

Einführung in die komponentenorientierte Programmierung mit Borland-Delphi

Übung 12

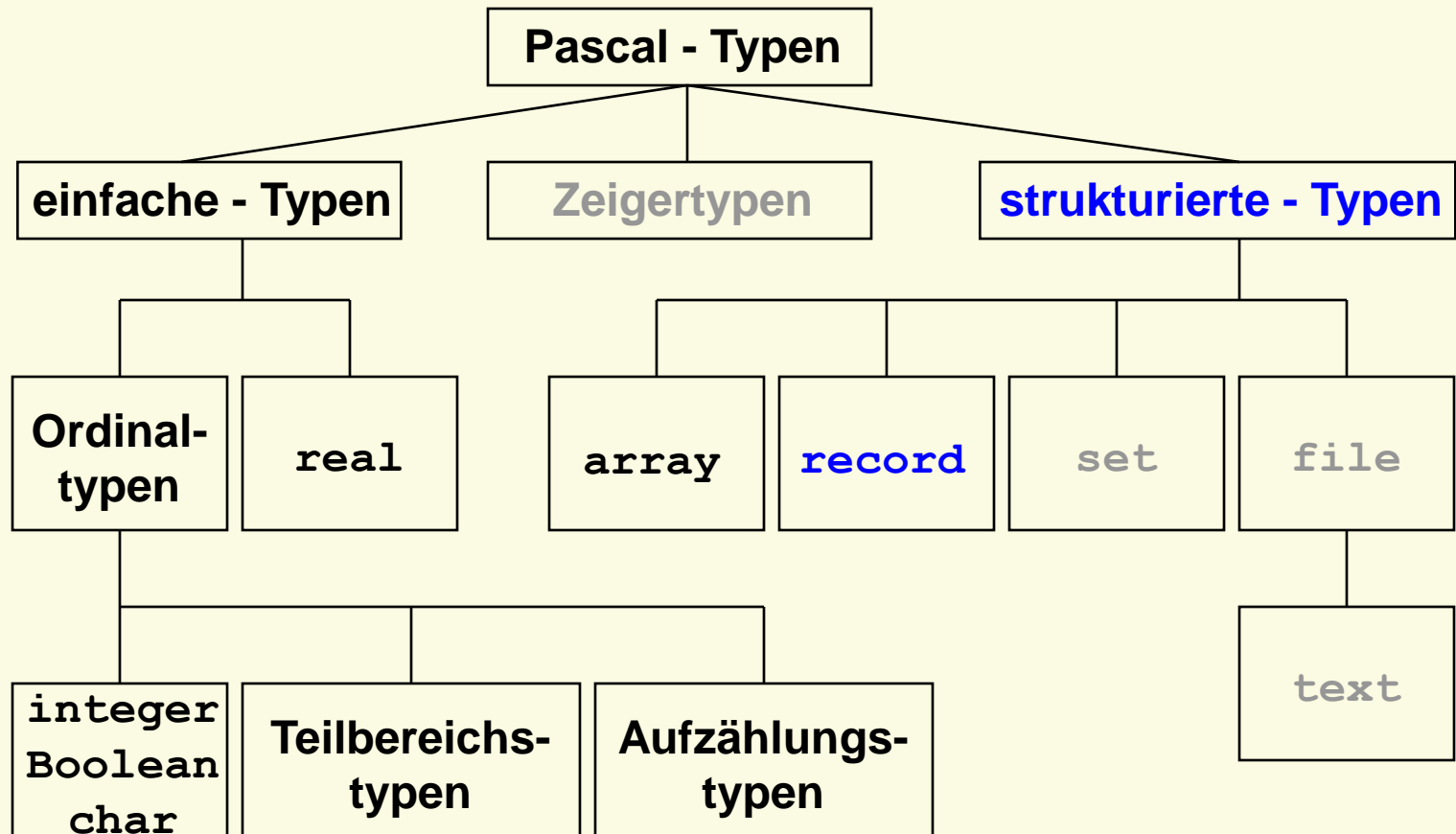
Strukturierte Datentypen - Datensätze

Dr. Henry Herper

Otto-von-Guericke-Universität Magdeburg

Institut für Simulation und Graphik

Typ-Terminologie in Pascal



Datensätze - Records

Der RECORD-Typ dient dazu, eine **festen Anzahl von Objekten beliebigen Typs** zu einer Datenstruktur zusammenzufassen. Die Records gehören zu den Verbundtypen. Sie bestehen aus einer **festen Anzahl von Komponenten**, den Datenfeldern.

