

27. Some new objekts

⇒ MainMenu



- befindet sich in der Objektpalette „Standard“.
- realisiert ein Pulldown-Menü. Egal wo das MainMenu-Symbol auf dem Formular ist: Das Mainmenü befindet sich immer oben am Formularrand.
- Pro Formular kann nur ein MainMenü verwendet werden („es kann nur Einen geben“).
- Die Menüstruktur wird mit dem Menüdesigner entwickelt (Doppelklick oder rechte Maustaste; die Beschriftungen der Menüpunkte stehen unter den Captions). *Einfügen, Löschen, Umsortieren und Untermenüs sind möglich!*
- Ein einzelner Bindestrich wird als Trennlinie angezeigt.

⇒ PopupMenu



- befindet sich ebenfalls in der Objektpalette „Standard“.
- realisiert ebenfalls ein Pulldown-Menü; für die Erstellung siehe „MainMenu“.
- Popup-Menüs erscheinen nur, wenn der Anwender beim Programmablauf über einem Objekt die rechte Maustaste drückt, sind sonst also unsichtbar.
- Bei diesem Objekt muss die Eigenschaft PopupMenu auf das PopupMenu zeigen, also dessen Namen enthalten. *Den muss man übrigens nicht selbst eintippen: Besagte Eigenschaft bietet ein Pulldown-Menü mit existierenden Popup-Menüs an zum Auswählen.*

⇒ UpDown




- befindet sich in der Objektpalette „Win32“.
- besteht aus 2 Buttons (Orientation horizontal oder vertikal), die beim Drücken den numerischen Wert in Position hoch oder herunterzählen.
- der aktuelle Wert steht wie gesagt in der Eigenschaft Position.
- Die Eigenschaften Min und Max begrenzen den einstellbaren Bereich.
- Die Eigenschaft Increment legt fest, um welchen Betrag der Wert der Eigenschaft Position geändert wird, wenn die Tasten gedrückt werden.

⇒ ColorDialog



- befindet sich in der Objektpalette „Dialoge“.
- Zur Laufzeit wird das Windows-Dialogfeld zur Farbauswahl erst dann eingeblendet, wenn es durch einen Aufruf der Methode *Execute* aktiviert wird. Wenn der Benutzer nach der Auswahl einer Farbe auf OK klickt, wird das Dialogfeld geschlossen und die gewählte Farbe in der Eigenschaft Color gespeichert.
- *Execute* öffnet das Dialogfeld zur Farbauswahl und gibt *true* zurück, wenn der Benutzer eine Farbe auswählt und auf OK klickt. Bricht der Benutzer ab, wird *false* zurückgegeben. *Typisch ist deshalb die Programmzeile* **IF ColorDialog1.Execute THEN ...**

28. Hinzufügen von Formularen zum Projekt

Wähle den Menüpunkt **Datei | Neues Formular** oder auch den Speedbutton  (3. Zeile, 4. von links). Ist ein neues Formular ausgewählt, so wird automatisch eine dazugehörige Unit erzeugt. Am oberen Rand der Units befinden sich jetzt kleine Laschen, um zwischen den verschiedenen Units zu wechseln; mit <F12> lässt's sich bekanntlich fix zwischen einer Unit und seinem Formular wechseln.

- Sind die Laschen irgendwann weg ☹, bevorzugt nach dem Neu-Laden des Programms, so können sie über **Ansichten | Units** wieder sichtbar gemacht werden ☺.
- Delphi ist etwas umständlich, deshalb muss immer, wenn von einer Unit auf eine andere Bezug genommen werden, bestätigt werden, dass man das auch wirklich will. Zu allem Überfluss muss das Programm dann noch einmal gestartet werden (natürlich nicht ganz Delphi, nur *dein* Programm...).
- Scheitert der Programmstart an der merkwürdigen Meldung „... kann nicht erstellt werden“, so klick auf „alles speichern“. Danach weiß auch Delphi, wo er die kompilierte neue Unit hinspeichern darf.

Zeigen und verstecken eines Formulars funktioniert wie bei jedem anderen Objekt auch, nämlich über den Wert der Eigenschaft *Visible*, ggf. auch mit den Methoden Show bzw. Hide.

Deutlich besser ist hier allerdings die Methode **ShowModal**. Wenn damit ein Formular angezeigt wird, kann das Programm erst fortgesetzt werden (auf einem anderen Formular), wenn dieses Formular geschlossen wurde. Dies geschieht, indem seine Eigenschaft *ModalResult* auf einen Wert ungleich Null gesetzt wird.