

8. Übungsblatt zu Programmiersprachenkonzepte

Manuel Schwarz, Michael Stypa

12. Januar 2011

Aufgabe 8.1

```
program aufg1;
type
  typeA = record
    m : integer;
  end;
  typeB = record
    m : integer;
  end;
var
  A : typeA;
  B : typeB;
  C : typeA;
begin
  B.m := 6;
  A := B; {Fehler: keine strukturelle Typgleichheit }
  C.m := -2; {kein Fehler: namentliche Typengleichheit}
end.
```

Pascal ledgt namentliche jedoch keine strukturelle Typengleichheit zugrunde.

Aufgabe 8.2

```
1. program kartei;
   type
     kraftstoffart = (Gas, Diesel, Benzin, Elektro, Hybrid);
     gebrauchtwagenrecord = record
       marke: string;
       modell: string;
       unfall: boolean;
       kraftstoff: kraftstoffart;
       erstzulassung : integer;
       kilometerstand : longint;
       preis : longint;
       plz : longint;
```

```

    end;

var
    garage          : array[1..2] of gebrauchtwagenrecord;
    i                : integer;
    option           : char;

function gebrauchtwagenHinzufuegen:gebrauchtwagenrecord;
var
    gebrauchtwagen : gebrauchtwagenrecord;
    wronginput      : boolean;
    key             : char;

begin
    writeln('Gebrauchtwagen-Eintrag');

    writeln('=====');

    write('Marke: ');
    readln(gebrauchtwagen.marke);

    write('Modell: ');
    readln(gebrauchtwagen.modell);

    repeat
        write('Unfallwagen? (j)a, (n)ein: ');
        readln(key);
        wronginput := false;
        Case key of
            'j' : gebrauchtwagen.unfall := true;
            'n' : gebrauchtwagen.unfall := false;
        else
            wronginput := true;
        end;
    until (not(wronginput));

    repeat
        write('Kraftstoffart (g)as, (d)iesel, (b)enzin, (e)lektro, (h)ybrid: ');
        readln(key);
        wronginput := false;
        Case key of
            'g' : gebrauchtwagen.kraftstoff := Gas;
            'd' : gebrauchtwagen.kraftstoff := Diesel;
            'b' : gebrauchtwagen.kraftstoff := Benzin;
            'e' : gebrauchtwagen.kraftstoff := Elektro;

```

```

        'h' : gebrauchtwagen.kraftstoff := Hybrid;
    else
        wronginput := true;
    end;
until (not(wronginput));

writeln('Kraftstoffart ', gebrauchtwagen.kraftstoff);

repeat
    write('Jahr der Erstzulassung(ab 1980): ');
    readln(gebrauchtwagen.erstzulassung);
until ((gebrauchtwagen.erstzulassung >= 1980)
    and (gebrauchtwagen.erstzulassung <= 2011));

repeat
    write('Kilometerstand: ');
    readln(gebrauchtwagen.kilometerstand);
until (gebrauchtwagen.kilometerstand >= 0);

repeat
    write('Preis: ');
    readln(gebrauchtwagen.preis);
until (gebrauchtwagen.preis >= 0);

repeat
    write('PLZ des Wohnorts des Verkäufers: ');
    readln(gebrauchtwagen.plz);
until ((gebrauchtwagen.plz >= 0)
    and (gebrauchtwagen.plz <= 99999));
gebrauchtwagenHinzufuegen := gebrauchtwagen;
end; { gebrauchtwagenHinzufuegen }

procedure ausgabeAll;
var
    j : integer;
begin
    for j:=1 to i-1 do
        begin
            writeln();
            writeln('Neuer Gebrauchtwagen-Eintrag');
            writeln('=====');
            writeln(garage[j].marke, ' - ', garage[j].modell);
            if(garage[j].unfall) then
                begin
                    writeln('Unfallwagen.');
```

```

        end;

        write(garage[j].kraftstoff, '-Fahrzeug Baujahr ');
        writeln(garage[j].erstzulassung);
        write(garage[j].kilometerstand, 'km gelaufen für ');
        write(garage[j].preis);
        writeln('EUR aus PLZ ', garage[j].plz);
    end;
end; { ausgabeAll }


procedure ausgabeMarke(marke : string);
var
    k : integer;

begin
    for k:=1 to i-1 do
        begin
            if(garage[k].marke = marke) then
                begin
                    writeln();
                    writeln('Neuer Gebrauchtwagen-Eintrag');
                    writeln('=====');
                    writeln(garage[k].marke, ' - ', garage[k].modell);
                    if(garage[k].unfall) then
                        begin
                            writeln('Unfallwagen. ');
                        end;

                    write(garage[k].kraftstoff, '-Fahrzeug Baujahr ');
                    writeln(garage[k].erstzulassung);
                    write(garage[k].kilometerstand, 'km gelaufen für ');
                    write(garage[k].preis);
                    writeln('EUR aus PLZ ', garage[k].plz);
                end;
            end;
        end;
    end; { ausgabeMarke }


procedure menu;
begin
    writeln('');
    writeln('===== HAUPTMENUE =====');
    writeln('Bitte wählen Sie eine der folgenden Optionen:');
    writeln('(1) Neuen Wagen eintragen. ');
    writeln('(2) Alle Wagen ausgeben. ');

```

```

        writeln('(3) Alle Wagen einer bestimmten Marke ausgeben.');
```

```

        writeln('(4) Programm beenden');
```

```

end; { menue }
```



```

function readMarke:string;
var
    marke : string;
begin
    writeln('Bitte geben Sie die Marke an, nach der Sie suchen moechten: ');
    readln(marke);
    readMarke:=marke;
end; { readMarke }
```



```

{***** MAIN *****)}
begin
    i := 1;
    repeat
        menue;
        readln(option);
        Case option of
            '1' :
                begin
                    if(i <= 2) then
                        begin
                            garage[i] := gebrauchtwagenHinzufuegen;
                            i:=i+1;
                        end
                    else begin
                        write('Sie koennen keine neuen Wagen mehr in die Liste');
                        write('eintragen, da die Liste voll ist, bitte waehlen Sie ');
                        writeln('eine andere Option.');
```

```

                    end;
                end;
            '2' : ausgabeAll;
            '3' : ausgabeMarke(readMarke);
        end;
    until (option = '4');
```

```

end.
```

2.
 - zubehoer ist Set aus predefmerkmal
 - sonderz ist String-Liste von Sonderzubehöerteilen