

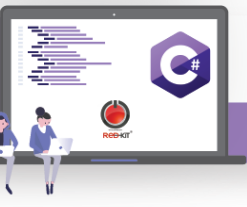
FORM EKRANI

FORM EKRANI

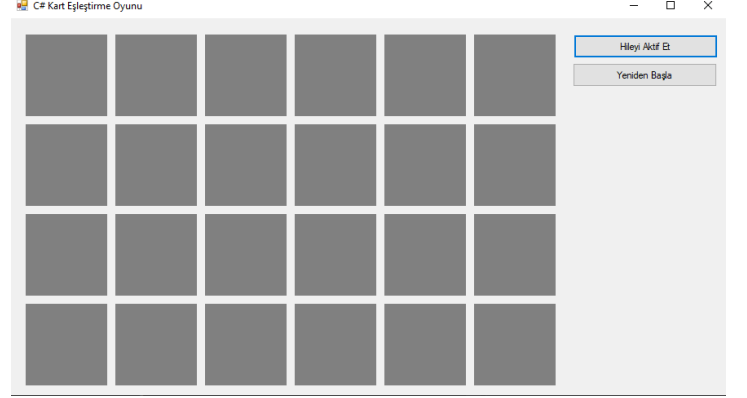
✓ Proje 3

KAZANIMLAR

✓ Genel Tekrar



Proje: C# dilini kullanarak Kart Eşleştirme Oyunu yapalım. Oyundaki kartlar form üzerinde değilde dinamik olarak belirlenen miktarda oluşturulmaktadır. max_oyun_alan değerini kaç yaparsanız kartların sayısı okadar olacaktır. İsterseniz hileyi aktif et seçeneği ile kartları tekrar 1.5 sn lik görünür yapabilir ve oyununuza devam edebilirsiniz. İsterseniz de yeniden başlat seçeneği ile oyunu yeniden başlatabilirsiniz.



Kod Blokları

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace WindowsFormsApplication1
{
    public partial class Form1 : Form
    {
        int TabIdIndex = -1;
        int tiklanan = -1;
        int max_oyun_alan = 24;
        List<Panel> picture = new List<Panel>();
        List<int> renkler_list = new List<int>();

        List<Color> renkler = new List<Color>();

        public Form1()
        {
            InitializeComponent();
        }
        public List<Color> Shuffle(List<Color> renk)
        {
            Random rng = new Random();
            int n = renk.Count;
```



```
while (n > 1)
{
    n--;
    int k = rng.Next(n + 1);
    Color value = renk[k];
    renk[k] = renk[n];
    renk[n] = value;
}
return renk;
}
private void Form1_Load(object sender, EventArgs e)
{
    oyunOlustur();
}
private void oyunOlustur()
{
    renkler.Clear();

    if (picture.Count > 0)
    {
        foreach (var item in picture)
        {
            this.Controls.Remove(item);
        }
        picture.Clear();
    }
    renkler_list.Clear();
    tiklanan = -1;
    TabIdIndex = -1;
    Random rastgele = new Random();
    for (int i = 0; i < max_oyun_alan / 2; i++)
    {
        var renk_r = rastgele.Next(0, 255);
        var renk_g = rastgele.Next(0, 255);
        var renk_b = rastgele.Next(0, 255);
        Color color = Color.FromArgb(renk_r, renk_g, renk_b);
        renkler.Add(color);
        renkler.Add(color);
    }
    renkler = Shuffle(renkler);

    var d_top = 20;
    var d_left = 20;

    for (int i = 1; i <= max_oyun_alan; i++)
    {
        Panel m = new Panel();
        m.Left = d_left;
        m.Top = d_top;
        m.Width = 100;
        m.Height = 100;
        d_left += 110;
        m.Click += tiklama_picture;
        m.TabIndex = i;
        m.Enabled = false;
        m.BackColor = renkler[i - 1];
        renkler_list.Add(i - 1);
    }
}
```



