

NOT HESAP

VizeNotu

FinalNotu

Geçme Notu

Ağırlama

Hesapla

NOT HESAP

The image shows a Java Swing window titled "NOT HESAP" with a graphical user interface. The window contains three text input fields labeled "VizeNotu", "FinalNotu", and "Gecme Notu", followed by a label "Açıklama" and a "Hesapla" button. A context menu is open over the "Hesapla" button, showing the following options:

- Edit Text
- Change Variable Name ...
- Bind
- Events** (selected)
- Align
- Anchor
- Auto Resizing
- Same Size
- Set to Default Size
- Enclose In
- Edit Layout Space...
- Design Parent
- Move Up

The "Events" menu is expanded, showing the following options:

- Action** (selected)
- Ancestor
- Change
- Component
- Container
- Focus
- Hierarchy
- HierarchyBounds
- Key
- Mouse
- MouseEvent

The "Action" menu is further expanded, showing the following options:

- actionPerformed [jButton1ActionPerformed]** (selected)

At the bottom of the IDE, a terminal window titled "tput - JavaApplication129 (run)" shows the following output:

```
run:  
BUILD SUCCESSFUL (total
```

NOT HESAP

```
double v;  
double f;  
double s;
```

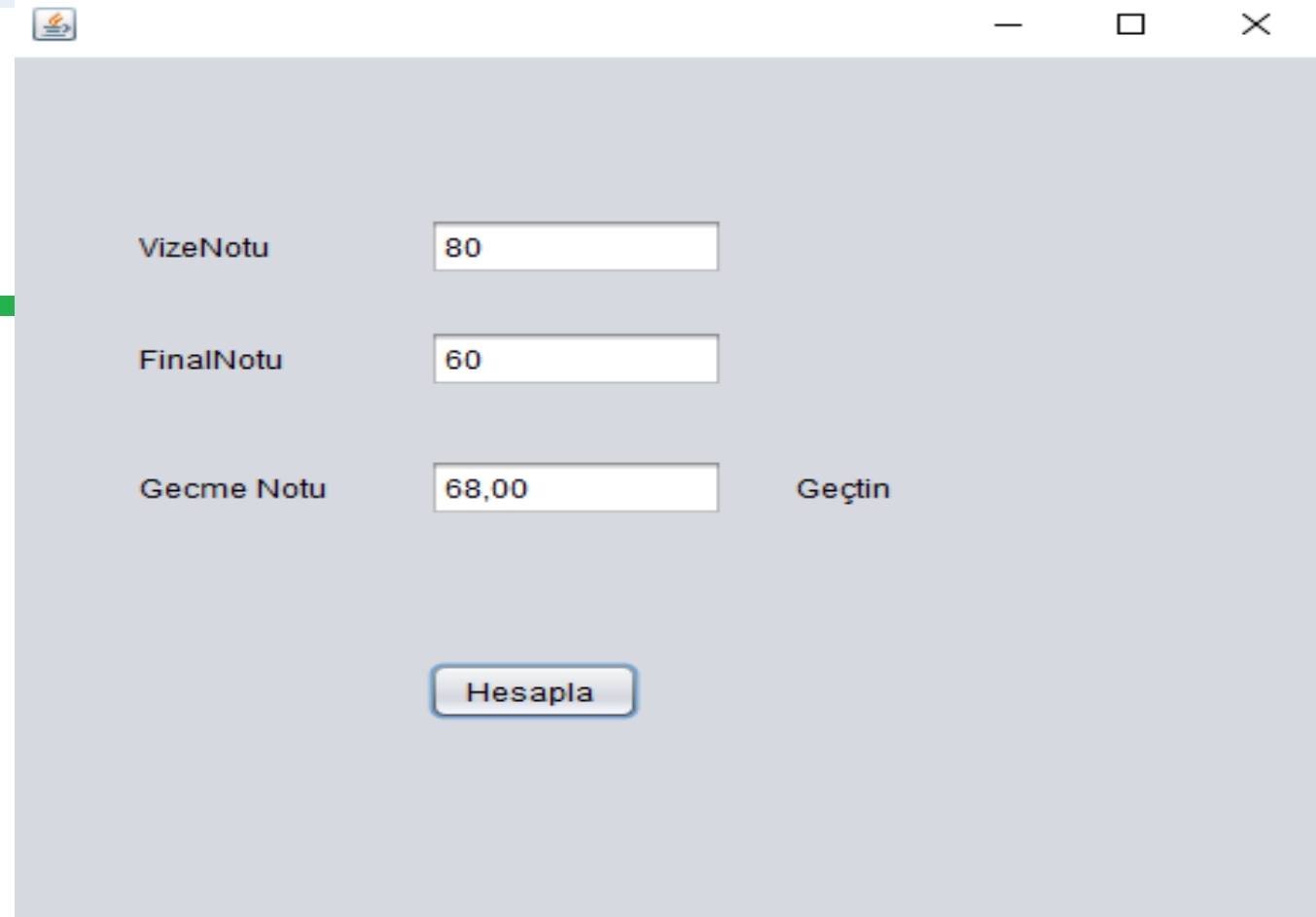
```
v = Double.parseDouble(vNotu.getText());  
f = Double.parseDouble(fNotu.getText());  
s = v*0.4 + f*0.6;
```

```
String sonuc = String.format("%.2f", s);
```

```
Sonuc.setText(sonuc);
```

NOT HESAP

```
if (s <60 && s >=0) {  
    jLabel14.setText ("kaldın");  
}else {  
    jLabel14.setText ("Geçtin");  
}
```



