

Calculate Time Difference Between Two Dates in UiPath

Dates and **file** comparisons may be required during the project design process.

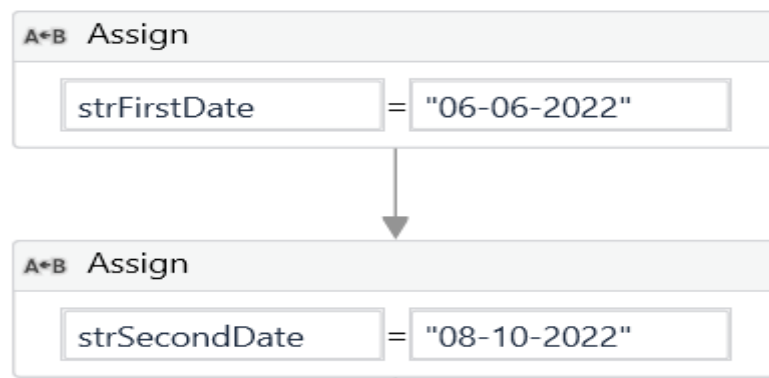
In this post, we'll compare **two dates** and show **how to calculate the day difference** using an **example**.

In the first step, we need to have **two date-containing variables**.

In this example, I chose to do it using **two string variables**.

Because; In this way, we will see **how to convert the string structure to datetime** and learn the general structure.

Let's make our **variables** like this:



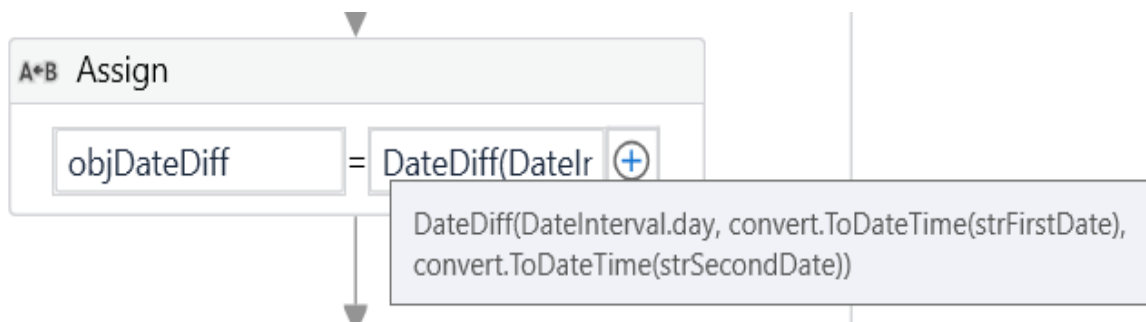
Note—When we write the dates as strings, it defines them as “**Month-Day-Year**”, but you can change the writing format of this. I will explain this in another article.

Let's go ahead and look at the **syntax**.

If we are going to use the dates in **two Datetime variables**;

DateDiff(DateInterval.Day, FirstDate, SecondDate)

If we are going to use the dates in **two String variables** as in our **example**;



**DateDiff(DateInterval.Day, convert.ToDateTime(strfirstDate),
convert.ToDateTime(strSecondDate))**

Let's explain;

objDateDiff— We have defined this variable as object, where we will compare two dates.

DateDiff—is the syntax we use for date differences.

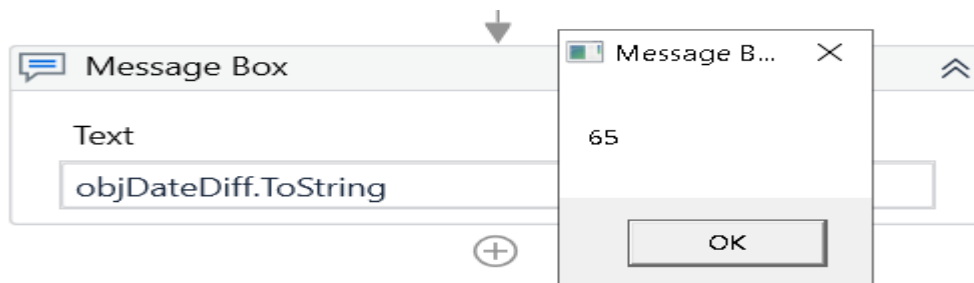
DateInterval.Day— Represents the time **interval**, and you can display this interval any way you want. **In other words**, you can show the time difference between two dates with the following **different structures**.

- **.Day**
- **.Hour**
- **.Minute**
- **.Month**
- **.Second**
- **.Year**

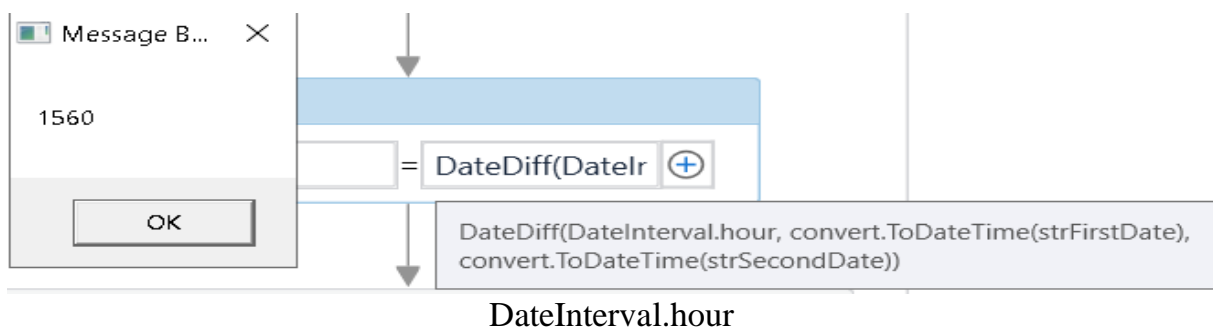
Convert.ToDateTime(strFirstDate)— Here we are converting the date we wrote in the string to DateTime variable. “Convert” does the conversion. “ToDateTime” leads. So it asks to convert it to DateTime. and our String variable is written in parentheses.

