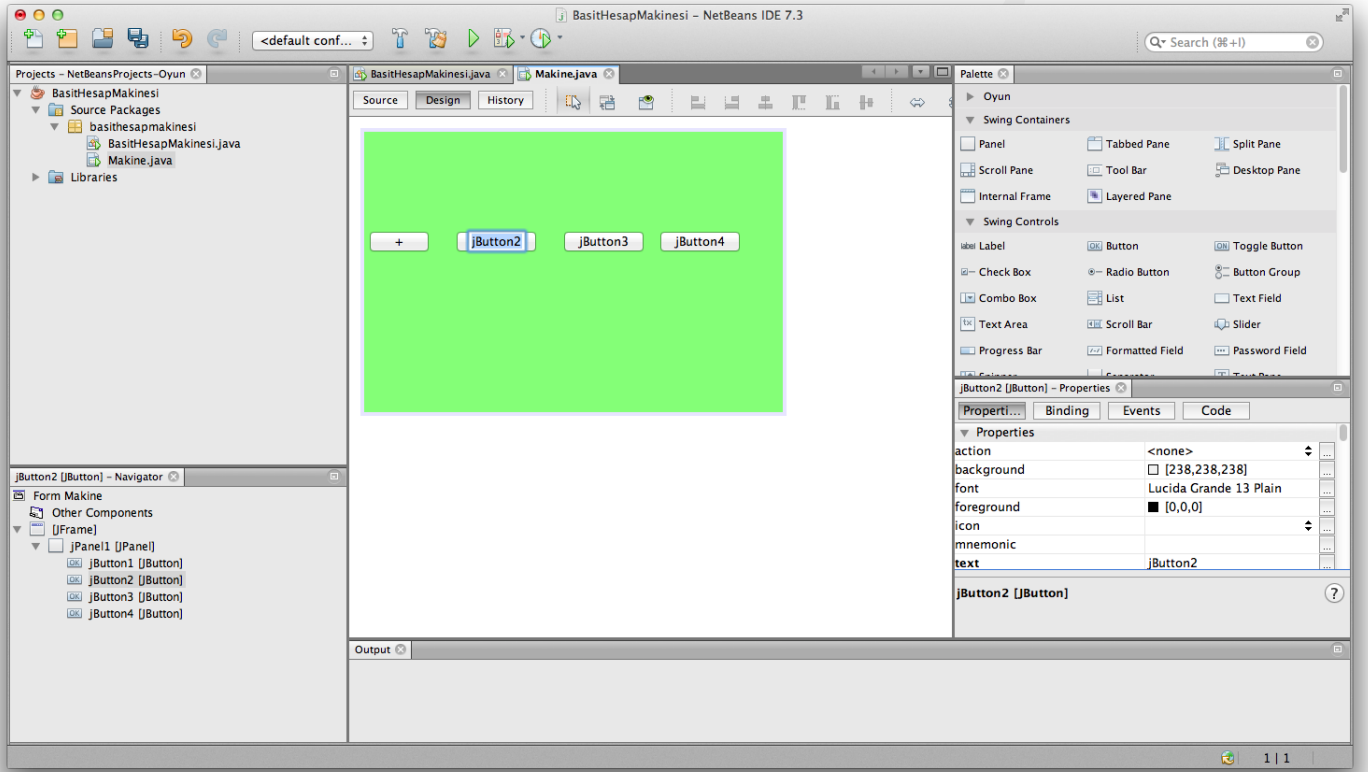




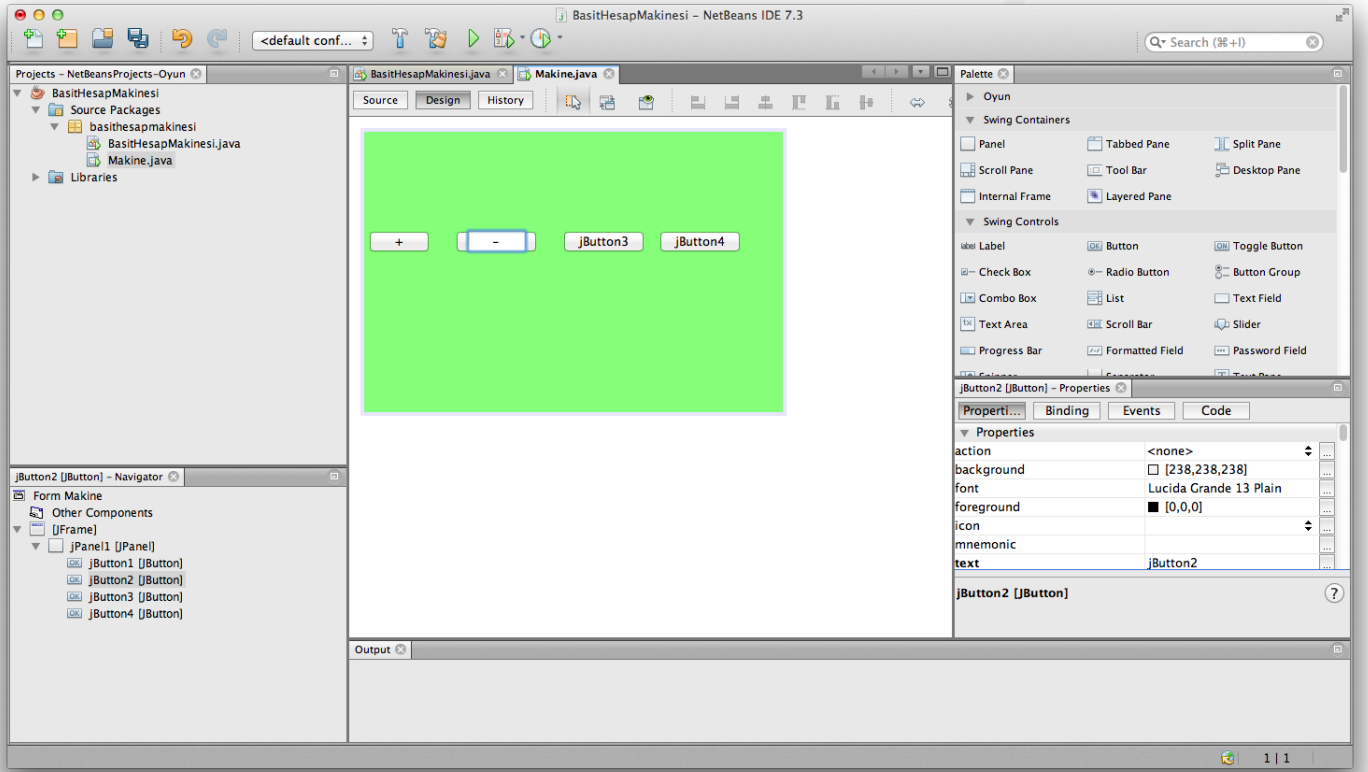
Şekil 1.14: jButton1 üzerine fareyle bir kez tıklandığında, yazının değişebileceğini fark etmişsinizdir.