VizeNotu

FinalNotu

Gecme Notu Geçtin

DEĞİŞKEN DÖNÜŞÜMLERİ



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20

```
package javaapplication130;

public class JavaApplication130 {

    public static void main(String[] args) {

        String sayi = "26.75";

        System.out.println("sayı "+ sayi);

        double sayi1 = Double.parseDouble(sayi);

        System.out.println("Sayı : "+ (sayi1+20));

        // TODO code application logic here

    }
}
```

```
int sayi = 263;
String sayi1 = Integer.toString(sayi);
```

```
public static void main(String[] args) {

    String sayi = "265";

    System.out.println("sayı "+ sayi);

    int sayi1 = Integer.parseInt(sayi);

    System.out.println("Sayı : "+ (sayi1+20));
}
```

Output - JavaApplication130 (run)

```
run:
sayı 26.75
Sayı : 46.75
BUILD SUCCESSFUL (total time: ...)
```

Project Explorer showing file structure:

- NewJFrame2.java
- Test Packages
- Libraries
- Test Libraries
- JavaApplication 130
 - Source Packages
 - javaapplication 130
 - JavaApplication 130.java
- Test Packages
- Libraries
- Test Libraries
- JavaApplication 131
 - Source Packages
 - javaapplication 131
 - JavaApplication 131.java
 - NewJFrame.java

Metre [JRadioButton] - N... x

- Form NewJFrame
 - Other Components
 - buttonGroup1 [ButtonGroup]
 - [JFrame]
 - jPanel1 [JPanel]
 - AbsoluteLayout
 - Hesap [JButton]
 - Uzun [JTextField]
 - Genis [JTextField]
 - Topla [JTextField]
 - label jLabel1 [JLabel]
 - label jLabel2 [JLabel]
 - label jLabel3 [JLabel]
 - Metre [JRadioButton]
 - Desimetre [JRadioButt]
 - Santimetre [JRadioButt]

The GUI consists of a green rectangular area containing:

- Two text input fields labeled 'Uzunluk' and 'Genişlik'.
- A 'Hesapla' button.
- A vertical column of four radio buttons: 'Metrekare', 'Desimetre', 'Santimetre', and 'Milimetre'.

Output - JavaApplication131 (run) x

```
run:
BUILD SUCCESSFUL (total time: 1 minute 57 seconds)
```

