

Buton yakalama oyunu

- Form açıldığında, ekranda bir **buton** (örneğin “Yakala Beni!”) gözükecek.
- Her 1 saniyede (veya daha kısa/uzun aralıklarda) buton **rasgele bir konuma** taşınacak.
- Oyuncu butona tıkladığında **puan** kazanacak (örneğin +1).
- Aynı zamanda bir **geri sayım** (örneğin 30 saniye) çalışacak. Süre bittiğinde oyun sonlanacak ve ekranda “Oyun Bitti, Puanınız: X” mesajı gösterilecek.

Form Tasarımı (Örnek)

1. Button

- Name: buttonCatch
- Text: "Yakala Beni!"
- Varsayılan konumu ortada veya herhangi bir yerde olabilir.

2. Label

- Name: labelScore
- Text: "Puan: 0" (başlangıç puanı)

3. Label

- Name: labelTime
- Text: "Kalan Süre: 30" (örneğin 30 saniyeden başlatacağız)

4. Timer (rasgele konum değiştirecek)

- Name: timerMove
- Interval: 1000 ms (1 saniye) veya 800 ms, 500 ms, vb.
- Enabled = false (oyun başlayınca aktif edilebilir)

5. Timer (geri sayım için)

- Name: timerCountdown
- Interval: 1000 ms (1 saniye)
- Enabled = false (oyun başlayınca aktif edilebilir)

6. Button (isteğe bağlı “Oyunu Başlat”)

- Name: buttonStart
- Text: "Oyunu Başlat"

Böylece kullanıcı *Oyunu Başlat* butonuna tıkladığında zaman işlemeye ve buton hareket etmeye başlayacak, kullanıcı yakalamaya çalışacak.

Örnek Kod

Aşağıdaki kod, *Form1* içinde minimal bir “buton yakalama” oyununu göstermektedir. Kendi formunuza eklediğiniz bileşenlerin Name özelliklerini bu kodla uyumlu hâle getirmelisiniz.

```
using System;

using System.Drawing;

using System.Windows.Forms;

namespace ButonYakalamaOyunu
{
    public partial class Form1 : Form
    {
        private int score = 0;    // Oyuncunun topladığı puan
        private int timeLeft = 30; // Geri sayım başlangıç süresi (saniye)
        private Random rnd = new Random(); // Rasgele konum için

        public Form1()
        {
            InitializeComponent();
        }

        private void Form1_Load(object sender, EventArgs e)
        {
            // Başlangıç ayarları

            labelScore.Text = "Puan: 0";

            labelTime.Text = "Kalan Süre: 30";

            timerMove.Interval = 1000; // 1 saniyede bir hareket
```

```
timerCountdown.Interval = 1000; // 1 saniyede bir geri sayım  
timerMove.Enabled = false;  
timerCountdown.Enabled = false;  
}
```

```
// "Oyunu Başlat" butonu
```

```
private void buttonStart_Click(object sender, EventArgs e)
```

```
{  
    // Değerleri sıfırla  
    score = 0;  
    timeLeft = 30;  
    labelScore.Text = "Puan: 0";  
    labelTime.Text = "Kalan Süre: 30";  
    // Timers aktif  
    timerMove.Enabled = true;  
    timerCountdown.Enabled = true;  
}
```

```
// Hareket timer'ı (timerMove) -> Tick olayı
```

```
private void timerMove_Tick(object sender, EventArgs e)
```

```
{  
    MoveButtonToRandomPosition();  
}
```

```
// Geri sayım timer'ı (timerCountdown) -> Tick olayı
```

```
private void timerCountdown_Tick(object sender, EventArgs e)
```

```
{  
    timeLeft--;
```

```
labelTime.Text = "Kalan Süre: " + timeLeft;
if (timeLeft <= 0)
{
    // Oyun Bitti
    timerMove.Enabled = false;
    timerCountdown.Enabled = false;
    MessageBox.Show("Süre doldu! Puanınız: " + score,
        "Oyun Bitti",
        MessageBoxButtons.OK,
        MessageBoxIcon.Information);
}
}

// Buton yakalama event'i
private void buttonCatch_Click(object sender, EventArgs e)
{
    // Her tıklamada puanı arttır
    score++;
    labelScore.Text = "Puan: " + score;
    // Tıklayınca anında başka yere kaçabilir (isteğe bağlı)
    MoveButtonToRandomPosition();
}

// Butonu Form içerisinde rasgele bir konuma taşıyan fonksiyon
private void MoveButtonToRandomPosition()
{
    // Form'un boyutuna göre butonun gidebileceği max X ve Y değerini
    hesaplarken,
```

// butonun genişlik ve yüksekliğini de hesaba katıyoruz ki buton formun dışına taşmasın.

```
int maxX = this.ClientSize.Width - buttonCatch.Width;
```

```
int maxY = this.ClientSize.Height - buttonCatch.Height;
```

```
if (maxX < 0 || maxY < 0) return; // Form çok küçükse hata engellemek için
```

```
int newX = rnd.Next(0, maxX);
```

```
int newY = rnd.Next(0, maxY);
```

```
buttonCatch.Location = new Point(newX, newY);
```

```
}
```

```
}
```

```
}
```

Kod Açıklaması

1. **score**: Oyuncunun topladığı puan sayısı.
2. **timeLeft**: Geri sayım için başlangıç değeri (30 saniye).
3. **timerMove**: Butonun rastgele konuma taşınmasından sorumlu. Interval ayarı (ör. 1000 ms), Tick'te `MoveButtonToRandomPosition()` çağırarak butonun konumunu değiştirir.
4. **timerCountdown**: Her saniye `timeLeft` değerini 1 azaltır. 0'a inince oyun biter, skor mesajı gösterilir.
5. **MoveButtonToRandomPosition()**: Random nesnesiyle (`rnd`) X ve Y değerleri hesaplanır, `buttonCatch.Location` güncellenir.
6. **buttonCatch_Click**: Kullanıcı butona tıklarsa `score` artar, ekrana yazılır; opsiyonel olarak o anda buton tekrar rasgele yere gidebilir.