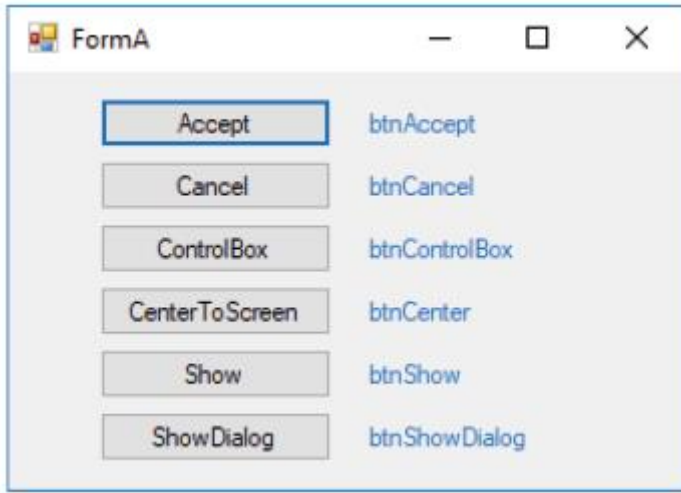


Uygulama 1: Form_Sinifi

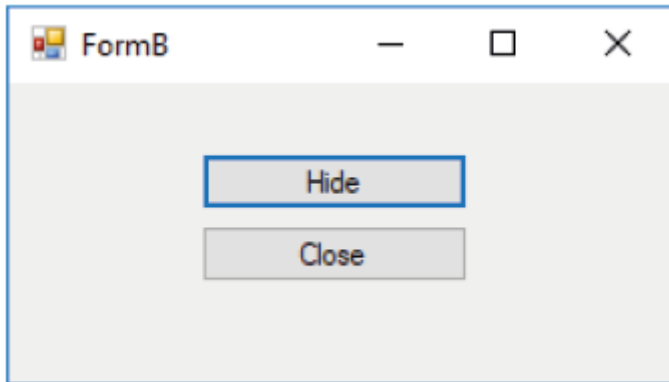
Form Sınıfı

Bu uygulamada Form sınıfı için bazı olayların ve özelliklerin kullanılması gösterilecektir.

- 1. Adım:** Yeni bir proje oluşturarak FormA ve FormB isiminde iki form hazırlayınız.
- 2. Adım:** FormA için Görsel 5.1'deki form tasarımını yapınız ve kontrollere name değerlerini veriniz.



- 3. Adım:** FormB için Görsel 5.2'deki form tasarımını yapınız ve kontrollere name değerlerini veriniz.



- 4. Adım:** Birinci formdaki **Accept** ve **Cancel** butonları Click olaylarına aşağıdaki kodlamaları yapınız. Kodlamaları yaptıktan sonra **Properties** penceresinden formun **AcceptButton** ve **CancelButton** özelliğine ilgili butonları atayınız.

```
1 reference
private void btnAccept_Click(object sender, EventArgs e)
{
    MessageBox.Show("Enter tuşu çalışıyor.");
}
```

```
1 reference
private void btnCancel_Click(object sender, EventArgs e)
{
    MessageBox.Show("Esc tuşu çalışıyor.");
}
```

5. Adım: ControlBox butonu Click olayına formun sağ üst köşesindeki butonların gösterilmesi veya gizlenmesi işlemini gerçekleştiren kodlamayı yapınız.

```
1 reference
private void btnControlBox_Click(object sender, EventArgs e)
{
    if (this.ControlBox == true)
        this.ControlBox = false;
    else
        this.ControlBox = true;
}
```

6. Adım: CenterToScreen butonu Click olayına formu ekranın merkezine yerleştirme işlemini gerçekleştiren kodlamayı yapınız.

```
1 reference
private void btnCenter_Click(object sender, EventArgs e)
{
    this.CenterToScreen();
}
```

7. Adım: Show butonu Click olayına oluşturulan ikinci formun gösterilmesini sağlayan kodlamayı yapınız.

```
1 reference
private void btnShow_Click(object sender, EventArgs e)
{
    FormB formB = new FormB();
    formB.Show();
}
```

8. Adım: ShowDialog butonu Click olayına oluşturulan ikinci formun diyalog penceresi olarak gösterilmesini sağlayan kodlamayı yapınız.

```
1 reference
private void btnShowDialog_Click(object sender, EventArgs e)
{
    FormB formB = new FormB();
    formB.ShowDialog();
}
```

9. Adım: İkinci formun Load olayına form yüklendiğinde çalışan kodlamaları yapınız.

```
1 reference
private void FormB_Load(object sender, EventArgs e)
{
    MessageBox.Show("FormB yükleniyor.");
}
```

10. Adım: İkinci formun FormClosing olayına form kapanırken çalışan kodlamaları yapınız.

```
1 reference
private void FormB_FormClosing(object sender, FormClosingEventArgs e)
{
    MessageBox.Show("FormB kapanıyor.");
}
```

11. Adım: İkinci formun FormClosed olayına form kapandıktan sonra çalışan kodlamaları yapınız.

```
1 reference
private void FormB_FormClosed(object sender, FormClosedEventArgs e)
{
    MessageBox.Show("FormB kapandı.");
}
```

12. Adım: İkinci form içindeki Hide butonu Click olayına formun gizlenmesini sağlayan kodlamaları yapınız

```
1 reference
private void btnHide_Click(object sender, EventArgs e)
{
    this.Hide();
}
```

13. Adım: İkinci form içindeki Close butonu Click olayına formun kapanmasını sağlayan kodlamaları yapınız.

```
1 reference
private void btnClose_Click(object sender, EventArgs e)
{
    this.Close();
}
```

14. Adım: İkinci formun Click olayına form üzerinde herhangi bir yere fare ile tıklandığında çalışan kodlamaları yapınız.

1 reference

```
private void FormB_Click(object sender, EventArgs e)
{
    MessageBox.Show("FormB click olayı çalışıyor.");
}
```