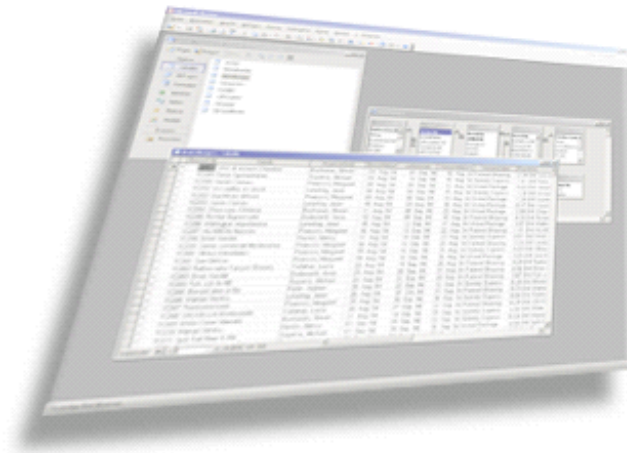


Entscheidungen I – IF-Block





Aussagen=Bedingungen

Eine Aussage ist dadurch charakterisiert, dass man bestimmen kann, ob sie entweder falsch(false) oder wahr(true) ist!

Beispiele für Aussagen:

Heute ist es 25°C

Peter ist müde

$a=3$ (Der Wert von a muss bekannt sein)

$a < b$ (Der Wert von a und b muss bekannt sein)

Keine Aussagen:

Gehen wir heute ins Schwimmbad?

a

$b+4$



Relationale Operatoren I

== gleich(Delphi, VB: =)
 != verschieden(Delphi, VB: <>)
 >, < größer, kleiner
 >=, <= „größer oder gleich“, „kleiner oder gleich“

Hinweise:

„< = “ ist nicht „kleiner oder gleich“; beachte das Leerzeichen zwischen „=“ und dem „<“-Zeichen

„=> “ ist nicht „größer oder gleich“, da hier die Reihenfolge der Zeichen nicht stimmt!

= = sind zwei Gleichheitszeichen hintereinander(!), „=“ akzeptiert der Compiler in manchen P.sprachen, führt aber zu einem logischen Fehler!



Relationale Operatoren II

Term	Wahrheitswert
1 <= 1	true
"car" < "cat"	true
1 < 1	false
"Dog" != "dog"	true
"1" == "01"	false
"XY" != "XYZ"	true
1 == 01	true
"XX" != "xx"	true



Vergleiche

Allgemeines:

Vergleiche werden immer in eine Klammer gesetzt:
if (i==0) {.....}

Vergleichsoperatoren:

=, !=, <=, <, >, >=

Logische Operatoren:

&&(Und), ||(Oder)

Beispiel: if ((i!=0) && (j==2)) {.....}



Pseudocode I

Wenn die Punktzahl von dem Student mindestens 60 ist, dann
ist der Test bestanden

Umsetzung in den Quellcode:

```
if ( studentPunktzahl >= 60 )  
    System.out.println("Bestanden" );
```



Pseudocode II

Wenn die Punktzahl von dem Student mindestens 60 ist, dann
 ist der Test bestanden
sonst
 ist der Test nicht bestanden

Umsetzung in den Quellcode:

```
int studentPunktzahl=30;  
if ( studentPunktzahl >= 60 )  
    System.out.println("Bestanden" );  
else  
    System.out.println("Nicht bestanden" );
```



Verschachtelte Bedingungen – Var I

```
if ( studentPunktzahl >= 90 )  
    System.out.println("1" );  
  
else  
    if ( studentPunktzahl >= 80 )  
        System.out.println("2" );  
    else  
        if ( studentPunktzahl >= 70 )  
            System.out.println("3" );  
        else  
            System.out.println("Nicht bestanden" );
```




Verschachtelte Bedingungen – Var II

```
int studentPunktzahl =60;
if ( studentPunktzahl >= 90 )
    System.out.println("1" );
else if ( studentPunktzahl >= 80 )
    System.out.println("2" );
else if ( studentPunktzahl >= 70 )
    System.out.println("3" );
else
    System.out.println("Nicht bestanden");
```

Blöcke

Lösung:

```
if ( Punktezahl >=50 ) {  
    System.out.println("Bestanden!" );  
}  
else  
{  
    System.out.println("Nicht bestanden!" );  
    System.out.println("Auf ein neues nächstes Jahr!" );  
}
```

„Block“

Alternative:

```
if ( Punktezahl >=50 )  
    System.out.println("Bestanden!" );  
else{  
    System.out.println("Nicht bestanden!" );  
    System.out.println("Auf ein neues nächstes Jahr!" );  
}
```

Dangling-else-Problem I

```
if ( x > 5 )
    if ( y > 5 )
        System.out.println("x und y sind größer 5!" );
else
    System.out.println( "x ist <=5!" );
```

Vorsicht! Ausgeführt wird der Quellcode wie folgt:

```
if ( x >= 5 )
    if ( y > 5 )
        System.out.println("x und y sind >5!" );
else
    System.out.println("x ist <=5!" );
```

d.h. ohne {...} wird das „else“ immer dem letzten „if“ zugeordnet!

