

TYP REKORDOWY

→ kilka danych różnego typu, które wspólnie charakteryzują ten sam obiekt

Definicja typu rekordowego:

DANE OSOBOWE	TYPE
Nazwisko	Dane_Osobowe = RECORD
Imię	Nazwisko : string [40];
Wiek	Imię : string [20]
Adres	Wiek : byte ;
Liczba lat pracy	Adres : string ;
	Liczba_lat_pracy : byte ;
	END ;

Deklaracja zmiennych rekordowych:

```
VAR
    kierownik, sekretarka, pracownik_1, pracownik_2 :
Dane_Osobowe;
    spis_osob : array [ 1 .. 100 ] of Dane_Osobowe;
```

Rekordy zagnieżdżone:

```
TYPE
    Dane_adresowe = record
        Miejscowosc : string [20];
        Kod_pocztowy : string [6];
        Ulica_nr_domu : string [30];
    end ;
```

Miejscowosc	Kod_pocztowy	Ulica_nr_domu
Warszawa	34-678	Wielka 4

```
Dane_osobowe = record
    Nazwisko : string [40];
    Imię : string [20];
    Adres : Dane_Adresowe; ← rekord zagnieżdżony
end;
```

NAZWISKO	IMIĘ	ADRES		
		Miejscowosc	Kod_pocztowy	Ulica_nr_domu
Kowalski	Jan	Warszawa	34-678	Wielka 4

Odwoływanie się do pól rekordów:

VAR

osoba : Dane_Osobowe

begin

osoba . Nazwisko := 'Kowalski';

osoba . Imie := 'Jacek';

osoba . Adres . Miejscowosc := 'Wroclaw';

osoba . Adres . Kod_pocztowy := '53-139';

osoba . Adres . Ulica_nr_domu := 'Swidnicka 23a';

end;

Instrukcja wiążąca:

procedure InicjujDane(**var** osoba : Dane_Osobowe);

begin

with osoba **do**

begin

Nazwisko := ' *Kowalski* ' ;

Imie := ' *Jacek* ' ;

Adres . Miejscowosc := ' *Wroclaw* ' ;

Adres . Kod_pocztowy := ' *53-139* ' ;

Adres . Ulica_nr_domu := ' *Swidnicka 23a* ' ;

end;

end;

Inicjowanie zmiennych rekordowych:

TYPE

Punkt = **record**

X, Y: Real;

end;

Miesiace = (Sty,Lut,Mar,Kwi,Maj,Cze,Lip,Sie,Wrz,Paz,Lis,Gru);

Data = **record**

Dzien: 1 .. 31;

Miesiac: Miesiace;

Rok: 1900..1999;

end;

CONST

srodek : Punkt = (X : 0.0; Y: 0.0);

urodziny : Data = (Dzien : 2; Miesiac : Sty; Rok : 1970);

Przypisywanie całych rekordów innym rekordom:

```
osoba_1 . nazwisko := osoba_2 . nazwisko;  
osoba_1 . imie := osoba_2 . imie;  
osoba_1 . adres := osoba_2 . adres;  
osoba_1 := osoba_2;
```

Rekordy jako elementy tablic:

	Nazwisko	Imie	adres		
			Miejscowosc	Kod_pocztowy	Ulica_nr_domu
1	Kowalski	Jan	Warszawa	34-678	Prosta 4
2	Nowak	Stanislaw	Gdansk	25-567	Wodna 2 m 3
3	Sienkiewicz	Henryk	Krakow	12-425	Mariacka 4
4	Mitura	Grzegorz	Poznan	23-123	Dworcowa 1
5	...				
6	...				
7	...				
8	...				
9	...				
10	...				

```
CONST IL_WIERSZY = 10;
```

```
TYPE
```

```
Spis_Osob : array [ 1 .. IL_WIERSZY ] of Dane_Osobowe;
```

```
VAR
```

```
Grupa_Stud : Spis_Osob;
```

```
Bufor : Dane_Osobowe;
```

```
Begin
```

```
.....
```

```
if Grupa_Stud[ i ] . nazwisko > Grupa_Stud[ i+1 ] . nazwisko then
```

```
  begin
```

```
    Bufor := Grupa_Stud[ i ];
```

```
    Grupa_Stud[ i ] := Grupa_Stud[ i+1 ];
```

```
    Grupa_Stud[ i+1 ] := Bufor;
```

```
  end;
```

```
.....
```

```
end;
```

```

program przyklad_0; {Wczytanie i wyświetlenie pojedynczego rekordu}

USES crt;

TYPE {definicje typow}
  dane_personalne = record
    nazwisko : string [30];
    imie : string [15];
    il_lat : integer ;
  end;

VAR {deklaracje zmiennych}
  osoba : dane_personalne;

begin {program glowny}
  clrscr;
  writeln( ' WCZYTYWANIE DANYCH PERSONALNYCH ' );
  write( ' Nazwisko: ' );
  readln( osoba.nazwisko );
  write( '      Imie: ' );
  readln( osoba.imie );
  write( '      Ilosc lat: ' );
  readln( osoba.il_lat );

  writeln( ' WYSWIETLANIE DANYCH PERSONALNYCH ' );
  writeln( ' Nazwisko: ' , osoba.nazwisko );
  writeln( '      Imie: ' , osoba.imie );
  writeln( '      Ilosc lat: ' , osoba.il_lat );
  writeln( ' Nacisnij ENTER aby zakonczyc program ' );
  readln;
end.