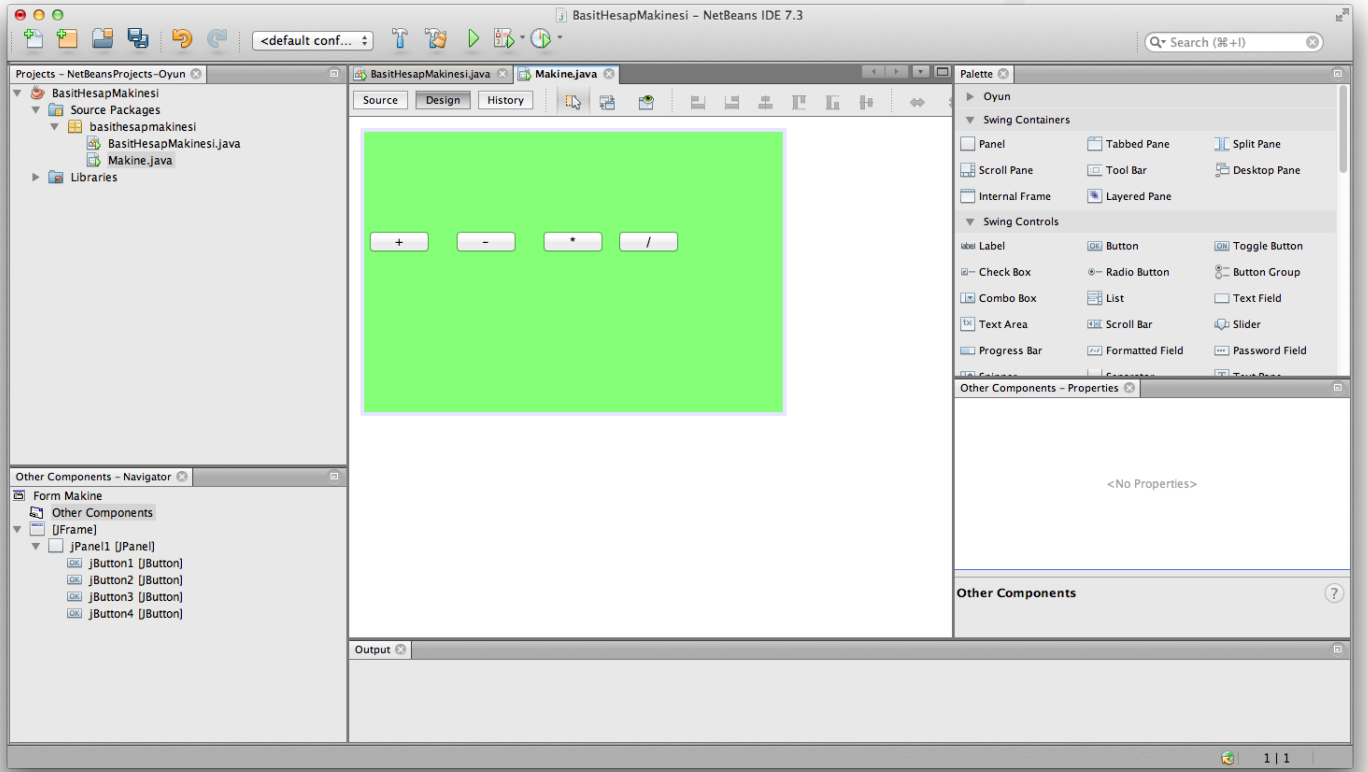
Şekil 1.15: jButton1 yazısını '+'ya dönüştürelim.



Şekil 1.16: jButton2 üzerine fareyle bir kez tıklayalım...