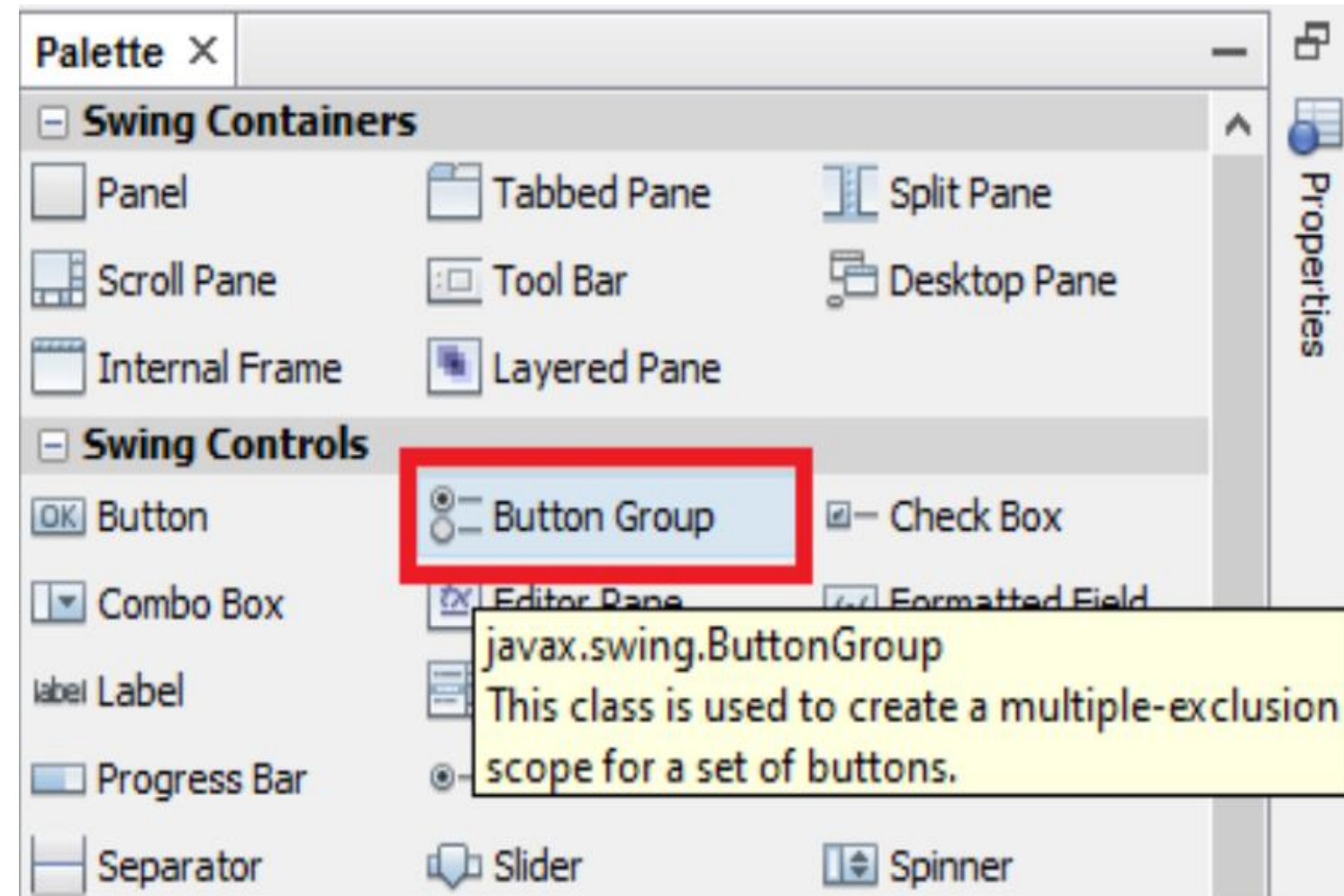
Metre [JRadioButton] - Properties

Properties	Binding	Events	Code
Accessible Description			...
Accessible Name			Metrekare
Accessible Parent			jPanel1
action			<none>
actionCommand			Metrekare
alignmentX			0.0
alignmentY			0.5
autoscrolls			<input type="checkbox"/>
background			<input type="checkbox"/> [240,240,240]
baselineResizeBehavior			CENTER_OFFSET
border			[CompoundBorderUIResource]
borderPainted			<input type="checkbox"/>
