

Çalar Saat Uygulaması

- **Label**
 - Name: labelCurrentTime
 - Text: "00:00:00" (Başlangıç)
 - Bu Label'da anlık saati göstereceğiz.
- **DateTimePicker**
 - Name: dateTimePickerAlarm
 - Format = Custom, CustomFormat = "HH:mm:ss", ShowUpDown = true
 - Kullanıcının alarm saatini seçmesi için.
- **Button**
 - Name: buttonSetAlarm, Text: "Alarm Kur"
 - Alarm saatini onaylamak için.
- **CheckBox**
 - Name: checkBoxEnableAlarm, Text: "Alarm Açık"
 - İstenirse alarmı devre dışı bırakma/açma amaçlı.
- **Label**
 - Name: labelAlarmStatus, Text: "" (Bilgi amaçlı: "Alarm Kuruldu: xx:xx:xx" şeklinde yazdıracağız.)
- **Timer**
 - Name: timer1
 - Interval = 1000 (1 saniye)
 - Enabled = true (veya kodda açabilirsiniz)

Ek olarak, proje klasörünüze veya belirli bir dizine **.wav** ses dosyası ekleyiniz (örneğin "alarm.wav").

Form1.cs Kodu (Örnek)

```
using System;
```

```
using System.Windows.Forms;
```

```

using System.Media; // SoundPlayer için gerekli
namespace AlarmClockApp
{
    public partial class Form1 : Form
    {
        private DateTime alarmTime; // Kullanıcının seçtiği alarm saati
        private bool alarmSet = false; // Alarm kurulu mu değil mi
        private SoundPlayer alarmSound; // Ses oynatıcı
        public Form1()
        {
            InitializeComponent();

            // Ses dosyasının tam yolunu girmeniz gerekir:
            alarmSound = new SoundPlayer(@"C:\AlarmSesleri\alarm.wav");
        }

        private void Form1_Load(object sender, EventArgs e)
        {
            // Timer ayarları (Properties'te de yapabilirsiniz)
            timer1.Interval = 1000; // 1 saniye
            timer1.Enabled = true;
            timer1.Tick += Timer1_Tick;

            // DateTimePicker ayarları
            dateTimePickerAlarm.Format = DateTimePickerFormat.Custom;
            dateTimePickerAlarm.CustomFormat = "HH:mm:ss";
            dateTimePickerAlarm.ShowUpDown = true; // Sadece saat-dakika-saniye
            seçilsin

            labelCurrentTime.Text = DateTime.Now.ToString("HH:mm:ss");
            labelAlarmStatus.Text = "Henüz alarm kurulmadı.";
        }
    }
}

```

```
private void Timer1_Tick(object sender, EventArgs e)
{
    // Her saniye geçerli saati göster
    labelCurrentTime.Text = DateTime.Now.ToString("HH:mm:ss");

    // Alarm kontrolü
    if (alarmSet && checkBoxEnableAlarm.Checked)
    {
        // Sadece saat/dakika/saniye karşılaştırıyoruz (bugüne alarm kuruldu varsayımı)
        DateTime now = DateTime.Now;
        // Alarm saatine ulaşıldı mı?
        if (now.Hour == alarmTime.Hour &&
            now.Minute == alarmTime.Minute &&
            now.Second == alarmTime.Second)
        {
            // Alarm çalsın
            alarmSet = false; // Tek sefer çalması için alarmı resetliyoruz
            try
            {
                alarmSound.Play(); // WAV dosyasını çal
            }
            catch (Exception ex)
            {
                MessageBox.Show("Ses dosyası çalınırken hata oluştu: " + ex.Message);
            }
            // Ekranda uyarı
            MessageBox.Show("Alarm Zamanı Geldi!", "Alarm",
```

```

        MessageBoxButtons.OK, MessageBoxIcon.Exclamation);
        labelAlarmStatus.Text = "Alarm çaldı! Tekrar kurabilirsiniz.";
    }
}
}
private void buttonSetAlarm_Click(object sender, EventArgs e)
{
    // dateTimePickerAlarm'dan seçilen saat/dak/saniyeyi alalım
    DateTime selectedTime = dateTimePickerAlarm.Value;

    // Bugünün tarihi + seçilen saat
    // (Yalnızca saat/dk/sn kullanıyoruz, tam tarih gerekirse
    // secondDateTimePicker vb. eklemeniz lazım)
    alarmTime = new DateTime(
        DateTime.Now.Year,
        DateTime.Now.Month,
        DateTime.Now.Day,
        selectedTime.Hour,
        selectedTime.Minute,
        selectedTime.Second
    );
    alarmSet = true;
    labelAlarmStatus.Text = "Alarm Kuruldu: " + alarmTime.ToString("HH:mm:ss");
}
}
}

```

Kod Açıklaması

1. SoundPlayer

- using System.Media; ekleyerek SoundPlayer alarmSound nesnesi oluşturuyoruz.
- Kurucu metotta (Form1 constructor) ses dosyasının yolunu parametre olarak veriyoruz.
- alarmSound.Play(); çağrısı ile .wav dosyası çalınır.

2. Timer

- 1 saniye aralıkla (Interval = 1000) Tick olayı çalışarak label'a geçerli saati yazıyor.
- Alarm aktif (alarmSet = true) ve checkBoxEnableAlarm.Checked == true ise, geçerli zaman DateTime.Now ile **alarmTime** eşit olduğunda alarm devreye giriyor.

3. Alarm Kurma

- Kullanıcı, dateTimePickerAlarm aracılığıyla bir saat-dakika-saniye seçer.
- "Alarm Kur" butonuna basınca, bugünün tarihi + seçilen saat/dakika/saniye birleşimiyle alarmTime oluşturuluyor.
- alarmSet = true yaparak alarmın "kurulu" olduğu işaretleniyor.
- labelAlarmStatus metnine de bilgi yazıyoruz (ör. "Alarm Kuruldu: 07:15:00").

4. Alarm Çalınca

- alarmSound.Play(); ile .wav dosyasını oynatır, MessageBox.Show ile kullanıcıyı uyarır.
- Tek seferlik çalması için alarmSet = false; yapıyoruz.
- İsterseniz kaldırıp sürekli aktif kalmasını sağlayabilir veya "tekrar kurma (snooze)" için süre ekleyebilirsiniz.

5. Ses Dosyası Yolu

- "C:\\AlarmSesleri\\alarm.wav" gibi tam yol vermelisiniz. Proje dizini içindeyse "alarm.wav" olarak da kullanılabilir; ancak göreceli yolun çalışıp çalışmadığını doğrulamak gerekir.