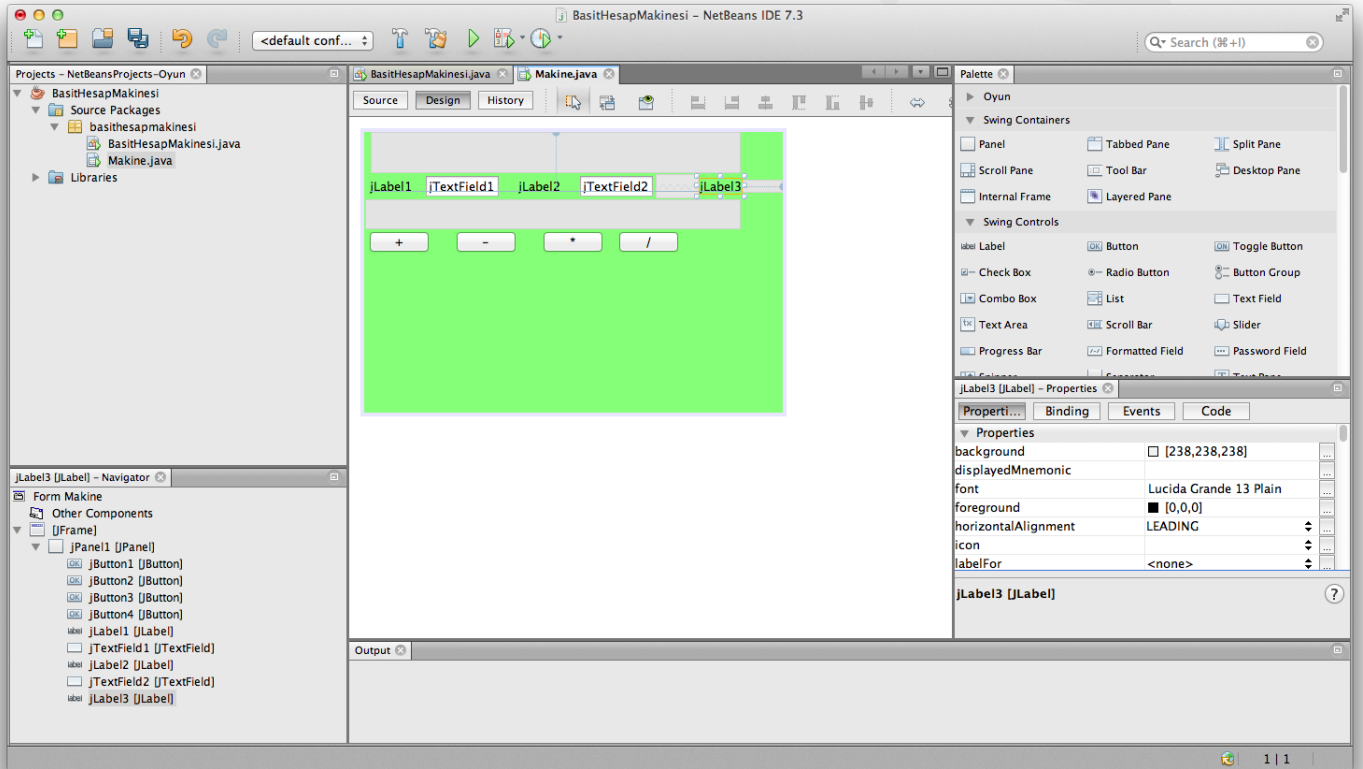
Şekil 1.17: Üzerindeki yazıyı – ile değiştirelim.



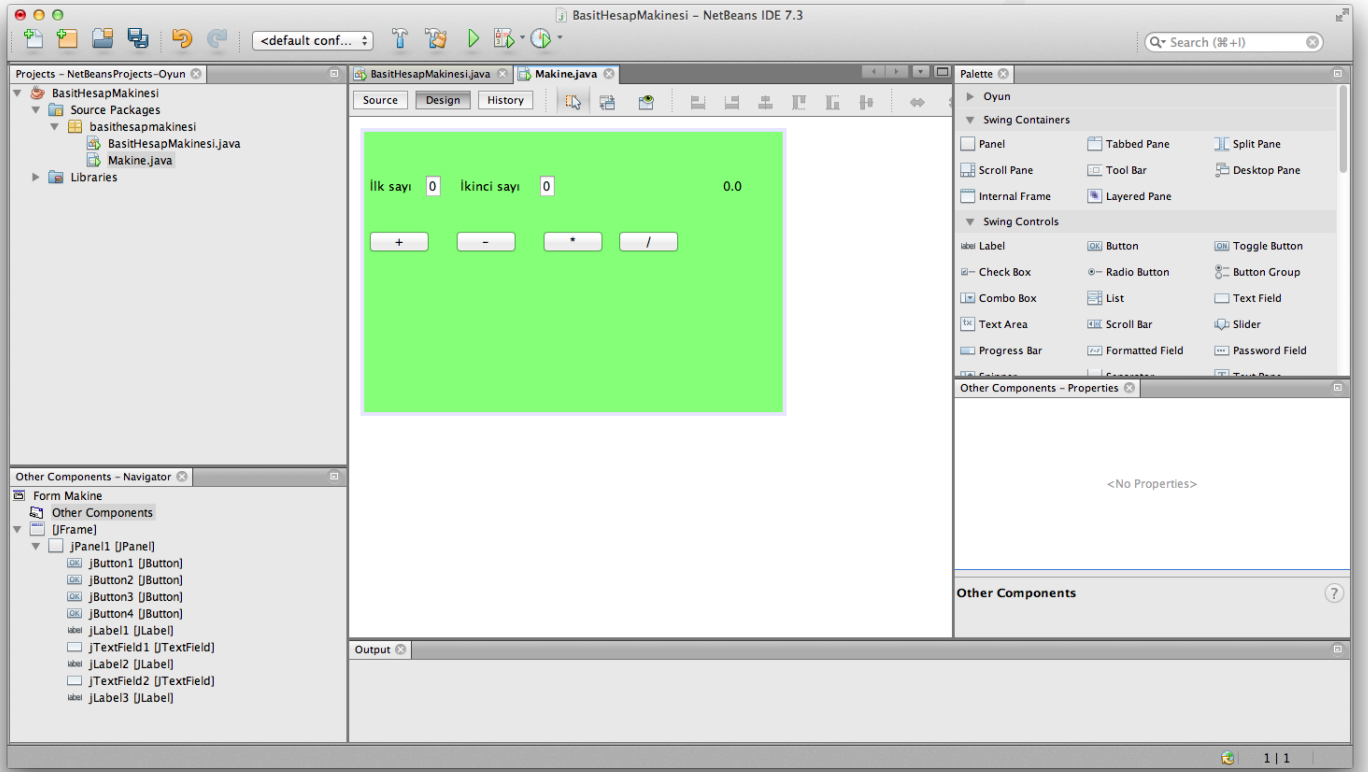
Şekil 1.18: 4 butonun da adlarını +, -, \*, / yaptıktan sonra, 3 adet JLabel ve 2 adet JTextField'i programımıza ekleyeceğiz.

