

Futbolcu kayıt programı.

Uygulamamızın amacı Futbolcularının bilgilerini alıp bir listboxa kaydetmek ve daha sonra butonlar ve comboboxlar sayesinde filtreleme yapmak.

Uygulamamızda öncelikle futbolcunun bilgilerini alabileceğimiz bir sınıf tanımladık. Daha sonra enum ile takımları tanımladık. Listboxta göstereceğimiz bilgiler futbolcunun bilgilerinden farklı olduğu için bu bilgileri göstermek için başka bir sınıf daha tanımladık. Daha sonra bunları listboxa yazdırdık ve filtreleme seçeneklerimizi yaptık:

İlk filtreleme yaşa göre filtreleme oldu. 2 textbox aldık ve min max yaş aralığı verdik. Butona basınca bu yaşlar arasındaki futbolcuları getirecek uygulamamız.

İkinci filtreleme takıma göre filtreleme. Bunun için bir combobox tanımladık ve içine takımlarımız attık. Comboboxtan takım seçer seçmez takımlara göre seçip listboxa yazıyor.

Üçüncü filtreleme mevkiye göre filtreleme oldu. Burada da aynı takıma göre filtreleme gibi bir combobox var ve oraya pozisyonları atadık ve seçer seçmez filtreleme yapıyor.

Uygulamamızın diğer bir özelliği de listboxta seçili bir elemanı silme işlemi yapıyor olması. Filtrenilmiş öğeleri de silebiliyoruz.

Clear butonumuz var bir de. Bu buton sayesinde alanları temizleyebiliyoruz.

Aynı zamanda Save butonuna basınca da alanlarımız temizleniyor.

Formumuz şu şekildedir:

The screenshot shows a web application titled "Football Player Registration System". The form is divided into two main sections. The left section, labeled "Registration", contains fields for "Name Surname", "Team", "Smash Force", "Rust Strength", "Date Of Birth", and "Position". Below these fields are "Save", "Cancel", and "Remove" buttons. The right section, labeled "Age Filtering", "Team Filtering", and "Position Filtering", contains "Min Age" and "Max Age" textboxes, a "Find" button, and dropdown menus for "Team Filtering" and "Position Filtering". Below these filters is a large listbox labeled "lstBx1".

Futbolcu bilgilerini tutan classımız:

```
class FootballPlayerInformation
{
    public string NameSurname{ get; set; }
    public string TeamName { get; set; }
    public string SmashForce { get; set; }
    public string RustStrength { get; set; }
    public DateTime BirthDate{ get; set; }
    public string Position { get; set; }
    public override string ToString()
    {
        return string.Format("{0},{1},{2},{3},{4},{5}", NameSurname, TeamName,
SmashForce, RustStrength, BirthDate, Position);
    }
}
```

Enum olarak tanımladığımız Team:

```
public enum Team
{
    Beşiktaş=1,
    Galatasaray=2,
    Fenerbahçe=3,
    Trabzonspor=4
};
```

Listboxa yazdığımız bilgileri tutan sınıfımız:

```
class FootballerListBx
{
    public string NameSurname { get; set; }
    public string TeamName { get; set; }
    public string Position { get; set; }
    public int age { get; set; }
    public override string ToString()
    {
        return string.Format("{0},{1},{2},{3}", NameSurname, TeamName,
Position,age);
    }
}
```