Die einzelnen **Datenfelder** können von unterschiedlichem Typ sein, und jedes Datenfeld hat einen eigenen Datenfeldbezeichner, der dem Zugriff auf dieses Datenfeld dient. Der Datenfeldbezeichner hat nur innerhalb des Records Gültigkeit.

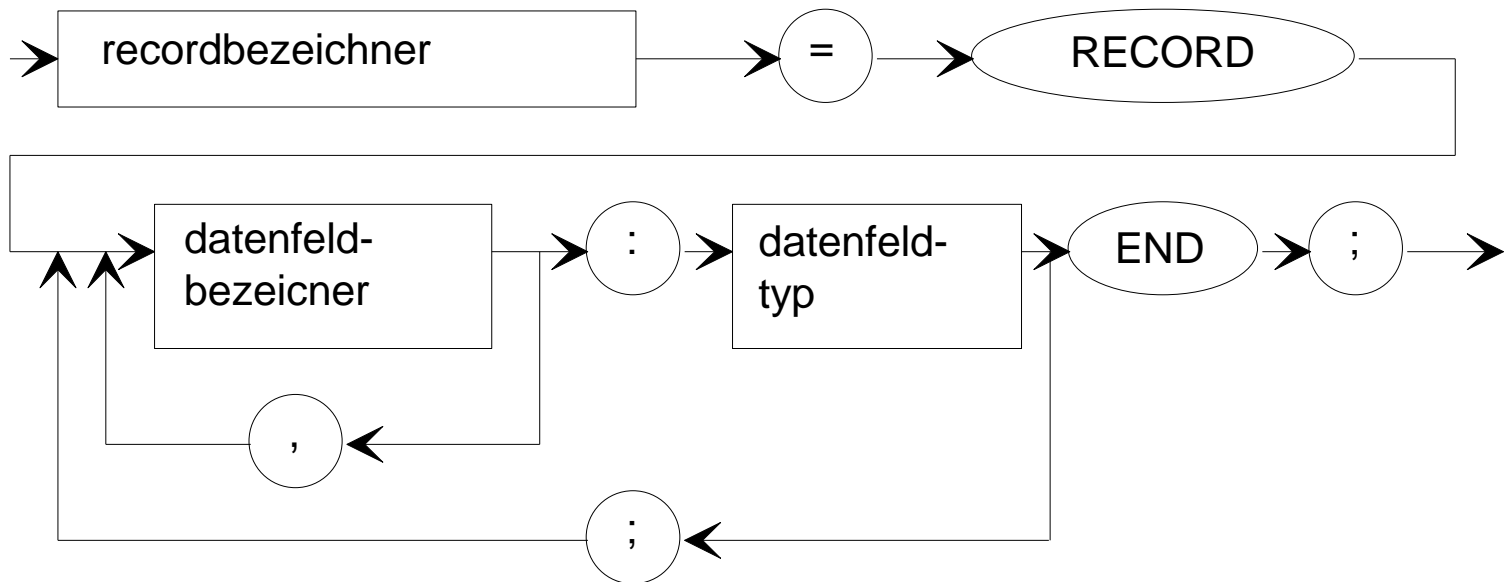
Datensätze - Records

Die Anzahl und der Typ der Elemente eines Records wird bei der Definition festgelegt und ist danach im Programm nicht mehr veränderbar. Die Deklaration des Record-Typs kann innerhalb des Typ- oder Variablendeklarationsteils erfolgen.

Der Datensatz wird mit dem Schlüsselwort **RECORD** definiert, gefolgt von Namen und Typ der einzelnen Elemente des RECORDs. Die Definition ist mit einem **END** abzuschließen.