```

program przyklad_1; *{Wczytanie i wyświetlenie pojedynczego rekordu}*

USES crt;

TYPE *{definicje typow}*

```
dane_personalne = record
    nazwisko : string [30];
    imie : string [15];
    il_lat : integer ;
end;
```

procedure WCZYTAJ(**var** dane_osoby : dane_personalne);

begin

```
    clrscr;
    writeln( ' WCZYTYWANIE DANYCH PERSONALNYCH ' );
    writeln;
    write( ' Nazwisko: ' );
    readln( dane_osoby.nazwisko );
    write( '      Imie: ' );
    readln( dane_osoby.imie );
    write( '  Ilosc lat: ' );
    readln( dane_osoby.il_lat );
```

end; {----- WCZYTAJ}

procedure WYPISZ(dane_osoby : dane_personalne);

begin

```
    writeln( ' WYSWIETLANIE DANYCH PERSONALNYCH ' );
    writeln;
    writeln( ' Nazwisko: ' , dane_osoby.nazwisko );
    writeln( '      Imie: ' , dane_osoby.imie );
    writeln( '  Ilosc lat: ' , dane_osoby.il_lat );
```

end; {----- WYPISZ}

VAR *{deklaracje zmiennych}*

```
    osoba : dane_personalne;
```

begin

{program glowny}

```
    WCZYTAJ( osoba );
    WYPISZ( osoba );
    writeln( ' Nacisnij ENTER aby zakonczyc program ' );
    readln;
```

end.

```

program przyklad_2;    {operacje na rekordach zawartych w tablicy}

USES crt;

CONST MAX_IL_OSOB = 10;

TYPE                                     {definicje typow}
  dane_personalne = record
    nazwisko : string [30];
    imie : string [15];
    il_lat : integer ;
  end ;

  T_spis_osob = array[ 1 .. MAX_IL_OSOB ] of dane_personalne;

VAR
  spis_osob : T_spis_osob ;
  i, poz_max, max : integer ;

begin
  clrscr;
  writeln( ' OPERACJA WPROWADZANIA DANYCH DO BAZY' );
  for i := 1 to MAX_IL_OSOB do
    with spis_osob[ i ] do
      begin
        writeln( ' ----- [ ' , i , ' ] ' );
        write( ' Nazwisko: ' );          readln( nazwisko );
        write( '   Imie: ' );           readln( imie );
        write( '   Ilosc lat: ' );      readln( il_lat );
      end;
    writeln;

  writeln( ' OPERACJA WYSZUKIWANIA MAKSIMUM' );
  poz_max := 1;
  max := spis_osob[ 1 ].il_lat ;
  for i := 2 to MAX_IL_OSOB do
    if spis_osob[ i ].il_lat > max then
      begin
        poz_max := i ;
        max := spis_osob[ i ].il_lat ;
      end;

  writeln( ' Najstarsza jest osoba nr ', poz_max );
  write( spis_osob[ poz_max ].nazwisko, ' ', spis_osob[ poz_max ].imie );
  readln;
end .

```

```

program przyklad_3;    {wczytywanie i wyswietlanie tablicy rekordów}
USES crt;
CONST MAX_IL_OSOB = 10;
TYPE
    dane_personalne = record
        nazwisko : string [30];
        imie : string [15];
        il_lat : integer ;
    end;

    tab_osob = array [ 1 .. MAX_IL_OSOB ] of dane_personalne;

procedure WCZYTAJ( var dane_osoby : dane_personalne;
                    numer : byte );

begin
    clrscr;
    writeln( 'WCZYTYWANIE DANYCH PERSONALNYCH [ ' , numer , ' ] ' );
    write( ' Nazwisko: ' );
    readln( dane_osoby.nazwisko );
    write( '   Imie: ' );
    readln( dane_osoby.imie );
    write( 'Ilosc lat: ' );
    readln( dane_osoby.il_lat );
end; {----- WCZYTAJ}

procedure WCZYTAJ_SPIS_OSOB( var tab : tab_osob ;
                              var ilosc : byte );

var
    i : byte;

begin
    clrscr;
    writeln( 'OPERACJA WPROWADZANIA DANYCH DO BAZY' );
    writeln;
    write( ' Podaj ile osob chcesz wprowadzic (0..' , MAX_IL_OSOB , ' ) : ' );
    repeat
        readln( ilosc );
    until (ilosc>=0) and (ilosc<=MAX_IL_OSOB) ;

    for i := 1 to ilosc do
        WCZYTAJ( tab[ i ] , i );
end; {-----WCZYTAJ_SPIS_OSOB}

