

TAlarm Component

[Properties](#)

[Methods](#)

[Events](#)

The TAlarm component let's you specify a predefined time as an alarm time. When the time has expired, the AlarmNotify event is fired along with any accompanying code you have included in the event handler. You can perform a series of tasks or simply notify the user that time has expired.. Use your imagination !

Properties

[AlarmTime](#)

[Enabled](#)

[Interval](#)

AlarmTime Property

Description

This is a TDateTime property that indicates the time you want to trigger the alarm. The alarm 'goes off' by triggering the AlarmNotify event handler. There you can put any code you need to alert the user of the alarm status.

See Also:

[AlarmNotify event](#)

[SetAlarm method](#)

Example:

```
Alarm1.Alarmtime := strtotime('12:00');  
Alarm1.enabled := true;  
Alarm1.SetAlarm;
```

The above example sets the alarm for noon.

AlarmNotify Event

Description

The AlarmNotify event is fired whenever the system time has reached the time specified for the Alarmtime property. The event has one parameter which is the system time when the event fired. This may or may not be the same as the Alarmtime property.

Example:

```
procedure TDemoAlarm.Alarm1AlarmNotify(const ATime: TDateTime);  
begin  
    showMessage('Time is up');  
end;
```

SetAlarm Method

Description

The SetAlarm method actually enables the alarm component and starts the timer. When the desired time is reached the component disables the timer and fires the AlarmNotify event. If the Alarmtime is not a valid TDateTime value an error message will be displayed.

See Also:

[AlarmNotify event](#)

Enabled Property

Description

This is the standard enabled property available to almost every Delphi component. At the time the AlarmNotify event is fired, the enabled property has been set to false, however, the call to the SetAlarm method sets the enabled property to true, so you will not need to do this in code.

Interval Property

Description

The alarm component is driven by its own timer. The frequency at which it examines the time and compares to the Alarmtime property is governed by the interval property. The interval property is the same as for a TTimer object.

Example:

```
Alarm1.alarmtime := strtotime('12:00');  
Alarm1.interval := 2000; {check every 2 secs}  
Alarm1.SetAlarm;
```