Datensätze - Recorddeklaration

RECORD - Definition:



Datensätze - Records

Definition eines Typs Datum im
Typendeklarationsteil

```
type datum = record
    tag      : 1..31;
    monat    :
(jan, feb, mrz, apr, mai, jun,
jul, aug, sep, okt, nov, dez) ;
    jahr     : 1950..2050;
end;
```

Definition der Variablen gebtag, arbeitsbeginn
vom Typ datum

```
var gebtag, arbeitsbeginn : datum;
```

Datensätze - Zugriffe

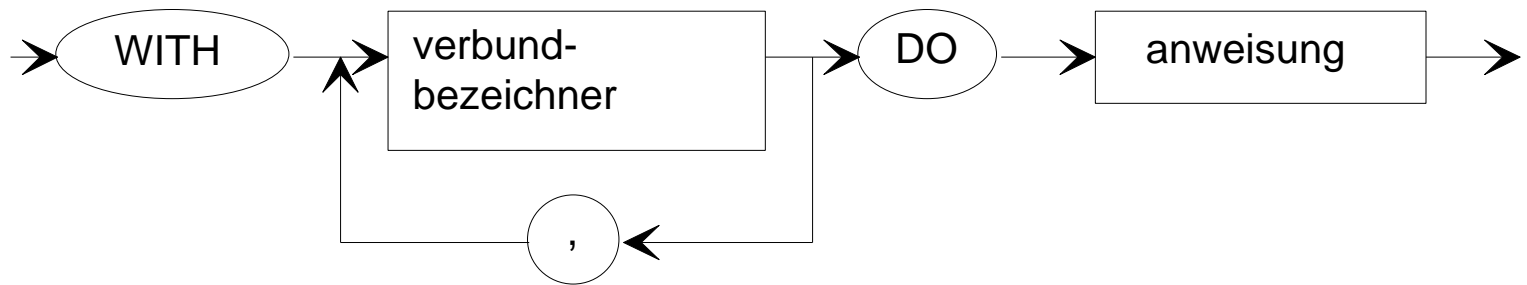
Den Zugriff zu einem **Datenfeld** erhält man durch den **Datenfeldbezeichner**, den man **getrennt durch einen Punkt** dem Variablenbezeichner für diesen Datensatz voranstellt.

Beispiele:

```
gebtag.tag      := 22;  
gebtag.monat   := mai;  
gebtag.jahr    := 1978;  
adr.name       := 'Mueller';  
adr.vname      := 'Max';
```

Datensätze – Zugriffe – with-Anweisung

with-Anweisung:



HA 12 - Lagerverwaltung

Schreiben Sie ein Struktogramm und eine Delphi-Applikation zur Verwaltung von Artikeln eines Lagers. Für jeden Artikel ist ein Datensatz zu erstellen. Jeder Datensatz hat folgende Struktur:

- Artikelnummer,
- Artikelname und
- aktueller Warenbestand.

Das Programm soll in der Lage sein,

- Datensätze einzulesen,
- Datensätze zu verändern und
- Datensätze auszugeben.

Die Auswahl des Datensatzes soll über die Artikelnummer erfolgen. Die Verwaltung der Datensätze erfolgt in einem Feld.

Aufgabe 12 – komplexe Zahlen

Schreiben Sie eine Delphi-Applikation zur Implementierung der algebraischen Grundfunktionen zum Rechnen mit komplexen Zahlen. Die Zahlen werden in algebraischer Darstellung erfasst (2 Eingabefenster) und in einem 3. Fenster ausgegeben. Die Operationen werden über Schalter eingegeben. Die Zahlen sind in Records zu verwalten.

Rechenregeln für $z_1 = a_1 + b_1i$ und $z_2 = a_2 + b_2i$

$$z_1 \pm z_2 = (a_1 \pm a_2) + (b_1 \pm b_2)i$$

$$z_1 * z_2 = (a_1 * a_2 - b_1 * b_2) + (a_1 * b_2 + a_2 * b_1)i$$

$$z_1 / z_2 = \left(z_1 * \overline{z_2} \right) / \left(z_2 * \overline{z_2} \right)$$

$$\overline{z} = a - bi$$

Kontrollfragen

1. Beschreiben Sie das Konzept der Datensätze in Objekt-Pascal. Welche Daten lassen sich in Datensätzen verwalten? Wie werden Datensätze deklariert? Beschreiben Sie die Möglichkeiten auf Datensätze bzw. auf ihre Komponenten zuzugreifen. Deklarieren Sie einen Beispieldatensatztyp zur Verwaltung von Adressen.
2. Welche Erweiterungen bieten Records mit variantem Anteil? Wie wird der variable Anteil deklariert?
3. Welche Datenstrukturen können in Records verwaltet werden?
4. Welche Funktion hat die WITH-Anweisung?