buttonGroup			buttonGroup1
componentPopupMenu			<none>
contentAreaFilled			buttonGroup1
cursor			Default Cursor
debugGraphicsOptions			NO_CHANGES
disabledIcon			<none>
disabledSelectedIcon			<none>
displayedMnemonicIndex			-1
doubleBuffered			<input type="checkbox"/>
enabled			<input checked="" type="checkbox"/>
...			...

buttonGroup
Group of buttons in which this button belongs

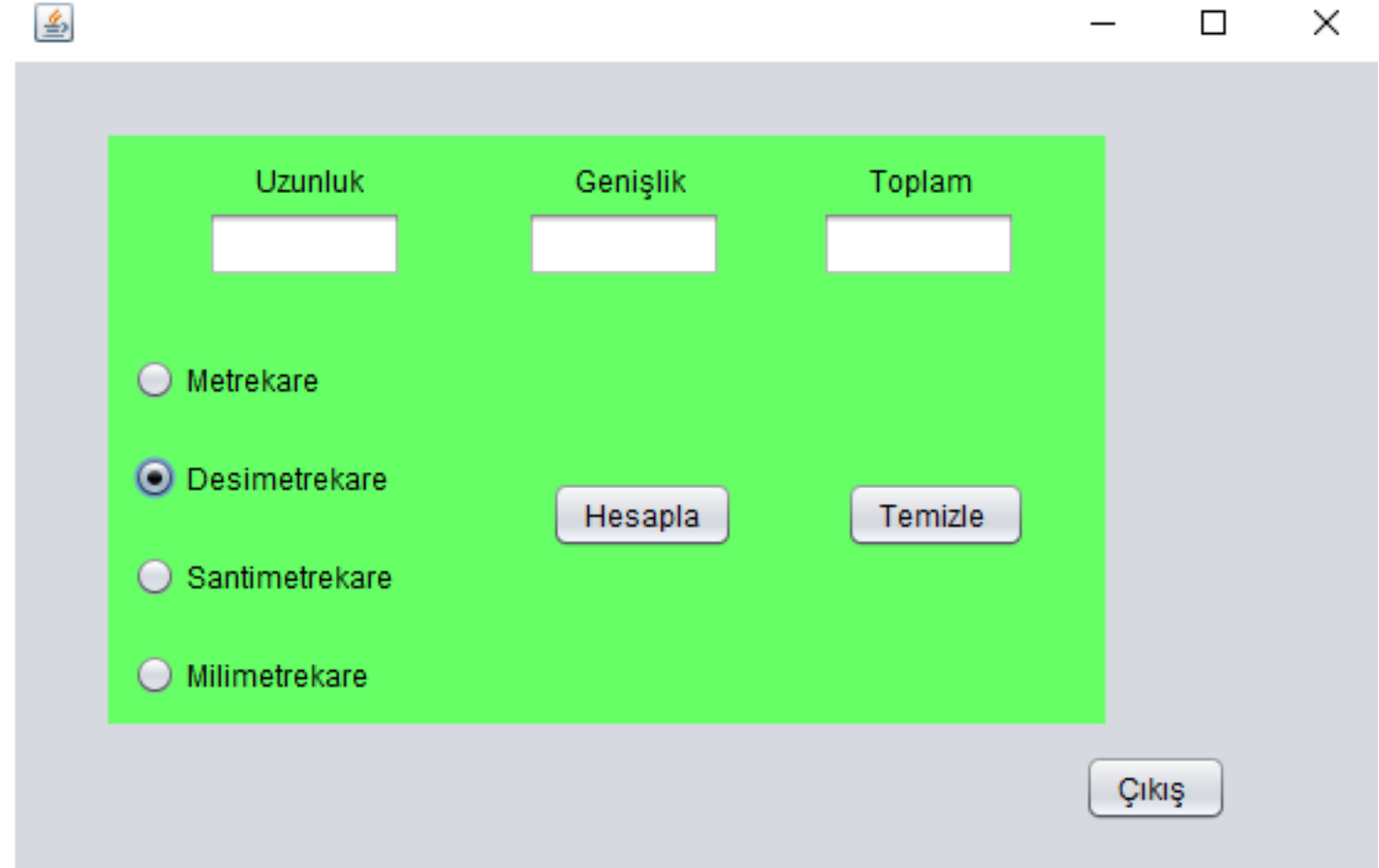
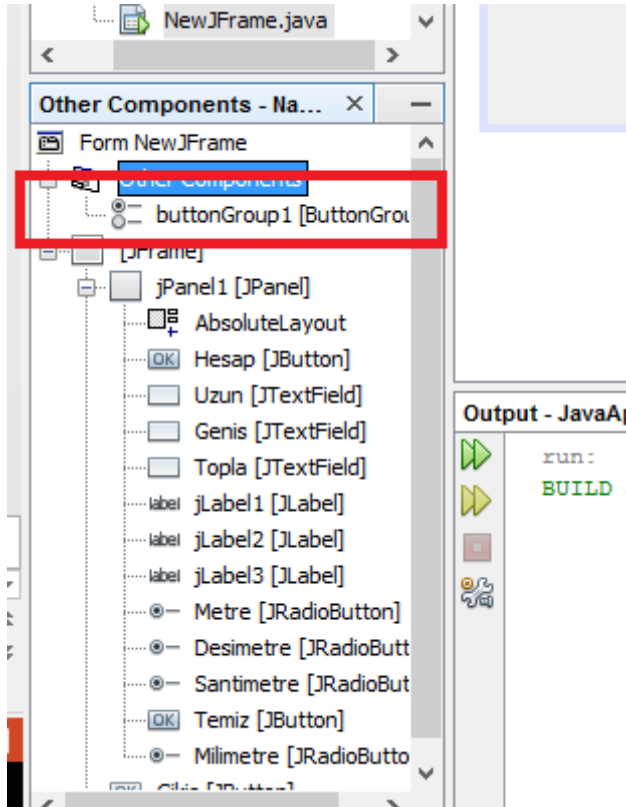
Close Help