```

{ dalszy ciąg programu przykład_3 }

```
procedure WYPISZ( dane_osoby : dane_personalne; numer : byte );  
begin  
  writeln( ' WYSWIETLANIE DANYCH PERSONALNYCH [ ' , numer , ' ] ' );  
  writeln;  
  writeln( '   Nazwisko: ' , dane_osoby.nazwisko );  
  writeln( '       Imie: ' , dane_osoby.imie );  
  writeln( '   Ilosc lat: ' , dane_osoby.il_lat );  
end; {----- WYPISZ}
```

```
procedure WYSWIETLAJ_SPIS_OSOB(  tab : tab_osob;  
                                ilosc : byte );  
  
var  
  i : byte;  
begin  
  clrscr;  
  if ilosc = 0 then  
    writeln( ' BAZA jest pusta (zawiera 0 elementow) ' )  
  else  
    for i:=1 to ilosc do  
      WYPISZ( tab[ i ] , i );  
end; {-----WYSWIETLAJ_SPIS_OSOB}
```

```
VAR                                {deklaracja zmiennych globalnych}  
  
  spis_osob_1, spis_osob_2 : tab_osob;  
  ilosc_osob_1, ilosc_osob_2 : byte;  
  
begin                                {program glowny}  
  WCZYTAJ_SPIS_OSOB( spis_osob_1, ilosc_osob_1 );  
  WCZYTAJ_SPIS_OSOB( spis_osob_2, ilosc_osob_2 );  
  writeln( ' Nacisnij ENTER aby przejsc do wyswietlania ' );  
  readln;  
  
  WYSWIETLAJ_SPIS_OSOB( spis_osob_1, ilosc_osob_1 );  
  WYSWIETLAJ_SPIS_OSOB( spis_osob_2, ilosc_osob_2 );  
  writeln( ' Nacisnij ENTER aby zakonczyc program ' );  
  readln;  
end.
```


{ interaktywna wersja procedury wyswietlania do programu przyklad_3 }

```
procedure WYSWIETLAJ_SPIS_OSOB_2( var tab : tab_osob ;  
                                     ilosc : byte );  
var  
  i : byte; klawisz : char;  
begin  
  if ilosc = 0 then  
    writeln( ' BAZA jest pusta (zawiera 0 elementow)' )  
  else  
    begin  
      i:=1;  
      repeat  
        clrscr;  
        WYPISZ( tab[ i ] , i );  
        writeln;  
        writeln( ' P <- pop. osoba, N <- nast. osoba, K <- koniec ' );  
        klawisz := upcase( readkey );  
        case klawisz of  
          'P' : if i>1 then  
            i:=i-1  
          else  
            write(#7);  
          'N' : if i<ilosc then  
            i:=i+1  
          else  
            write(#7);  
          'K' : ;  
          else write(#7);  
        end; {case klawisz of}  
      until klawisz='K';  
    end; {else}  
  end; {-----WYSWIETLAJ_SPIS_OSOB_2}
```

```

program przyklad_4;      { dopisywanie danych do tablicy rekordów }

USES crt;

CONST
    MAX_IL_OSOB = 10;

TYPE
    dane_personalne = record
        nazwisko : string[30];
        imie : string[15];
        il_lat : integer;
    end;

    tab_osob = array[ 1 .. MAX_IL_OSOB ] of dane_personalne;

VAR
    spis_osob : tab_osob;
    ilosc_osob : byte;

procedure WCZYTAJ( var dane_osoby : dane_personalne;
                    numer : byte );

begin
    . . .      { tekst procedury identyczny jak w programie przyklad_3 }
end;

procedure DOPISZ_OSOBY( var tab : tab_osob; var ilosc : byte );
var
    i, ilosc_dopisywanych : byte;
begin
    clrscr;
    writeln( ' OPERACJA DOPISYWANIA DANYCH DO BAZY' );
    writeln;

    write( 'Podaj ile osob chcesz dopisac ( 0..', MAX_IL_OSOB-ilosc , ' ): ' );
    repeat
        readln( ilosc_dopisywanych );
    until (ilosc_dopisywanych>=0)
        and (ilosc_dopisywanych<=MAX_IL_OSOB-ilosc);

    for i := ilosc+1 to ilosc+ilosc_dopisywanych do
        WCZYTAJ( tab[ i ] , i );
    ilosc := ilosc + ilosc_dopisywanych;
end; {-----DOPISZ_OSOBY}

```