As a result **Output;**

Day;



Hour(Bonus);



Thank you for reading,

Akın Cömertpay

Uipath'te İki Tarih Arasındaki Zaman Farkı Hesaplama

Proje dizayn aşamasında dosyaların tarihleri ve karşılaştırmaları yapılması gereken durumlara ihtiyaç duyulabiliyor.

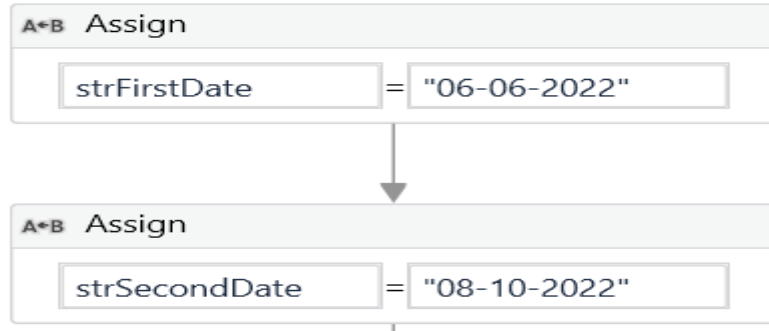
Bu yazımda iki tarihin karşılaştırmasını yapıp, gün farkını hesaplama örneğini yapacağız.

İlk aşamada iki tane tarih içeren değişkenimiz olması gerekli.

Ben bu örnekte **iki string** değişkeni kullanarak yapmayı tercih ettim.

Çünkü; bu yolla hem **string** yapısını **datetime** a çevirmeyi göreceğiz hemde **genel yapıyı** öğreneceğiz.

Değişkenlerimiz böyle olsun;



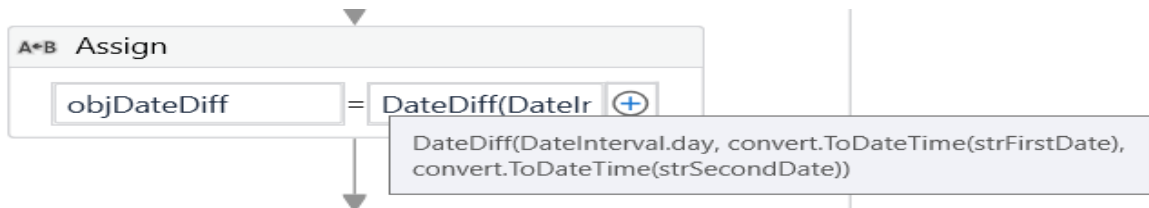
Not olarak—Tarihleri **string** olarak yazdığımızda “ **Ay-Gün-Yıl** “ şeklinde tanımlıyor ama bunun yazılış formatını değiştirebilirsiniz. Bunu başka bir yazı da anlatacağım.

Devam edelim ve **söz dizimine** bakalım.

Eğer tarihleri iki **Datetime** değişkeni içerisinde **kullanacaksak;**

DateDiff(DateInterval.Day, FirstDate, SecondDate)

Eğer **örneğimizdeki** gibi tarihleri iki **String** değişkeni içerisinde **kullanacaksak;**



**DateDiff(DateInterval.Day, convert.ToDateTime(strfirstDate),
convert.ToDateTime(strSecondDate))**

Açıklayalım;

objDateDiff—İki tarihin karşılaşmasını yapacağımız bu değişkenimizi **object** olarak tanımladık.

DateDiff—Tarih farklılıkları için kullandığımız söz dizimidir.

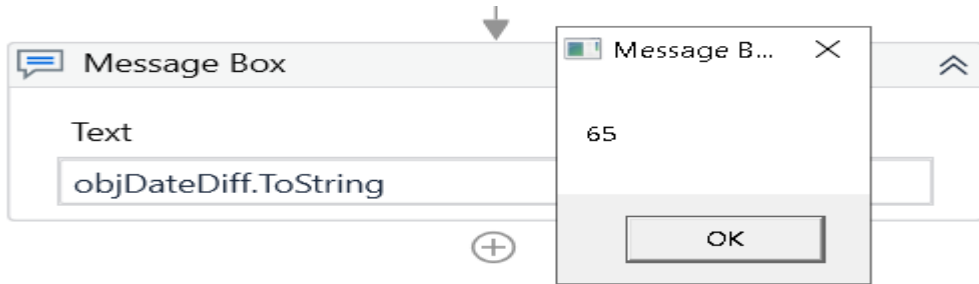
DateInterval.Day— Zaman aralığını ifade eder, ve bu zaman aralığını istediğiniz şekilde gösterebilirsiniz. Yani iki tarih arasındaki zaman farkını aşağıdaki farklı yapılarla da gösterebilirsiniz.

- **.Day**
- **.Hour**
- **.Minute**
- **.Month**
- **.Second**
- **.Year**

Convert.ToDateTime(strFirstDate)— Burada string içerisinde yazdığımız tarihimizi DateTime değişkenine çevirme işlemini yapıyoruz. “**Convert**” çevirme işlemini yapar. “**ToDateTime**” yol gösterir. Yani **DateTime** a çevirmesini ister. ve Parantez içine de **String** değişkenimiz yazılır.

Sonuç olarak **Output**;

Gün olarak;



Saat olarak(Bonus);