Formumuz:

```
namespace FootballPlayerRegistrationSystem
{
    public partial class FrmRegistration : Form
    {
        FootballPlayerInformation footballer1;
        string[] PositionArray = { "Defense", "Midfield", "Striker" };
        string[] PositionArrayClone = { "Defense", "Midfield", "Striker" };

        public int age;
        List<FootballerListBx> footballerlist;
        List<FootballerListBx> footballerlistlist;
        FootballerListBx footballPlayer1;
```

```

List<FootbalerListBx> list;
//List<FootbalerListBx> listpos;
//List<FootbalerListBx> listteam;

public FrmRegistration()
{
    InitializeComponent();
}

public void ClearItems()
{
    txtBxNameSurname.Clear();
    cmbBoxTeam.SelectedItem = null;
    nmrcUDSmash.Value = 0;
    nmrcUDRust.Value = 0;
    DtTmPckerBirthDate.Value = DateTime.Now;
    cmbBxPosition.SelectedIndex = 0;
}

private void FrmRegistration_Load(object sender, EventArgs e)
{
    cmbBoxTeam.DataSource = Enum.GetValues(typeof(Team));

    cmbBxPosition.DataSource = PositionArrayClone;
    footballerlist = new List<FootbalerListBx>();
    cmbBxTeamFiltering.DataSource = Enum.GetValues(typeof(Team));
    cmbBxPositionFiltering.DataSource = PositionArray;
    footballerlistlist = new List<FootbalerListBx>();
}

private void btnnSave_Click(object sender, EventArgs e)
{
    footballer1 = new FootballPlayerInformation();
    footballPlayer1 = new FootbalerListBx();

    footballer1.NameSurname = txtBxNameSurname.Text;
    footballer1.TeamName = cmbBoxTeam.Text;
    footballer1.SmashForce = nmrcUDSmash.Value.ToString();
    footballer1.RustStrength = nmrcUDRust.Value.ToString();
    footballer1.BirthDate = DtTmPckerBirthDate.Value.ToUniversalTime();
    footballer1.Position = cmbBxPosition.Text;

    //Yaş hesaplama:
    DateTime birthDate;
    birthDate = Convert.ToDateTime(DtTmPckerBirthDate.Text);
    age = DateTime.Now.Year - birthDate.Year;

    footballPlayer1.NameSurname = footballer1.NameSurname;
    footballPlayer1.TeamName = footballer1.TeamName;
    footballPlayer1.Position = footballer1.Position;
    footballPlayer1.age = age;

    footballerlist.Add(footballPlayer1);
    lstBx1.DataSource = null;
    lstBx1.DataSource = footballerlist;
    lstBx1.Refresh();

    ClearItems();
}

```

Diğer buton ve metodlar:

```
private void btnFind_Click(object sender, EventArgs e)
{
    int minage;
    Int32.TryParse(txtBxMinAge.Text, out minage);
    int maxage;
    Int32.TryParse(txtBxMaxAge.Text, out maxage);
    list = footballerlist.Where(x => x.age > minage && x.age <
maxage).ToList();
    lstBx1.DataSource = null;
    lstBx1.DataSource = list;
    lstBx1.Refresh();
}

private void cmbBxTeamFiltering_SelectedIndexChanged(object sender, EventArgs
e)
{
    if (cmbBxTeamFiltering.Text == "Beşiktaş")
    {
        list = footballerlist.Where(x => x.TeamName == "Beşiktaş").ToList();
        lstBx1.DataSource = null;
        lstBx1.DataSource = list;
        lstBx1.Refresh();
    }
    else if (cmbBxTeamFiltering.Text == "Galatasaray")
    {
        list = footballerlist.Where(x => x.TeamName ==
"Galatasaray").ToList();
        lstBx1.DataSource = null;
        lstBx1.DataSource = list;
        lstBx1.Refresh();
    }
    else if (cmbBxTeamFiltering.Text == "Fenerbahçe")
    {
        list = footballerlist.Where(x => x.TeamName == "Fenerbahçe").ToList();
        lstBx1.DataSource = null;
        lstBx1.DataSource = list;
        lstBx1.Refresh();
    }
    else
    {
        list = footballerlist.Where(x => x.TeamName ==
"Trabzonspor").ToList();
        lstBx1.DataSource = null;
        lstBx1.DataSource = list;
        lstBx1.Refresh();
    }
}

private void cmbBxPositionFiltering_SelectedIndexChanged(object sender,
EventArgs e)
{
    if (cmbBxPositionFiltering.Text == "Defense")
```

```

{
    list = footballerlist.Where(x => x.Position == "Defense").ToList();
    lstBx1.DataSource = null;
    lstBx1.DataSource = list;
    lstBx1.Refresh();
}
else if (cmbBxPositionFiltering.Text == "Midfield")
{
    list = footballerlist.Where(x => x.Position == "Midfield").ToList();
    lstBx1.DataSource = null;
    lstBx1.DataSource = list;
    lstBx1.Refresh();
}
else
{
    list = footballerlist.Where(x => x.Position == "Striker").ToList();
    lstBx1.DataSource = null;
    lstBx1.DataSource = list;
    lstBx1.Refresh();
}
}

private void btnRemove_Click(object sender, EventArgs e)
{
    list.Remove((FootbalerListBox)lstBx1.SelectedItem);
    //lstBx1.DataSource = null;
    lstBx1.DataSource = list;
    lstBx1.Refresh();

    list.Remove((FootbalerListBox)lstBx1.SelectedItem);
    //lstBx1.DataSource = null;
    lstBx1.DataSource = list;
    lstBx1.Refresh();

    list.Remove((FootbalerListBox)lstBx1.SelectedItem);
    //lstBx1.DataSource = null;
    lstBx1.DataSource = list;
    lstBx1.Refresh();

    list = footballerlist;
    list.Remove((FootbalerListBox)lstBx1.SelectedItem);
    lstBx1.DataSource = null;
    lstBx1.DataSource = list;
    lstBx1.Refresh();
}

private void btnCancel_Click(object sender, EventArgs e)
{
    ClearItems();
}

}
}

