

## 20.2 Balloon

Die Klasse *Balloon* (*gb.form*) implementiert ein Anzeige-Objekt für Nachrichten. Diese Klasse zeigt eine Nachricht und ein optionales Icon in einem farbigen Ballon an.

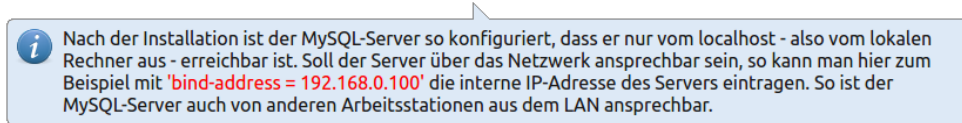


Abbildung 20.2.1: Ballon mit Icon 'Information' und farbig ausgezeichnetem Text-Abschnitt

### 20.2.1 Eigenschaften

Die Klasse *Balloon* verfügt über vier Eigenschaften:

Eigenschaft	Datentyp	Beschreibung
Control	Control	Es wird NULL zurückgegeben, wenn kein Ballon angezeigt wird oder das Steuerelement, mit dem das Balloon-Objekt assoziiert ist.
Delay	Integer	Ermittelt oder legt die Anzahl der Millisekunden fest, wie lange der Ballon angezeigt wird. Die Standard-Zeit sind 5 Sekunden.
Font	Font	Gibt den Font zurück oder setzt den Font für den Ballon.
Visible	Boolean	Gibt TRUE zurück, wenn der Ballon sichtbar ist.

Tabelle 20.2.1.1 : Eigenschaften der Klasse Balloon

### 20.2.2 Methoden

Die Klasse *Balloon* hat 2 grundlegende Methoden – eine zum Anzeigen des Ballons und eine zum Verstecken des Ballons. Bei der Anzeige können *optional* 5 verschiedene Icon mit angezeigt werden, die den Typ der Mitteilung (Fehler, Frage, Information, Löschen und Warnung) kennzeichnen.

Methode	Beschreibung
Balloon ( Message As String, Control As Control [ , X As Integer, Y As Integer ] )	Zeigt die Mitteilung in einem Ballon mit farbigem Hintergrund an.
Balloon.TYPE ( Message As String, Control As Control [ , X As Integer, Y As Integer ] )	Zeigt die Mitteilung in einem Ballon mit farbigem Hintergrund an, dem ein Icon vorangestellt wird.
Hide ( [ Control As Control ] )	Der Ballon wird versteckt. Sie können optional die Komponente angeben, auf die der Ballon zeigt.

Tabelle 20.2.2.1 : Methoden der Klasse Balloon

- *Message* ist die anzuzeigende Mitteilung. Sie können den Text vom Mime-Typ 'text/html' auch auszeichnen → Beispiel 1.
- *Control* ist die Komponente, auf die der Ballon zeigt.
- *X* und *Y* sind die (relativen) Koordinaten des Punktes auf der Komponente, auf die der Ballon zeigt. Die Angabe der Koordinaten ist optional.
- *TYPE* gibt optional den Typ der Mitteilung an: Delete, Info, Error, Question und Warning. Ohne die Typ-Angabe wird nur der Mitteilungstext angezeigt.

### 20.2.3 Hinweise zur Verwendung der Klasse Balloon

- Für kleine, auf eine Komponente bezogene Hinweise eignet sich als Alternative der Einsatz der ToolTip-Eigenschaft.
- Für längere Mitteilungen stellt die Klasse *Message* sechs unterschiedlichen Varianten zur Verfügung → Kapitel 20.1 Message.
- Für ausführliche Mitteilungen – zum Beispiel im Rahmen einer Programm-Hilfe – stehen Ihnen weitere Anzeige-Objekte zur Verfügung, deren Übersicht Sie im → Kapitel 11.8 Programmhilfen nachlesen können. Dort finden Sie auch Anwendungshinweise und viele Beispiele.

### 20.2.4 Beispiele

Die folgenden zwei Beispiele zeigen typische Einsatzfälle für Ballons.

#### Beispiel 1

Das Beispiel 1 bezieht sich auf die → Abbildung 20.2.1 zur Anzeige eines Ballons mit Icon. Der folgende Quelltext-Ausschnitt ist typisch für den Einsatz eines Ballons:

```
[1] bbPublic Sub btnHilfe_Click()
[2]   Dim sMitteilung As String
[3]
[4]   Balloon.Font = Font["Ubuntu,12"]
[5]   Balloon.Delay = 10000
[6]
[7]   sMitteilung = "Nach der Installation ist der MySQL-Server so konfiguriert, dass er nur vom "
[8]   sMitteilung &= "localhost - also vom lokalen Rechner aus - erreichbar ist. Soll der Server "
[9]   sMitteilung &= "über das Netzwerk ansprechbar sein, so kann man hier zum Beispiel mit "
[10]  sMitteilung &= "<font color='red'>'bind-address = 192.168.0.100' </font>die interne "
[11]  sMitteilung &= " IP-Adresse des Servers eintragen. So ist der MySQL-Server auch von anderen "
[12]  sMitteilung &= " Arbeitsstationen aus dem LAN ansprechbar."
[13]
[14]  If Not Balloon.Visible Then
[15]    Balloon.Info(sMitteilung, btnHilfe, btnHilfe.W - 5, btnHilfe.H - 3)
[16]  Else
[17]    Balloon.Hide
[18]  Endif ' If Not Balloon.Visible ?
[19] End ' btnHilfe_Click()
```

#### Hinweise:

- Die Anzeige des (Hilfe-)Textes erfolgt mit einem Einfach-Klick auf den Button 'btnHilfe' und ist diesem Button zugeordnet → Zeile 15 Mitte.
- Zusätzlich ist auch die Verwendung eines Tool-Tips für diesen Button möglich, da das Ereignis btnHilfe\_Enter() nicht verwendet wird.
- Es wird durch *Balloon.Info(...)* in der Zeile 15 auch ein Icon angezeigt.
- Sie können dem Ballon optional einen *eigenen* Font [4] zuordnen, der immer ein Kompromiss zwischen Lesbarkeit und (Anzeige-)Größe sein wird. Sonst wird der Standard-Font des Projekts verwendet.
- Die Anzeige-Dauer von 10 Sekunden [5] scheint sehr groß zu sein. Bedenken Sie aber, dass der Text schon ungewöhnlich lang ist. Sie können aber die Anzeige mit einem weiteren Klick auf den Button 'btnHilfe' sofort beenden, wie Sie der Kontroll-Struktur in den Zeilen 14 bis 18 entnehmen.

#### Beispiel 2

Setzen Sie einen Ballon auf dem Formular ein – hier gebunden an einen kleinen Button neben einer Textbox mit einem ⓘ als Symbol – um *dezent* einen Hinweis anzuzeigen. Der Ballon mit dem Hinweis-text wird angezeigt, wenn Sie mit der Maus über dem Button mit dem Icon stehen. Der Ballon verschwindet nach 3 Sekunden automatisch oder genau dann, wenn Sie mit der Maus den Button innerhalb der Anzeigezeit wieder verlassen:

```
Public Sub btnInformation_Enter() ' Ballon anzeigen
  Dim sMitteilung As String

  Balloon.Font = Font["Ubuntu,11"]
  Balloon.Delay = 3000
  sMitteilung = "Eingabe in GROSSBUCHSTABEN erforderlich!" & gb.Tab
  Balloon.Info(sMitteilung, Last)
End ' Sub btnInformation_Enter()

Public Sub btnInformation_Leave() ' Ballon verstecken
  Balloon.Hide
End ' btnInformation_Leave()
```