RADIO BUTTON



*****Button Group
yapılmazsa radio
düğmelerinin hepsi
aynı seçilir !!!!

*****ButtonGroup
sürüklenerek ekrana
bırakılır tasarım
alanında görülmez
kütükte görülür !!!!



PENCERE KAPAMA (EXIT)

```
}  
  
private void CikisActionPerformed(java.awt.event.ActionEvent evt) {  
  
    System.exit(0);  
  
    // TODO add your handling code here:  
  
}
```

Alan Hesabı

Uzunluk	Genişlik	Toplam
<input type="text"/>	<input type="text"/>	<input type="text"/>

Metrekare
 Desimetrekare
 Santimetrekare
 Milimetrekare

Hesapla Temizle

Çıkış

ÖRNEK 2: ALAN HESABI

- Uzunluk ve Geniřliđi verilen alanı, metrekare, desimetrekare, santimetrekare veya milimetrekare olarak hesaplayınız. Birimlerini yazdırınız.

1. Alan Hesabı Arayüz Tasarımı



Uzunluk Genişlik Toplam Alan Birim

 jLabel5

Metrekare

Desimetrekare

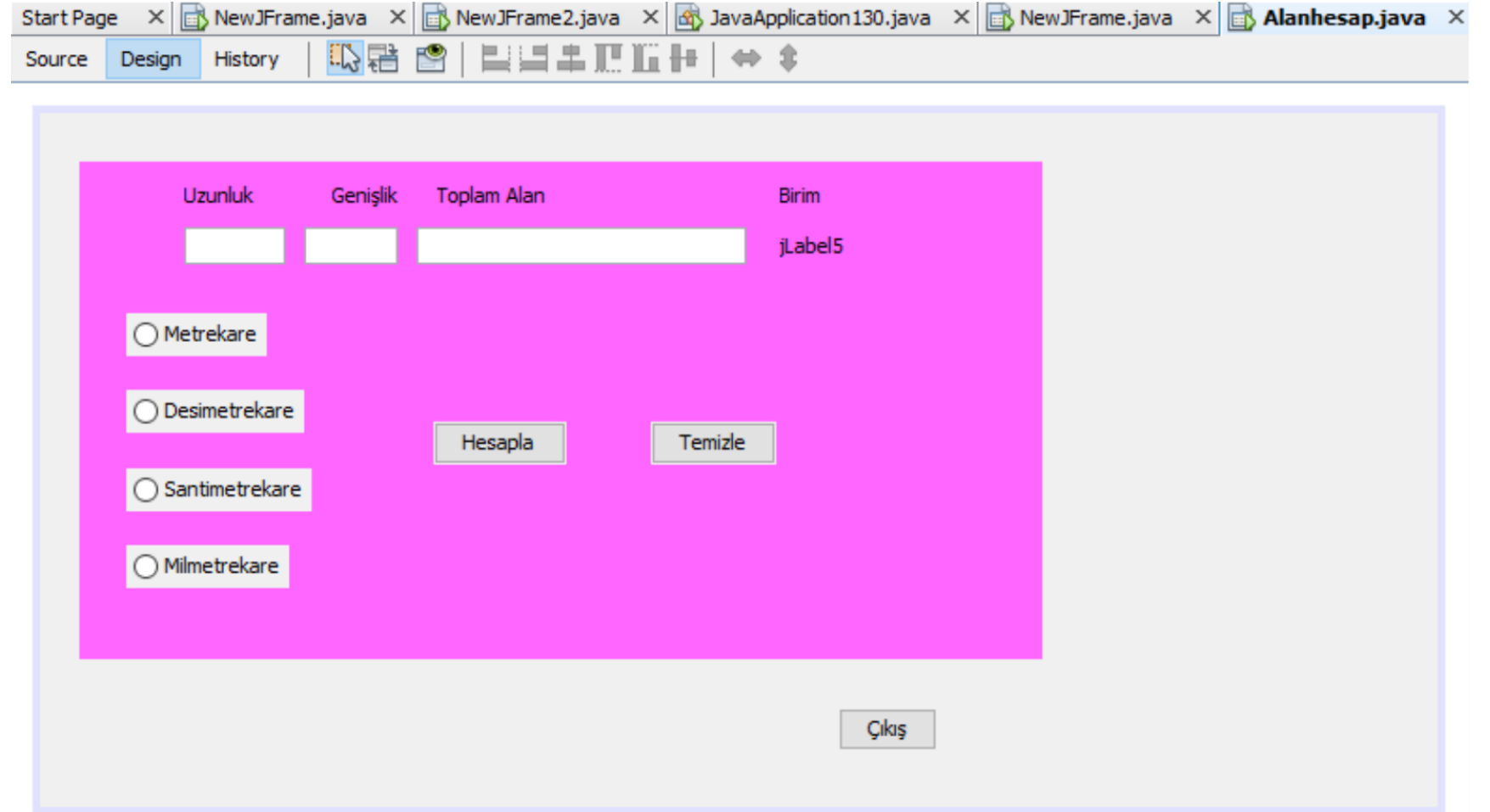
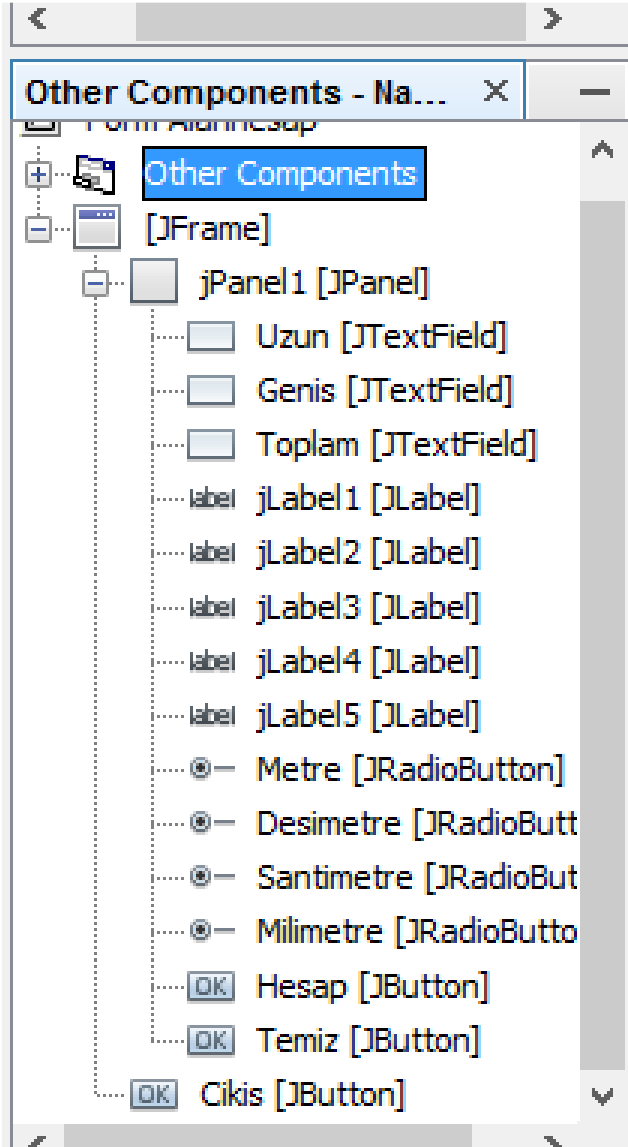
Santimetrekare