```

Bu da main sınıfımız:

```

static class Program
{
    /// <summary>
    /// The main entry point for the application.
    /// </summary>

```

```

[STAThread]
static void Main()
{
    Application.EnableVisualStyles();
    Application.SetCompatibleTextRenderingDefault(false);
    Application.Run(new FrmRegistration());
}
}

```

Listboxta seçili olan kayıta tıkladığımızda bilgilerin alanlara dolmasını sağlayan eklentiler yaptık.

Daha sonra seçtiğimiz bir kaydı güncelleyen bir eklenti yazdık.

İsim kısmının boş kalmamasını sağladık.

18 yaşından büyük olmayan kişilerin kayıt olmamasını sağladık.

Oyunculara lisans verdik ve bununla ilgili gerekli kontroller yapıldı.

Listboxta seçili olan veriyi başka bir forma gönderdik. Ve orada başka bir listboxa yazdırdık.

Kodumuzun ve formumuzun son hali şu şekildedir:

Form1 kodları:

FootballPlayerInformation sınıfımız aynı.

Team sınıfımız aynı.

FootballerListBx sınıfımızı güncelledik.

```

class FootballerListBx
{
    public string NameSurname { get; set; }
    public string TeamName { get; set; }
    public string Position { get; set; }
    public int age { get; set; }
    public bool License { get; set; }
    public override string ToString()

```

```

        {
            return string.Format("{0},{1},{2},{3}", NameSurname, TeamName,
Position,age);
        }
    }
}

```

Form1'in son hali:

```

namespace FootballPlayerRegistrationSystem
{
    public partial class FrmRegistration : Form
    {

        FootballPlayerInformation footballer1;
        string[] PositionArray = { "Defense", "Midfield", "Striker" };
        string[] PositionArrayClone = { "Defense", "Midfield", "Striker" };

        public int age;
        public int age2;
        bool FootballerLicense=false;
        List<FootbalerListBx> footballerlist;
        FootbalerListBx footballPlayer1;
        List<FootbalerListBx> list;
        FootbalerListBx selectedfill;
        FootbalerListBx footballPlayer2;
        FootballPlayerInformation footballer2;
        List<FootbalerListBx> footballerlist2;
        FootbalerListBx sendingBetweenForms;

        public FrmRegistration()
        {
            InitializeComponent();
        }
        public void ClearItems()
        {
            txtBxNameSurname.Clear();
            cmbBoxTeam.SelectedItem = null;
            nmrcUDSmash.Value = 0;
            nmrcUDRust.Value = 0;
            DtTmPckerBirthDate.Value = DateTime.Now;
            cmbBxPosition.SelectedItem = null;
            chckBxLicense.Checked = false;
            chckBxTakeLicense.Checked = false;
        }
        private void FrmRegistration_Load(object sender, EventArgs e)
        {
            cmbBoxTeam.DataSource = Enum.GetValues(typeof(Team));

            cmbBxPosition.DataSource = PositionArrayClone;
            footballerlist = new List<FootbalerListBx>();
            cmbBoxTeamFiltering.DataSource = Enum.GetValues(typeof(Team));
            cmbBxPositionFiltering.DataSource = PositionArray;
            footballerlist2 = new List<FootbalerListBx>();
        }
        private void chckBxLisans_CheckedChanged(object sender, EventArgs e)
        {
            if(chckBxLicense.Checked==true)
            {

