Konspekt

Podstawowe działanie bazy danych - przykłady, słowa kluczowe, wstęp do SQL

OPERATORY

### **Operatory logiczne: AND, OR, NOT**

Operatory logiczne AND, OR oraz NOT są podstawowymi narzędziami w logice i programowaniu, które pozwalają tworzyć złożone warunki i podejmować decyzje na podstawie różnych kryteriów. Działają one na zasadzie prawdziwości (ang. truth values), gdzie każdy warunek może być prawdziwy (true) lub fałszywy (false).

#### **1. Operator AND**

* Operator AND zwraca wynik **prawdziwy (true)** tylko wtedy, gdy oba warunki są prawdziwe.
* Jeśli choć jeden z warunków jest fałszywy, wynik będzie fałszywy.

**Przykład**:

* "Jestem w domu **i** pada deszcz" – zdanie jest prawdziwe tylko wtedy, gdy oba fakty są prawdziwe jednocześnie.

**Tabela prawdy dla AND**:

|  |  |  |
| --- | --- | --- |
| **Warunek 1** | **Warunek 2** | **Wynik (AND)** |
| true | true | true |
| true | false | false |
| false | true | false |
| false | false | false |

#### **2. Operator OR**

* Operator OR zwraca wynik **prawdziwy (true)**, jeśli przynajmniej jeden z warunków jest prawdziwy.
* Wynik będzie fałszywy tylko wtedy, gdy oba warunki są fałszywe.

**Przykład**:

* "Idę na spacer **lub** oglądam telewizję" – zdanie jest prawdziwe, jeśli spełniony jest przynajmniej jeden z tych warunków.

**Tabela prawdy dla OR**:

|  |  |  |
| --- | --- | --- |
| **Warunek 1** | **Warunek 2** | **Wynik (OR)** |
| true | true | true |
| true | false | true |
| false | true | true |
| false | false | false |

#### **3. Operator NOT**

* Operator NOT zmienia wartość logiczną na przeciwną.
* Jeśli warunek jest prawdziwy (true), wynik NOT będzie fałszywy (false), i odwrotnie.

**Przykład**:

* "Nie pada deszcz" – zdanie będzie prawdziwe, jeśli faktycznie nie pada, i fałszywe, jeśli deszcz pada.

**Tabela prawdy dla NOT**:

|  |  |
| --- | --- |
| **Warunek** | **Wynik (NOT)** |
| true |  false |
| false |  true |

#### **Kombinacja operatorów**

Operatory można łączyć, aby budować bardziej złożone warunki. Kolejność ich wykonywania (priorytety) można kontrolować za pomocą nawiasów ().

**Przykład**:

* "Jestem w domu **i** (nie pada deszcz **lub** jest ciepło)" – zdanie jest prawdziwe, jeśli jestem w domu i spełniony jest jeden z warunków w nawiasie.

Te operatory są uniwersalne i używane w matematyce, logice, programowaniu, a także w codziennym życiu do wyrażania złożonych zależności logicznych.



### **AND**

**Wstęp:**

W tym module nauczysz się łączyć więcej niż jeden warunek za pomocą operatora AND. Pozwala to precyzyjniej filtrować dane – oba warunki muszą być spełnione, by rekord został wyświetlony.

**Składnia**:

SELECT kolumna
FROM nazwa\_tabeli
WHERE warunek1 AND warunek2;

**Przykład**:

Aby wybrać klientów z Niemiec i mających siedzibe w Berlinie:

SELECT \* FROM Customers
WHERE Country='Germany' AND City='Berlin';

**Zadanie**:

Znajdź klientów z Niemiec ale mają ID większe od 15.

### **OR**

**Wstęp:**

W tym module dowiesz się, jak korzystać z operatora OR, który pozwala łączyć warunki, z których wystarczy spełnić jeden, aby rekord został wyświetlony.

**Składnia**:

SELECT kolumna
FROM nazwa\_tabeli
WHERE warunek1 OR warunek2;

**Przykład**:

Aby wybrać klientów z Berlina lub z miasta Munchen:

SELECT \* FROM Customers
WHERE City='Berlin' OR City='München';

**Zadanie**:

Znajdź klientów z tabeli customers którzy są z Madrytu lub Berlina.

### **NOT**

**Wstęp:**

Tutaj nauczysz się, jak używać operatora NOT, który neguje warunek – zwraca rekordy, które NIE spełniają danego warunku.

**Składnia**:

SELECT kolumna
FROM nazwa\_tabeli
WHERE NOT warunek;

**Przykład**:

Aby wybrać klientów którzy nie są z Niemiec:

SELECT \* FROM Customers
WHERE NOT Country='Germany';

**Zadanie**:

Wyświetl zamówienia z tabeli orders gdzie ShipperID nie jest równe 1.

Zachęcam do obejrzenie filmiku w którym tłumaczę zastowanie operatorów. Operatory są bardzo ważne nie tylko w SQL ale także ogólnie w programowaniu.