```
        if (i % 6 == 0)
        {
            d_top += 110;
            d_left = 20;
        }

        this.Controls.Add(m);
        picture.Add(m);
    }
    button1.Enabled = false;
    button2.Enabled = false;
    timer1.Start();
}

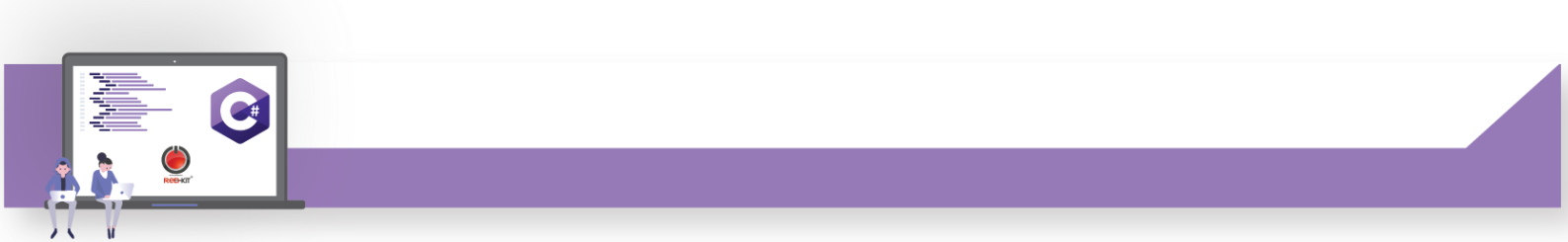
private void tiklama_picture(object sender, EventArgs e)
{
    Panel resim = (Panel)sender;
    TabIdIndex = resim.TabIndex - 1;

    if(TabIdIndex != tiklanan)
    {
        if (tiklanan != -1)
        {
            picture[TabIdIndex].BackColor = renkler[renkler_list[TabIdIndex]];

            if (renkler[renkler_list[TabIdIndex]] == renkler[renkler_list[tiklanan]])
            {
                timer3.Start();
            }
            else
            {
                for (int i = 0; i < max_oyun_alan; i++)
                {
                    picture[i].Enabled = false;
                }

                timer2.Start();
            }
        }
        else
        {
            tiklanan = TabIdIndex;
            picture[TabIdIndex].BackColor = renkler[renkler_list[TabIdIndex]];
        }
    }
}

private void timer1_Tick(object sender, EventArgs e)
{
    button1.Enabled = true;
    button2.Enabled = true;
    for (int i = 0; i < max_oyun_alan; i++)
    {
        picture[i].Enabled = true;
        picture[i].BackColor = Color.Gray;
    }
    timer1.Stop();
}
```



```
private void button1_Click(object sender, EventArgs e)
{
    button1.Enabled = false;
    for (int i = 0; i < max_oyun_alan; i++)
    {
        picture[i].BackColor = renkler[i];
        picture[i].Enabled = false;
    }
    timer1.Stop();
    timer1.Start();
}
private void timer2_Tick(object sender, EventArgs e)
{
    picture[TabIdIndex].BackColor = Color.Gray;
    picture[tiklanan].BackColor = Color.Gray;
    tiklanan = -1;
    for (int i = 0; i < max_oyun_alan; i++)
    {
        picture[i].Enabled = true;
    }
    tiklanan = -1;

    timer2.Stop();
}
private void timer3_Tick(object sender, EventArgs e)
{
    picture[TabIdIndex].Visible = false;
    picture[tiklanan].Visible = false;
    tiklanan = -1;
    timer3.Stop();
    var sonuc = oyunBitti();
    if(sonuc)
    {
        MessageBox.Show("Oyun Bitti! Harikasın Adamım");
    }
}
private bool oyunBitti()
{
    int bitti = 0;
    for (int i = 0; i < max_oyun_alan; i++)
    {
        if(picture[i].Visible == false)
        {
            bitti++;
        }
    }
    return (bitti == max_oyun_alan ? true : false);
}

private void button2_Click(object sender, EventArgs e)
{
    this.oyunOlustur();
}

}
```