### 1.1.6 jLabel ve jTextField Ekleme

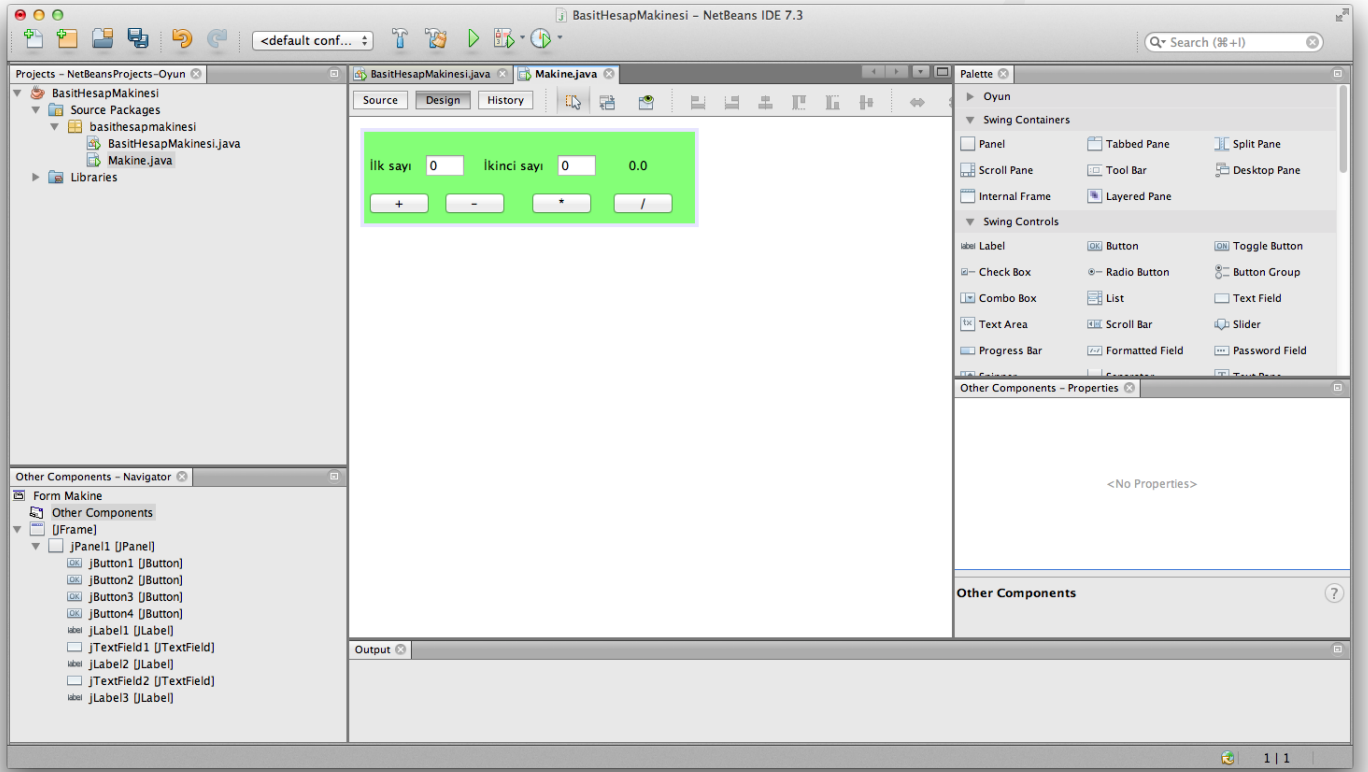
Sağ tarafta, Palette penceresinin içinde bulunan, Swing Controls sekmesinde, ilk satırda "Label" 3.satırda ise "TextField" bulunmaktadır. Fareyle tutup, Form alanına sürükleyelim. 3 adet jLabel ve 2 adet jTextField'i programımıza ekleyeceğiz.



Şekil 1.19: Şimdi sıra, bu Label(etiket) ve TextField(Metin Girdi Alanı) üzerindeki yazıları değiştirmeye geldi. Bunun için fareyle üzerlerine gelip, bir kez tıklayıp, istediğimiz yazının üzerlerine yazılmasını sağlayacağız.