```

```

        FootballerLicense = true;
    }
}
private void btnnSave_Click(object sender, EventArgs e)
{
    try
    {
        if(String.IsNullOrEmpty(txtBxNameSurname.Text))
        {
            MessageBox.Show("Namespace is required!");
        }
        else {

            footballer1 = new FootballPlayerInformation();
            footballPlayer1 = new FootbalerListBx();

            footballer1.NameSurname = txtBxNameSurname.Text;
            footballer1.TeamName = cmbBoxTeam.Text;
            footballer1.SmashForce = nmrcUDSmash.Value.ToString();
            footballer1.RustStrength = nmrcUDRust.Value.ToString();
            footballer1.BirthDate = DtTmPckerBirthDate.Value.ToUniversalTime();
            footballer1.Position = cmbBxPosition.Text;

            //Yaş hesaplama:
            DateTime birthDate;
            birthDate = Convert.ToDateTime(DtTmPckerBirthDate.Text);
            age = DateTime.Now.Year - birthDate.Year;
            if(age<18)
            {
                MessageBox.Show("Error!Footballer cannot be under 18.");
            }
            else
            {
                footballPlayer1.NameSurname = footballer1.NameSurname;
                footballPlayer1.TeamName = footballer1.TeamName;
                footballPlayer1.Position = footballer1.Position;
                footballPlayer1.age = age;
                footballPlayer1.License = FootballerLicense;

                footballerlist.Add(footballPlayer1);
                lstBx1.DataSource = null;
                lstBx1.DataSource = footballerlist;
                lstBx1.Refresh();

                ClearItems();
            }
        }
    }
    catch (Exception error)
    {
        MessageBox.Show(error.ToString());
    }
}

private void btnFind_Click(object sender, EventArgs e)
{
    try
    {

```



```

        int minage;
        Int32.TryParse(nmrcUDMinage.Text, out minage);
        int maxage;
        Int32.TryParse(nmrcUDMaxAge.Text, out maxage);
        list = footballerlist.Where(x => x.age > minage && x.age < maxage).ToList();
        lstBx1.DataSource = null;
        lstBx1.DataSource = list;
        lstBx1.Refresh();
    }

    catch (Exception error)
    {
        MessageBox.Show(error.ToString());
    }
}

e) private void cmbBxTeamFiltering_SelectedIndexChanged(object sender, EventArgs
{
    try
    {

        if (cmbBxTeamFiltering.Text == "Beşiktaş")
        {
            list = footballerlist.Where(x => x.TeamName == "Beşiktaş").ToList();
            lstBx1.DataSource = null;
            lstBx1.DataSource = list;
            lstBx1.Refresh();
        }
        else if (cmbBxTeamFiltering.Text == "Galatasaray")
        {
            list = footballerlist.Where(x => x.TeamName == "Galatasaray").ToList();
            lstBx1.DataSource = null;
            lstBx1.DataSource = list;
            lstBx1.Refresh();
        }
        else if (cmbBxTeamFiltering.Text == "Fenerbahçe")
        {
            list = footballerlist.Where(x => x.TeamName == "Fenerbahçe").ToList();
            lstBx1.DataSource = null;
            lstBx1.DataSource = list;
            lstBx1.Refresh();
        }
        else
        {
            list = footballerlist.Where(x => x.TeamName == "Trabzonspor").ToList();
            lstBx1.DataSource = null;
            lstBx1.DataSource = list;
            lstBx1.Refresh();
        }
    }
    catch (Exception error)
    {
        MessageBox.Show(error.ToString());
    }
}

```

```

        private void cmbBxPositionFiltering_SelectedIndexChanged(object sender,
EventArgs e)
        {
            try
            {

                if (cmbBxPositionFiltering.Text == "Defense")
                {
                    list = footballerlist.Where(x => x.Position == "Defense").ToList();
                    lstBx1.DataSource = null;
                    lstBx1.DataSource = list;
                    lstBx1.Refresh();
                }
                else if (cmbBxPositionFiltering.Text == "Midfield")
                {
                    list = footballerlist.Where(x => x.Position == "Midfield").ToList();
                    lstBx1.DataSource = null;
                    lstBx1.DataSource = list;
                    lstBx1.Refresh();
                }
                else
                {
                    list = footballerlist.Where(x => x.Position == "Striker").ToList();
                    lstBx1.DataSource = null;
                    lstBx1.DataSource = list;
                    lstBx1.Refresh();
                }
            }
            catch (Exception error)
            {
                MessageBox.Show(error.ToString());
            }
        }

        private void btnRemove_Click(object sender, EventArgs e)
        {
            try
            {

                list.Remove((FootbalerListBx)lstBx1.SelectedItem);
                //lstBx1.DataSource = null;
                lstBx1.DataSource = list;
                lstBx1.Refresh();

                list.Remove((FootbalerListBx)lstBx1.SelectedItem);
                //lstBx1.DataSource = null;
                lstBx1.DataSource = list;
                lstBx1.Refresh();

                list.Remove((FootbalerListBx)lstBx1.SelectedItem);
                //lstBx1.DataSource = null;
                lstBx1.DataSource = list;
                lstBx1.Refresh();

                list = footballerlist;
                list.Remove((FootbalerListBx)lstBx1.SelectedItem);
                lstBx1.DataSource = null;
                lstBx1.DataSource = list;
                lstBx1.Refresh();
            }
        }
    }

