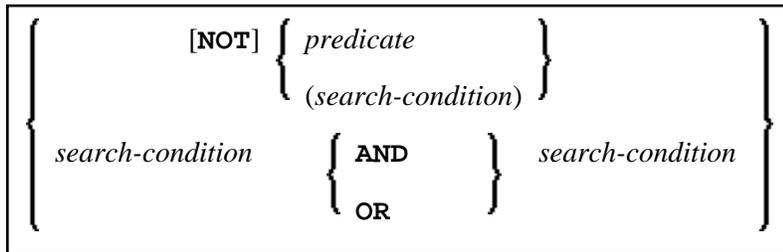


# Suchbedingungen



Dieses Kapitel behandelt folgende Themen:

- search-condition
- predicate

## *search-condition*

Eine *search-condition* kann aus einer einfachen Bedingung (*predicate*) bestehen oder aus mehreren *search-conditions*, die durch die Boole'schen Operatoren AND, OR oder NOT verknüpft werden, wobei die Reihenfolge der Auswertung außerdem durch entsprechende Klammerung bestimmt werden kann.

### Beispiel:

```

DEFINE DATA LOCAL
01 NAME      (A20)
01 AGE       (I2)
END-DEFINE
...
SELECT *
  INTO NAME, AGE
  FROM SQL-PERSONNEL
  WHERE AGE = 32 AND NAME > 'K'
END-SELECT
...

```

## *predicate*

<i>scalar-expression comparison</i>	{ <i>scalar-expression</i> <i>subquery</i> }	
<i>scalar-expression</i> [NOT] BETWEEN <i>scalar-expression</i> AND <i>scalar-expression</i>		
<i>column-reference</i> [NOT] LIKE <i>atom</i>		
<i>column-reference</i> IS [NOT] NULL		
<i>scale-expression</i> [NOT] IN	{ <i>subquery</i> ( <i>atom</i> , ... ) }	
<i>scalar-expression comparison</i>	{ ALL ANY SOME }	<i>subquery</i>
EXISTS <i>subquery</i>		

Das *predicate* gibt eine Bedingung an, die wahr, falsch oder unbekannt sein kann.

In einer *search-condition* kann ein *predicate* aus einer einfachen oder komplexen Vergleichsoperation oder einer anderen Art von Bedingung bestehen.

### Beispiel:

```
SELECT NAME, AGE
INTO VIEW PERS
FROM SQL-PERSONNEL
WHERE AGE BETWEEN 20 AND 30
      OR AGE IN ( 32, 34, 36 )
      AND NAME LIKE '%er'
...
```

### Anmerkung:

Das Prozentzeichen (%) kann zu Konflikten mit Natural-Terminalkommandos führen. In diesem Fall definieren Sie als Steuerzeichen für Terminalkommandos ein anderes Zeichen als % (siehe Session-Parameter CF).

Die einzelnen *predicates* sind auf den folgenden Seiten beschrieben (weitere Informationen zu *predicates* finden Sie in der betreffenden Literatur). Entsprechend der obigen Syntax heißen sie wie folgt:

- Comparison Predicate
- BETWEEN Predicate
- LIKE Predicate
- NULL Predicate
- IN Predicate
- Quantified Predicate

- EXISTS Predicate

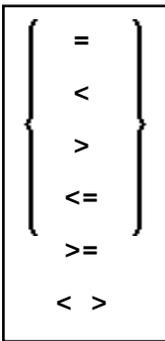
## Comparison Predicate



Ein Comparison Predicate vergleicht zwei Werte.

Siehe Informationen zu *scalar-expression*.

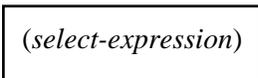
### *comparison*



*comparison* kann einer der folgenden Operatoren sein:

=	gleich
<	kleiner als
>	größer als
<=	kleiner gleich
>=	größer gleich
<>	ungleich

### *subquery*



Eine *subquery* ist eine *select-expression* innerhalb einer anderen *select-expression*.

### Beispiel:

```

DEFINE DATA LOCAL
1 #NAME      (A20)
1 #PERSNR    (I4)
END-DEFINE
...
SELECT NAME, PERSNR
      INTO #NAME, #PERSNR
    
```

```

FROM SQL-PERSONNEL
WHERE PERSNR IN
  ( SELECT PERSNR
    FROM SQL-AUTOMOBILES
    WHERE COLOR = 'black' )
  ...
END-SELECT

```

Siehe *Select Expressions*.

## BETWEEN Predicate

*scalar-expression* [NOT] **BETWEEN** *scalar-expression* **AND** *scalar-expression*

Ein BETWEEN Predicate vergleicht einen Wert mit einem Bereich von Werten.

Siehe *scalar-expression*.

## LIKE Predicate

*column-reference* [NOT] **LIKE** *atom*

Ein LIKE Predicate sucht nach Zeichenketten, die ein bestimmtes Muster haben.

Siehe *column-reference* und *atom*.

## NULL Predicate

*column-reference* **IS** [NOT] **NULL**

Ein NULL Predicate prüft auf Nullwerte.

Siehe *column-reference*.

## IN Predicate

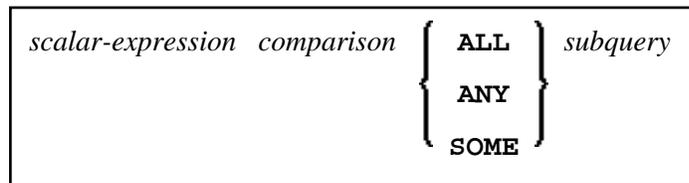
*scalar-expression* [NOT] **IN** { *subquery* ... }  
(*atom*)

Ein IN Predicate vergleicht einen Wert mit einer Sammlung von Werten.

Siehe *scalar-expression* und *atom*.

Siehe *subquery*.

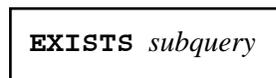
## Quantified Predicate



Ein Quantified Predicate vergleicht einen Wert mit einer Sammlung von Werten.

Siehe *scalar-expression*, *comparison* und *subquery*.

## EXISTS Predicate



Ein EXISTS Predicate prüft, ob bestimmte Reihen vorhanden sind.

Die Bedingung des EXISTS Predicate kann nur erfüllt werden, wenn die ausgewertete *subquery* tatsächlich ein Ergebnis liefert, d.h. wenn mindestens eine Reihe in der FROM-Tabelle der *subquery* die WHERE-Bedingung dieser *subquery* erfüllt.

Beispiel für EXISTS:

```

DEFINE DATA LOCAL
1 #NAME      (A20)
END-DEFINE
...
SELECT NAME
  INTO #NAME
  FROM SQL-PERSONNEL
  WHERE EXISTS
    ( SELECT *
      FROM SQL-EMPLOYEES
      WHERE PERSNR > 1000
        AND NAME < 'L' )
    ...
END-SELECT
...

```

Siehe *subquery*.