Milimetrekare

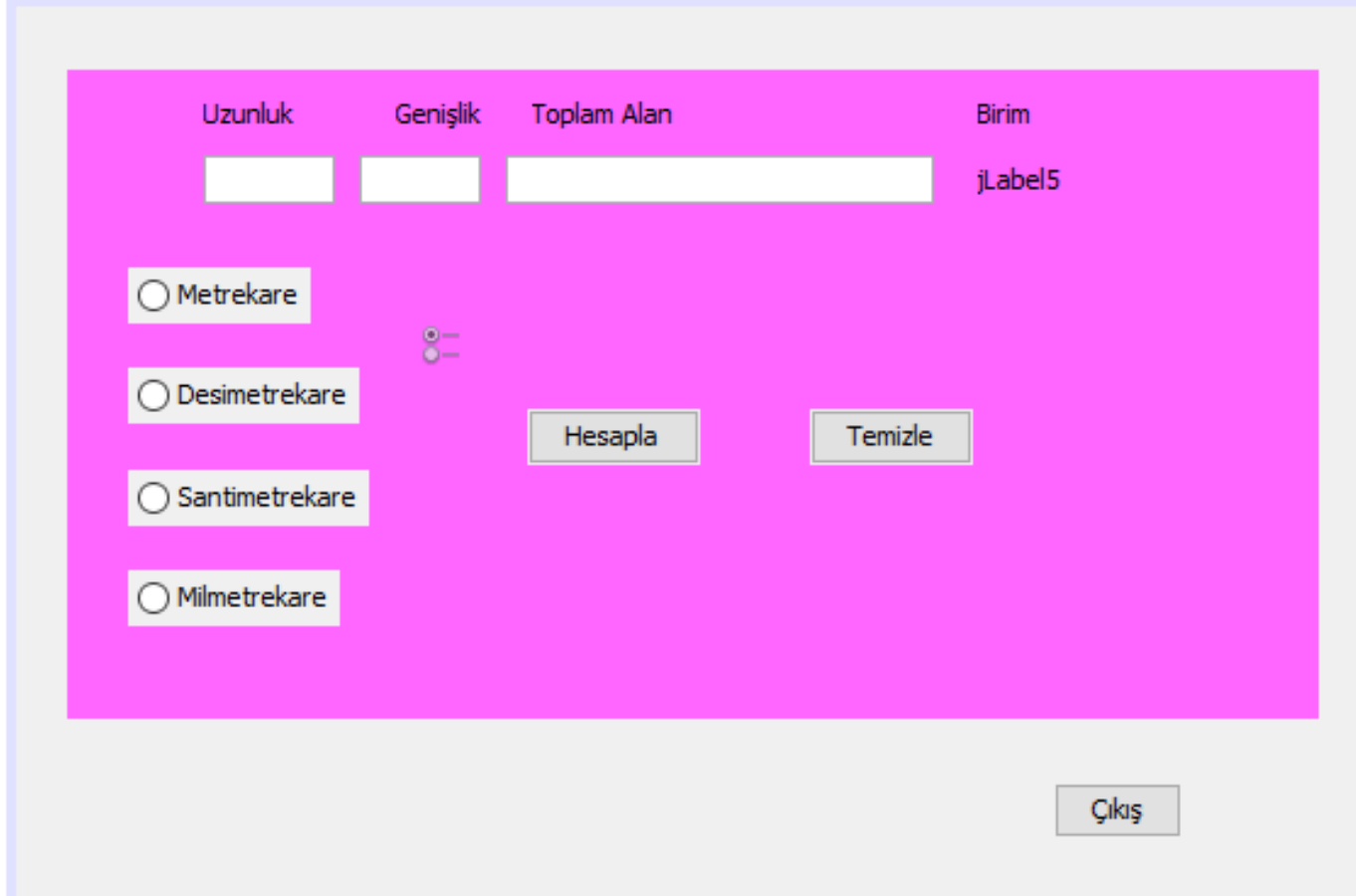
Hesapla Temizle

Çıkış

2. Komutların Adlandırılması



3. Button Group Oluştur



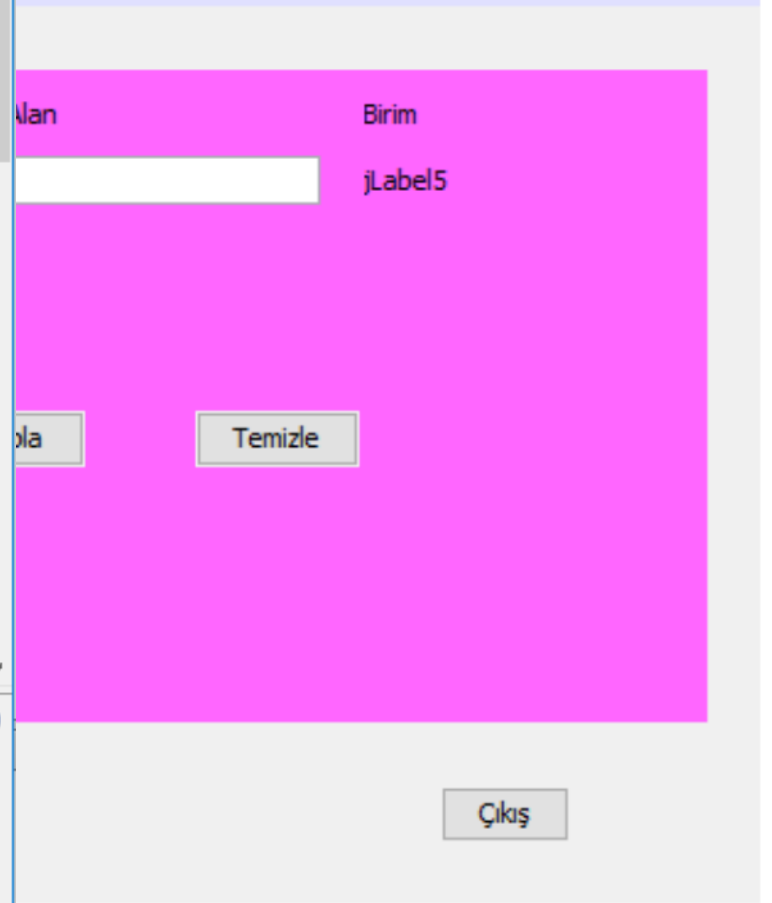
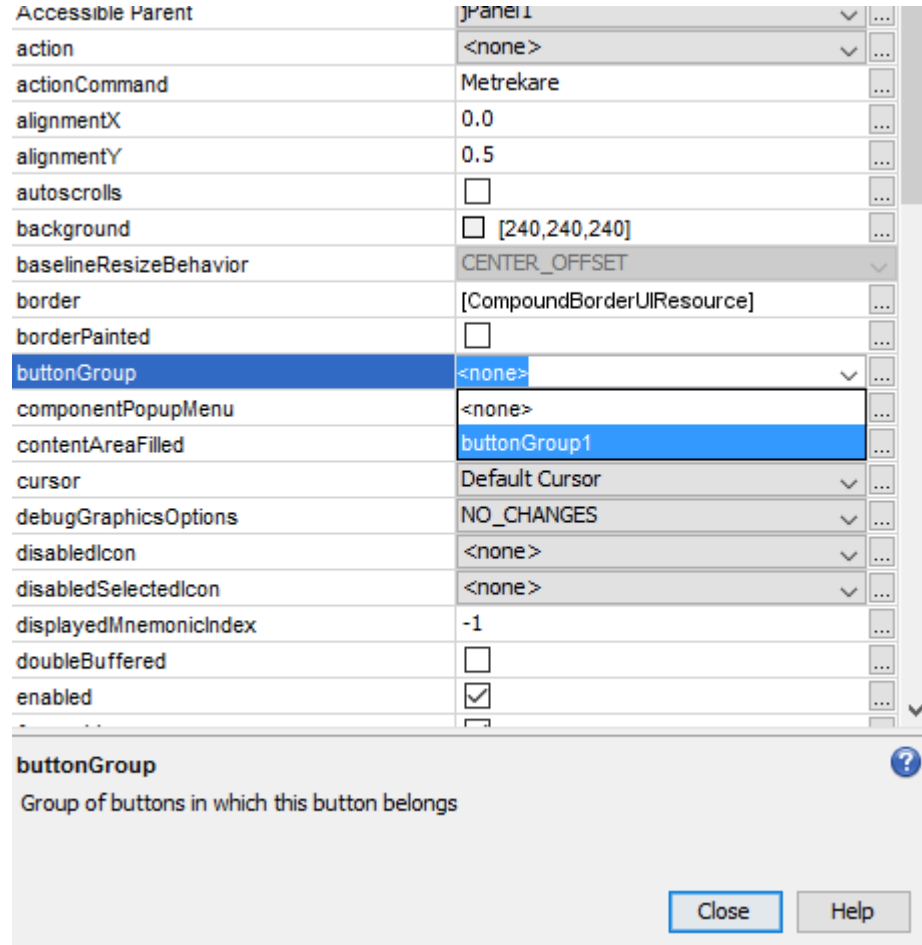
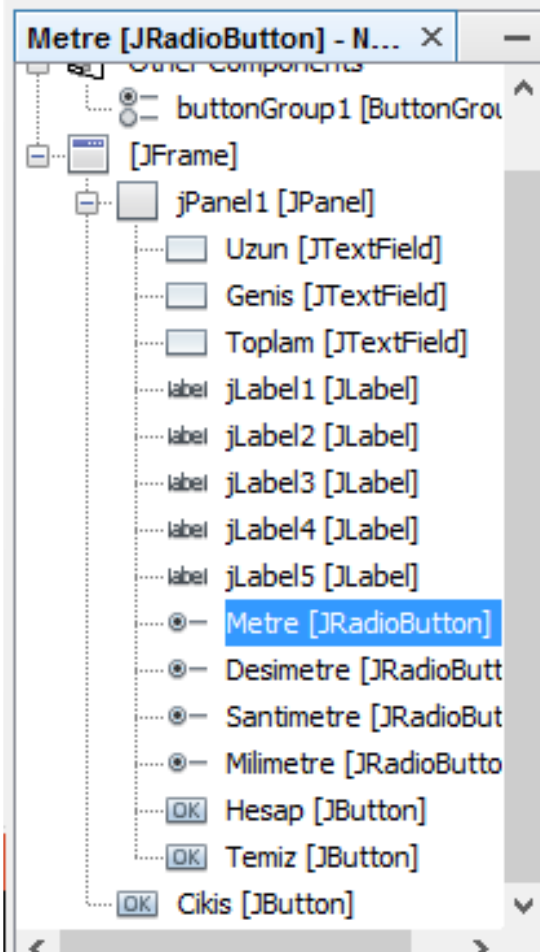
The screenshot shows a Java Swing window with a light gray background. Inside the window, there is a pink rectangular area containing the following elements:

- Four text input fields labeled "Uzunluk", "Genişlik", "Toplam Alan", and "Birim". The "Birim" field contains the text "jLabel5".
- A vertical list of four radio buttons with labels: "Metrekare", "Desimetrekare", "Santimetrekare", and "Milmetrekare".
- Two buttons labeled "Hesapla" and "Temizle" positioned below the radio buttons.
- A "Çıkış" button located at the bottom right of the window.

*****Button Group yapılmazsa radio düğmelerinin hepsi aynı seçilir !!!!

*****ButtonGroup sürüklenerek ekrana bırakılır tasarım alanında görülmez kütükte görülür !!!!

3. Properties Button Group Oluşturulması



*Tüm Radiobuttonlar teker teker Button Group a eklenir !!!

4. Temizle Konumutu Kodlarının yazımı

```
private void TemizActionPerformed(java.awt.event.ActionEvent evt) {  
  
    Uzun.setText (null) ;  
    Genis.setText (null) ;  
    Toplam.setText (null) ;  
  
    // TODO add your handling code here  
  
}
```

Desimetrekare

Hesapla

Temizle

Santimetrekare

Milmetrekare

Çıkış

5. Pencereyi kapama komutu kodları

```
private void CikisActionPerformed(java.awt.event.ActionEvent evt) {  
  
    System.exit(0);  
  
    // TODO add your handling code here  
}
```

The screenshot shows a Java Swing window with a pink background. At the top, there are four labels: "Uzunluk", "Genişlik", "Toplam Alan", and "Birim". Below "Uzunluk" and "Genişlik" are text input fields. Below "Toplam Alan" is a larger text input field. To the right of "Birim" is a label "jLabel5". Below these are four radio buttons for unit selection: "Metrekare", "Desimetrekare", "Santimetrekare", and "Milimetrekare". In the center, there are two buttons: "Hesapla" and "Temizle". At the bottom right, there is a button labeled "Çıkış" (Exit), which is highlighted with a red box. The window has a standard Mac OS-style title bar with a red close button.

