

Uygulama_3: GroupBox Kontrolü

Bu kontroller, diğer kontroller için özelleştirilebilir konteyner görevi gören özel kontrollerdir. Form uygulaması geliştirilirken eklenen kontrollerin fazlalığı, formun görüntüsünü karmaşık hâle getirebilir. Formun karmaşıklığını önlemek ve birbirleriyle ilişkili kontrolleri bir arada tutmak için konteyner kontroller kullanılır. Bunlar; Toolbox penceresinde Containers sekmesi içinde bulunan Panel, Groupbox, TabControl, FlowLayoutPanel kontrolleridir. Örneğin öğrenci ve veli bilgileri kaydedilen bir arabirim formu tasarlandığında öğrenci bilgilerine ait kontroller bir konteyner içinde, veli bilgilerine ait kontroller ise başka bir konteyner içinde tutulabilir.

Konteyner kontrolünün belirli özellikleri değiştirildiğinde içinde bulunan kontrollerin özellikleri de bu durumdan etkilenir. Konteyner kontrolün **Enable** veya **Visible** özellikleri değiştirildiğinde konteyner için- deki tüm kontroller aynı şekilde etkilenir.

GroupBox Kontrolü: Formu bölümlere ayırarak ilişkili kontrolleri bir arada tutmak için kullanılan bir konteyner kontrolüdür.

Bu uygulamada form üzerindeki ilişkili kontrolleri GroupBox içine alarak basit bir bilgisayar fiyat hesaplama uygulaması yapılacaktır.

1. Adım: Görsel 5.3'teki form tasarımını yapınız ve kontrollere name değerlerini veriniz.

GroupBox Kontrolü Uygulama 1

İşlemciler

- ☒ Intel Core i7 `rbCpuI7`
- ☐ Intel Core i5 `rbCpuI5`
- ☐ Intel Core i3 `rbCpuI3`
- ☐ AMD Ryzen 5 `rbCpuR5`
- ☐ AMD Ryzen 3 `rbCpuR3`

Ram Bellek

- ☐ 16 GB `rbRam16`
- ☐ 8 GB `rbRam8`
- ☐ 4 GB `rbRam4`

Sabit Disk

- ☐ 1 TB `rbHdd1000`
- ☐ 500 GB `rbHdd500`
- ☐ 320 GB `rbHdd320`

Ek Donanımlar

- ☐ DVD RW `cbDvd`
- ☐ Web Cam `cbWebCam`
- ☐ Wireless Kart `cbWifi`

Hesapla

`btnHesapla`

2. Adım: Hesapla butonu Click olayına belirlenen taban fiyata, seçilen her donanımın fiyatını ekleyen ve toplam fiyatı hesaplayıp, mesaj olarak gösteren kodlamayı yapınız.

1 reference

```
private void btnHesapla_Click(object sender, EventArgs e)
{
    decimal tabanFiyat = 500;

    // İşlemci fiyat hesaplaması
    decimal cpuFiyat = 0;
    if (rbCpuI7.Checked)
        cpuFiyat = 300;
    else if (rbCpuI5.Checked)
        cpuFiyat = 200;
    else if (rbCpuI3.Checked)
        cpuFiyat = 100;
    else if (rbCpuR5.Checked)
        cpuFiyat = 250;
    else if (rbCpuR3.Checked)
        cpuFiyat = 150;

    tabanFiyat += cpuFiyat;

    // Ram bellek fiyat hesaplaması
    decimal ramFiyat = 0;
    if (rbRam16.Checked)
        ramFiyat = 125;
    else if (rbRam8.Checked)
        ramFiyat = 75;
    else if (rbRam4.Checked)
        ramFiyat = 45;

    tabanFiyat += ramFiyat;

    MessageBox.Show(string.Format("Toplam Fiyat ={0:C}", tabanFiyat));
}
```

Not: Uygulamadaki diğer donanım özellikleri için fiyat hesaplamasını yapınız.