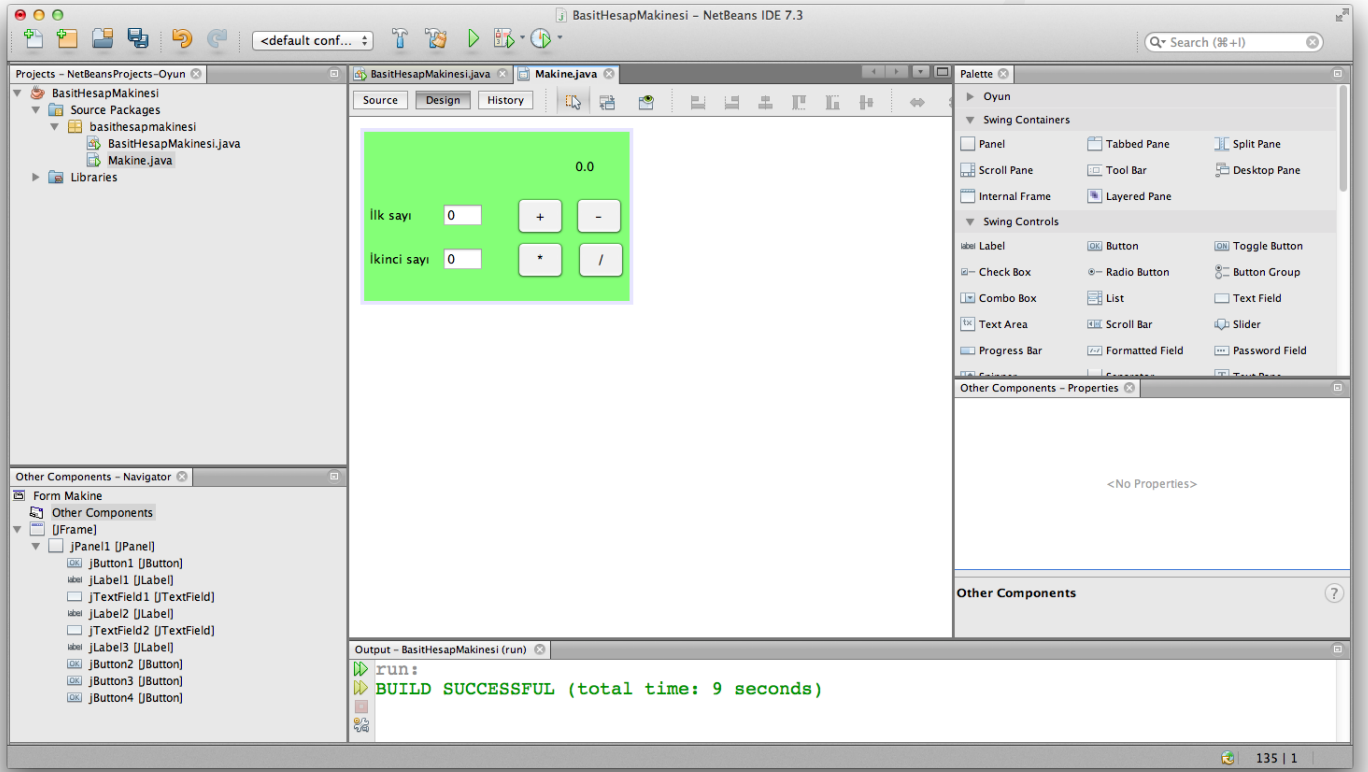


Şekil 1.20: İstenen yazıları yazdık, yalnız üzerine 0 yazdığımız TextField'ler biraz küçük oldu. Fareyle köşelerinden tutup programımızın görüntüsünü istediğimiz gibi değiştirelim. Hatta Form alanımızın boyutlarını da biraz küçültelim.