Try catch metodu :

- Oluşacak istisnaları yakalamak için kullanılır.
- Dene ve yakala.

6. Hesapla Komutu

1.

```
private void HesapActionPerformed(java.awt.event.ActionEvent evt) {  
  
    double u,g,t;  
    u = Double.parseDouble(Uzun.getText());  
    g = Double.parseDouble(Genis.getText());  
  
}
```

2.

```
private void hesapActionPerformed(java.awt.event.ActionEvent evt) {  
  
    trycatchdouble u,g,t;  
    u = Double.parseDouble(Uzun.getText());  
    g = Double.parseDouble(Genis.getText());  
  
    // TODO add your handling code here:  
  
}
```

Trycatch + TAB Tuşu

3.

```
private void HesapActionPerformed(java.awt.event.ActionEvent evt) {  
  
    try {  
  
    } catch (Exception e) {  
    }  
  
    double u,g,t;  
    u = Double.parseDouble(Uzun.getText());  
    g = Double.parseDouble(Genis.getText());  
  
}
```

4.

```
private void HesapActionPerformed(java.awt.event.ActionEvent evt) {  
  
    try {  
        double u,g,t;  
        u = Double.parseDouble(Uzun.getText());  
        g = Double.parseDouble(Genis.getText());  
  
    } catch (Exception e) {  
  
    }  
  
}
```

***Hatalı deęer girilmesi veya radio düęmesi seçimi yapılmaması gerektiren uyarı mesajları !!!!

```
private void HesapActionPerformed(java.awt.event.ActionEvent evt) {
```

```
    try {  
        double u,g,t;  
        u = Double.parseDouble(Uzun.getText());  
        g = Double.parseDouble(Genis.getText());
```

5.

```
        if (!(Metre.isSelected()) && !(Desimetre.isSelected()) && !(Santimetre.isSelected()) && !(Milimetre.isSelected())) {  
            Toplam.setText("Metre, desimetre, santimetre veya milimetre seç");  
        }
```

6.

```
    } catch (NumberFormatException e) {  
        Topla.setText("Sayı girilmeli ");
```



Uzunluk	Genişlik	Toplam Alan	Birim
<input type="text"/>	<input type="text"/>	<input type="text" value="Sayı girilmeli"/>	jLabel5
<input type="radio"/> Metrekare			
<input type="radio"/> Desimetrekaire		<input type="button" value="Hesapla"/>	<input type="button" value="Temizle"/>
<input type="radio"/> Santimetrekaire			
<input type="radio"/> Milimetrekaire			

```
private void HesapActionPerformed(java.awt.event.ActionEvent evt) {
```

```
try {
```

```
double u,g,t;
```

```
u = Double.parseDouble(Uzun.getText());
```

```
g = Double.parseDouble(Genis.getText());
```

```
if (!(Metre.isSelected()) && !(Desimetre.isSelected()) && !(Santimetre.isSelected()) && !(Milimetre.isSelected())) {
```

```
Toplam.setText("Metre, desimetre, santimetre veya milimetre seç");
```

```
}
```

```
} catch (NumberFormatException e) {
```

```
Topla.setText("Sayı girilmeli ");
```

```
}
```

5.

6.

Radyo düğmesi seçilmemesi
durumunda uyarı !!!

Uzunluk 10 Genişlik 5 Toplam Alan santimetre veya milimetre seç Birim jLabel5

Metrekare
 Desimetrekaire
 Santimetrekaire
 Milmetrekare

Hesapla Temizle

Çıkış

Radio düğmeleri görev kodları

```
if (!(Metre.isSelected()) && !(Desimetre.isSelected()) && !(Santimetre.isSelected()) && !(Milimetre.isSelected())) {  
Topla.setText("Metre, desimetre, santimetre veya milimetre seç");  
}
```

```
if (Metre.isSelected()) {  
t = 2 * (u + g) * 1;  
String top = Double.toString(t);  
Topla.setText(top);  
jLabel5.setText("metrekare");  
}  
if (Desimetre.isSelected()) {  
t = 2 * (u + g) * 100;  
String top = Double.toString(t);  
Topla.setText(top);  
jLabel5.setText("Desimetrekare");  
}  
if (Santimetre.isSelected()) {  
t = 2 * (u + g) * 10000;  
String top = Double.toString(t);  
Topla.setText(top);  
jLabel5.setText("Santimetrekare");  
}  
if (Milimetre.isSelected()) {  
t = 2 * (u + g) * 1000000;  
String top = Double.toString(t);  
Topla.setText(top);  
jLabel5.setText("Milimetrekare");  
}
```

The screenshot shows a Java Swing application window with a pink background. The window contains a form with the following elements:

Uzunluk	Genişlik	Toplam Alan	Birim
20	30	10000.0	Desimetrekare

Below the table, there are four radio buttons for selecting the unit:

- Metrekare
- Desimetrekare
- Santimetrekare
- Milimetrekare

There are two buttons: "Hesapla" and "Temizle". At the bottom right, there is a "Çıkış" button.

```
} catch (NumberFormatException e) {
```