```

```

    }

    catch (Exception error)
    {
        MessageBox.Show(error.ToString());
    }
}
private void btnnCancel_Click(object sender, EventArgs e)
{
    ClearItems();
}

private void lstBx1_Click(object sender, EventArgs e)
{
    try
    {

        selectedfill = new FootballerListBx();
        //txtBxNameSurname.Text = Convert.ToString(lstBx1.SelectedItem);

        selectedfill = (FootballerListBx)lstBx1.SelectedItem;
        ClearItems();

        txtBxNameSurname.Text = selectedfill.NameSurname;
        cmbBoxTeam.Text = selectedfill.TeamName;
        cmbBxPosition.Text = selectedfill.Position;
        chckBxLicense.Visible = false;
    }
    catch (Exception error)
    {
        MessageBox.Show(error.ToString());
    }
}

private void btnnUpdate_Click(object sender, EventArgs e)
{
    try
    {
        chckBxLicense.Visible = true;

        footballerlist2 = footballerlist;
        footballerlist2.Remove((FootballerListBx)lstBx1.SelectedItem);
        lstBx1.DataSource = null;
        lstBx1.DataSource = footballerlist2;
        lstBx1.Refresh();

        footballer2 = new FootballPlayerInformation();
        footballPlayer2 = new FootballerListBx();

        footballer2.NameSurname = txtBxNameSurname.Text;
        footballer2.TeamName = cmbBoxTeam.Text;
        footballer2.SmashForce = nmrcUDSmash.Value.ToString();
        footballer2.RustStrength = nmrcUDRust.Value.ToString();
        footballer2.BirthDate = DtTmPckerBirthDate.Value.ToUniversalTime();
        footballer2.Position = cmbBxPosition.Text;
        footballPlayer2.License = FootballerLicense;
        //Yaş hesaplama:
        DateTime birthDate2;
        birthDate2 = Convert.ToDateTime(DtTmPckerBirthDate.Text);
    }
    catch (Exception error)
    {
        MessageBox.Show(error.ToString());
    }
}

```

```

        age2 = DateTime.Now.Year - birthDate2.Year;

        if (age2 < 18)
        {
            MessageBox.Show("Error!Footballer cannot be under 18.");
        }
        else
        {
            if (chckBxTakeLicense.Checked)
            {
                if (footballPlayer2.License == true)
                {
                    MessageBox.Show("This footballer already have a license!");
                }

                else {
                    FootballerLicense = true;

                    footballPlayer2.NameSurname = footballer2.NameSurname;
                    footballPlayer2.TeamName = footballer2.TeamName;
                    footballPlayer2.Position = footballer2.Position;
                    footballPlayer2.age = age;

                    footballerlist2.Add(footballPlayer2);
                    //lstBx1.DataSource = null;
                    lstBx1.DataSource = list;
                    lstBx1.DataSource = footballerlist2;
                    //lstBx1.Refresh();
                }
            }
        }
        ClearItems();
    }

    catch (Exception error)
    {
        MessageBox.Show(error.ToString());
    }

}

private void btnnDetail_Click(object sender, EventArgs e)
{
    Form2 form = new Form2();
    sendingBetweenForms = ((FootbalerListBx)lstBx1.SelectedItem);
    form.Form2Method(sendingBetweenForms.NameSurname,
    sendingBetweenForms.TeamName, sendingBetweenForms.Position, sendingBetweenForms.age);
    form.ShowDialog();
}
}
}

```

Form2 sınıfı:

```

public partial class Form2 : Form
{
    public void Form2Method(string nameSurname,string takim,string position,int
age)
    {

```

```

        lstBxForm2.Items.Add(nameSurname + takim + position + age);
    }
    public Form2()
    {
        InitializeComponent();
    }

    private void Form2_Load(object sender, EventArgs e)
    {

    }

}

```

Ve Main sınıfımız:

```

static class Program
{
    /// <summary>
    /// The main entry point for the application.
    /// </summary>
    [STAThread]
    static void Main()
    {
        Application.EnableVisualStyles();
        Application.SetCompatibleTextRenderingDefault(false);
        Application.Run(new FrmRegistration());
    }
}

```