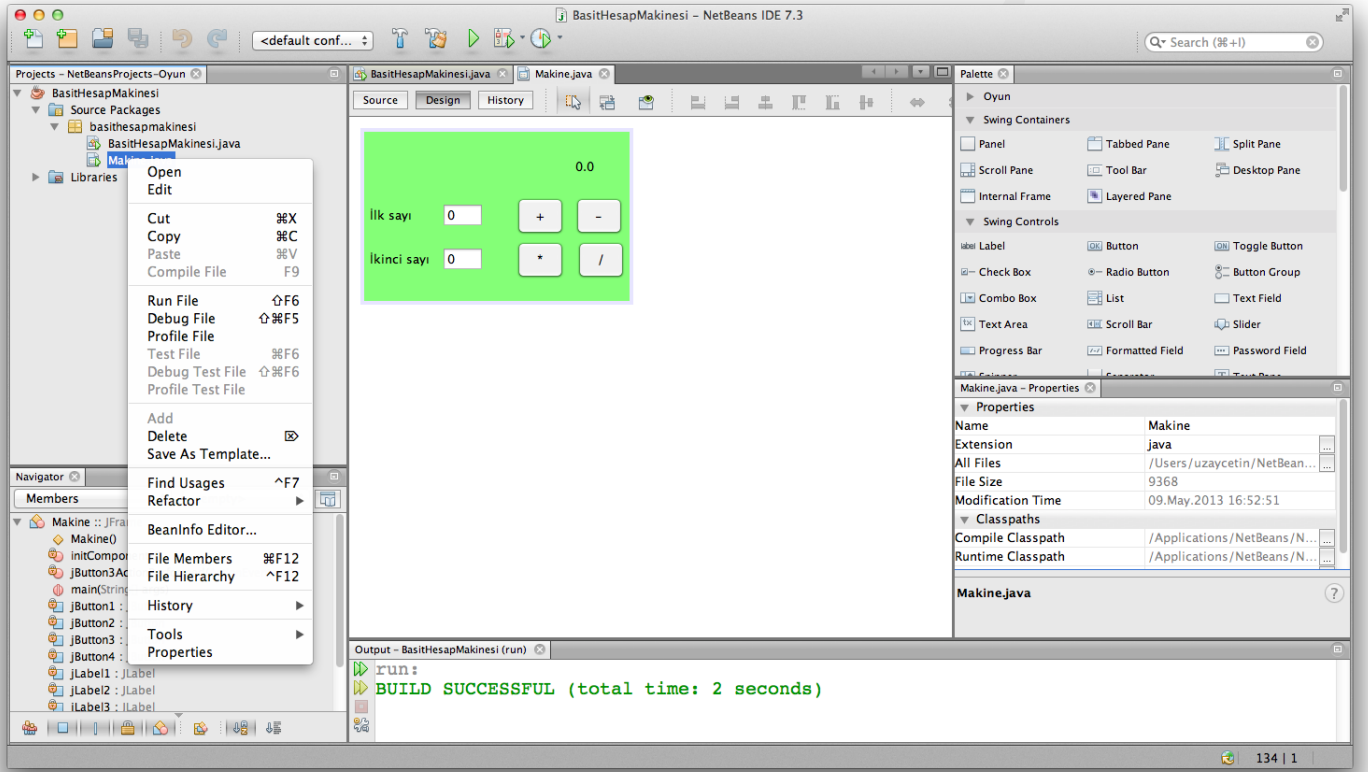


Şekil 1.21: Program tasarımı ile istediğiniz gibi oynamanız mümkün.

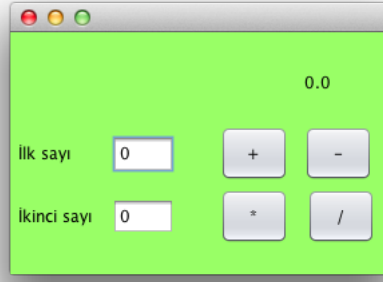




Şekil 1.22: Programımızın tasarımını bitirdik.



Şekil 1.23: Programımızı çalıştıralım. Soldaki Project penceresi içinde bulunan, "Makine" sınıfının üzerine gelip sağ tıklayalım ve "Run File" diyelim.



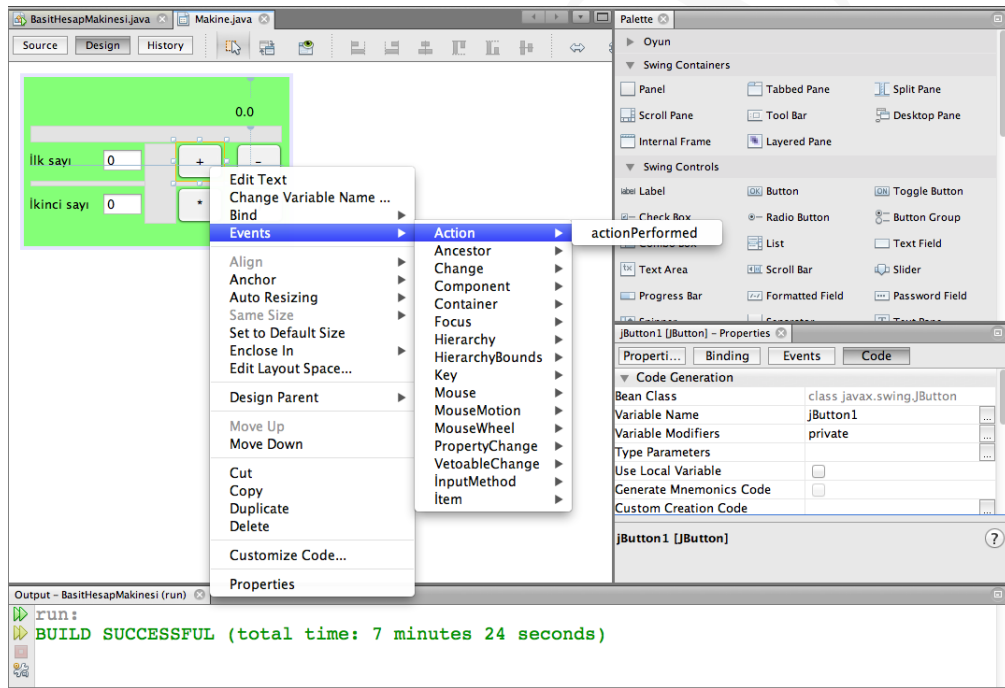
Şekil 1.24: Programımızı çalıştı, fakat bir şeyler eksik. Butona basıldığında olay agılama gerçekleşmiyor.