Dangling-else-Problem II

Lösung:

„Block“

```
if ( x > 5 )  
  {  
    if ( y > 5 )  
      System.out.println( "x und y sind >5!" );  
    }  
  else  
    System.out.println( "x ist <=5!" );
```



Mehrere Bedingungen

```
if (a > 2 && b < 3) {  
    System.out.println( "a größer als 2 und b kleiner als 3!" );  
    .....  
}
```

```
if (a > 2 || b < 3) {  
    System.out.println( "a größer als 2 oder b kleiner als 3!" );  
    .....  
}
```



Logische Operatoren I

&& (und), || (oder), !(nicht)

x	y	x && y
false	false	false
false	true	false
true	false	false
true	true	true

x	y	x y
false	false	false
false	true	true
true	false	true
true	true	true

x	!(x)
false	true
true	false



Die Und-Verknüpfung: &&

a	b	(a>2)	(b>4)	((a>2) &&(b>4))
3	5	TRUE	TRUE	TRUE
3	0	TRUE	FALSE	FALSE
0	5	FALSE	TRUE	FALSE
0	0	FALSE	FALSE	FALSE



Die Oder-Verknüpfung: ||

a	b	(a>2)	(b>4)	((a>2) (b>4))
3	5	TRUE	TRUE	TRUE
3	0	TRUE	FALSE	TRUE
0	5	FALSE	TRUE	TRUE
0	0	FALSE	FALSE	FALSE

Logische Operatoren II

Die Variablen haben die Werte: $n=4$; $answ="Y"$

$(2 < n \ \&\& \ n < 6)$ _____

$(2 < n \ || \ n = 6)$ _____

$(!(n = 6))$ _____

$(answ = "Y" \ || \ answ = "y")$ _____

$(answ = "Y" \ \&\& \ answ = "y")$ _____

$(!(answ = "Y"))$ _____

$(2 < n \ \&\& \ n = 5 + 1 \ || \ answ = "NO")$ _____

$(n = 2 \ \&\& \ n = 7 \ || \ answ = "Y")$ _____

$(n = 2 \ \&\& \ (n = 7 \ || \ answ = "Y"))$ _____



De Morgansche Gesetze I

Gesetze:

$$(1.) \overline{A \vee B} = \overline{A} \wedge \overline{B}$$

$$(2.) \overline{A \wedge B} = \overline{A} \vee \overline{B}$$

 \overline{A}

Nicht A

 $A \vee B$

A oder B

 $A \wedge B$

A und B

Beispiele:

$$!(c > 3 \ \&\& \ d \neq 4) = (c \leq 3 \ \|\ d = 4)$$

$$!(c > 3 \ \|\ d \neq 4) = (c \leq 3 \ \&\& \ d = 4)$$



De Morgansche Gesetze II

```
if (!(a > 2 && b < 3)) {  
    System.out.println("????????????????");  
    .....  
}
```

```
if (a <= 2 || b >= 3) {  
    System.out.println("????????????????");  
    .....  
}
```



Operatoren

Op	Bedeutung
*,/,+,-,%	Operatoren
&&, 	Logische Operatoren
<, >, ==, !=	Relationaler Vergleich
=	Zuweisung
+=, -=, *=, /=	Zuweisung
++, --	Increment, Decrement

Beispiele:

a+=2	entspricht	a=a+2
a/=2	entspricht	a=a/2
a++	entspricht	a=a+1
++a	entspricht	a=a+1
a--	entspricht	a=a-1



Operatoren II

```
public class Aufgabe {  
  
    public static void main(String[] args) {  
        double a=3;  
        System.out.println("a:" + ++a);  
    }  
}
```

a wird
(1.) inkrementiert
und dann
(2.) ausgegeben

```
C:\WINDOWS\system32\cmd...  
a:4.0  
C:\Documents and Settings\MrBi  
Press any key to continue . .
```



Operatoren III

```
public class Aufgabe {  
  
    public static void main(String[] args) {  
        double a=3;  
        System.out.println("a:" + a++);  
    }  
}
```

a wird
(1.) ausgegeben und dann
(2.) inkrementiert

```
C:\WINDOWS\system32\cmd.exe  
a:3.0  
C:\Documents and Settings\MrBig\works  
Press any key to continue . . .
```