## 1.2 Palette yardımıyla Olay algılamak

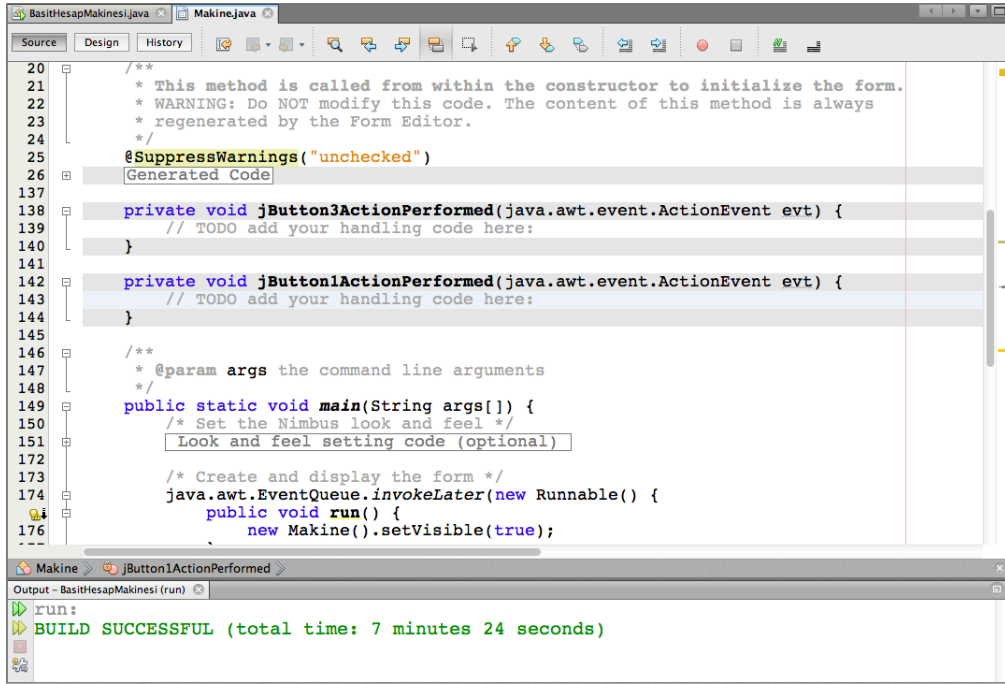
Bu noktadan sonra, tasarımın palet yardımıyla tamamlanmasından sonra, yavaş yavaş kod yazmaya başlayacağız. Ama Netbeans Palette bize hala daha yardım edecek.

### 1.2.1 Butona basıldığında olay algılamak

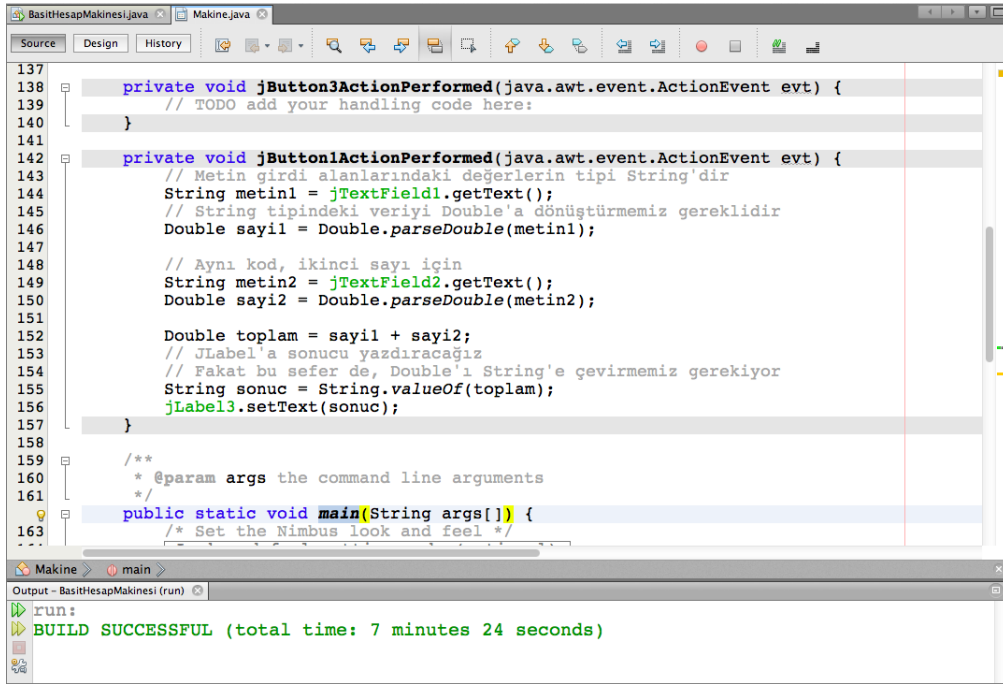
Bunun için tek yapmamız gereken, ilgili butonun üzerindeyken fareyle sağ tıklayıp, sırasıyla Events → Action → ActionPerformed seçeneklerin seçmek. Bu adımları izlediğimizde, Netbeans bizi Design(Tasarım)'dan Source(Kaynak) koda götürecektir. Burda ilgili olay algılama metodunun tam da içinde olacağız. Olay algılandığında ne yapılmasını istiyorsak, buraya yazmalıyız.



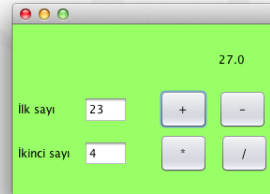
Şekil 1.25: Butona basıldığında olay algılama için, butona sağ tıklayıp, sırasıyla Events → Action → ActionPerformed seçeneklerin seçmeliyiz.



Şekil 1.26: Netbeans bizi jButton1ActionPerformed adlı metodun içine yönlendirdi.



Şekil 1.27: jButton1 butonu, bizim + butonumuz. Bu butona basıldığında, jTextField1 ve jTextField2 metin girdi alanlarına girilen, ilk ve ikinci sayıları alıp, bu sayıların toplamını jLabel3 adlı etikete yazmalıyız. Bununla ilgili kodu yazacağımız yer jButton1ActionPerformed adlı metodun içidir.



Şekil 1.28: jButton1(+) butonu, için kod çalışmaktadır. Diğer butonlar için de çok benzer olan kodları programımıza benzer şekilde eklemeliyiz: "ilgili butonun üzerindeki fareyle sağ tıklayıp, sırasıyla Events → Action → ActionPerformed seçeneklerini seçip, Netbeans'in bizi yönlendirdiği jButtonActionPerformed metodunun içine gerekli kodu eklemeliyiz."

```

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
    // EKSI BUTONU
    // Metin girdi alanlarındaki değerlerin tipi String'dir
    String metin1 = jTextField1.getText();
    // String tipindeki veriyi Double'a dönüştürmemiz gereklidir
    Double sayi1 = Double.parseDouble(metin1);

    // Aynı kod, ikinci sayı için
    String metin2 = jTextField2.getText();
    Double sayi2 = Double.parseDouble(metin2);

    Double fark = sayi1 - sayi2;
    // JLabel'a sonucu yazdıracağız
    // Fakat bu sefer de, Double'ı String'e çevirmemiz gerekiyor
    String sonuc = String.valueOf(fark);
    jLabel3.setText(sonuc);
}

```

Şekil 1.29: jButton2( - ) butonu içim eklenen kod.

```

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
    // CARPIM BUTONU
    // Metin girdi alanlarındaki değerlerin tipi String'dir
    String metin1 = jTextField1.getText();
    // String tipindeki veriyi Double'a dönüştürmemiz gereklidir
    Double sayi1 = Double.parseDouble(metin1);

    // Aynı kod, ikinci sayı için
    String metin2 = jTextField2.getText();
    Double sayi2 = Double.parseDouble(metin2);

    Double carpim = sayi1 * sayi2;
    // JLabel'a sonucu yazdıracağız
    // Fakat bu sefer de, Double'ı String'e çevirmemiz gerekiyor
    String sonuc = String.valueOf(carpim);
    jLabel3.setText(sonuc);
}

```

Şekil 1.30: jButton3( \* ) butonu içim eklenen kod.

```

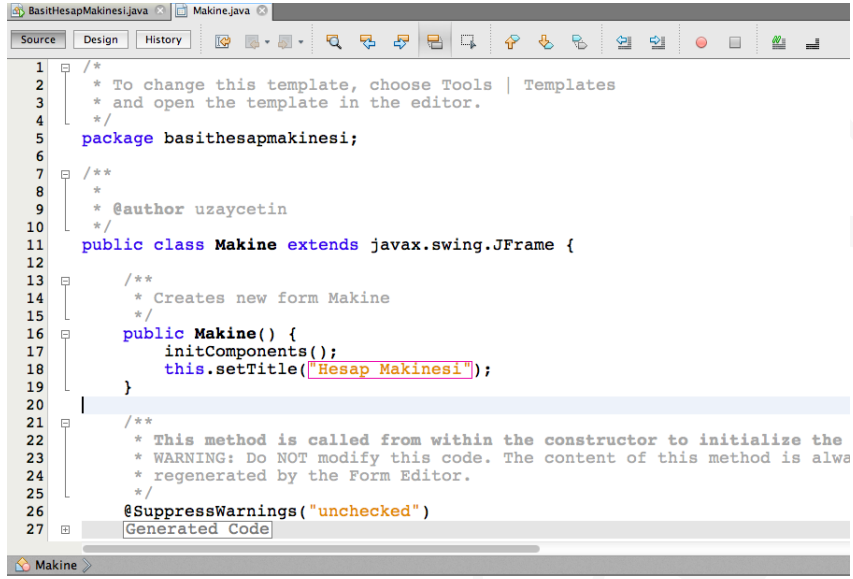
private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {
    // BOLU BUTONU
    // Metin girdi alanlarındaki değerlerin tipi String'dir
    String metin1 = jTextField1.getText();
    // String tipindeki veriyi Double'a dönüştürmemiz gereklidir
    Double sayi1 = Double.parseDouble(metin1);

    // Aynı kod, ikinci sayı için
    String metin2 = jTextField2.getText();
    Double sayi2 = Double.parseDouble(metin2);

    Double bolum = 0.0;
    if(sayi2 != 0)
        bolum = sayi1 / sayi2;
    // JLabel'a sonucu yazdıracağız
    // Fakat bu sefer de, Double'ı String'e çevirmemiz gerekiyor
    String sonuc = String.valueOf(bolum);
    jLabel3.setText(sonuc);
}

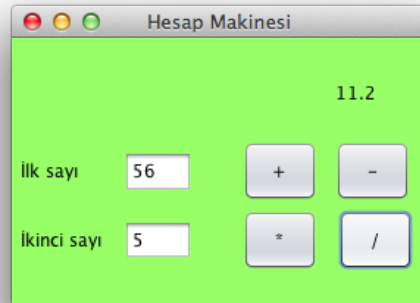
```

Şekil 1.31: jButton4( / ) butonu içim eklenen kod. Tek farkı, sayi2'nin sıfır olduğu durumu kontrol etmesidir.



```
1  /**
2   * To change this template, choose Tools | Templates
3   * and open the template in the editor.
4   */
5   package basithesapmakinesi;
6
7   /**
8    *
9    * @author uzaycetin
10   */
11   public class Makine extends javax.swing.JFrame {
12
13       /**
14        * Creates new form Makine
15        */
16       public Makine() {
17           initComponents();
18           this.setTitle("Hesap Makinesi");
19       }
20
21       /**
22        * This method is called from within the constructor to initialize the
23        * WARNING: Do NOT modify this code. The content of this method is always
24        * regenerated by the Form Editor.
25        */
26       @SuppressWarnings("unchecked")
27       Generated Code
```

Şekil 1.32: Son olarak pencremizin başlığını, `this.setTitle()` komutuyla, Makine adlı kurucu fonksiyonun içinde yazalım.



Şekil 1.33: Projemizin son hali