



Datenbanken und SQL

(Woche 2 - Tag 4)

Agenda

Boolesche Algebra (Aussagenlogik)

- Definition
- Motivation
- Regeln
 - AND
 - OR
 - NOT
 - XOR
 - Gründe, warum XOR selten gebraucht wird
 - Nachweis der Redundanz von XOR
- Stolperfallen
 - Unterschiedliche Verwendung von „UND“ / „ODER“ in Aufgabenstellung und Lösung
 - „Regel versus Ausnahme“
 - Allgemeine Empfehlung (siehe auch: „weder noch“)
 - Ein „etwas unseriöser“ Tipp 😊
- Motivation (Nachtrag)

Boolesche Algebra

Definition

- Die **Boolesche Algebra** ist eine formalisierte Betrachtung von Sachverhalten, denen wir bei der Mengenlehre und Aussagenlogik begegnen werden.
- Wir werden uns im Laufe dieses Kurses mit beiden Themenfeldern auseinandersetzen, betrachten heute aber zunächst ausschließlich die **Aussagenlogik**.
- Bei dieser geht es um die Frage, welchen **Wahrheitswert** („wahr oder „falsch“) eine aus mehreren Einzel-Aussagen verknüpfte „Gesamt-Aussage“ besitzt.
- **Aussagen werden üblicherweise mit großen Buchstaben abgekürzt. So können wir beispielsweise die Aussagen ...**
 - „ $1+1=2$ “
 - „Der Eiffelturm steht in Rom“... mit dem Buchstaben **A** und **B** abkürzen.
- Gefragt wird dann (z.B.) nach dem Wahrheitswert (wahr oder falsch) von **A** oder **B** = ?

Motivation

- Die Motivation für dieses Themenfeld ergibt sich aus unserer Beschäftigung mit **WHERE-Klauseln**. Hier werden zum Teil auch verknüpfte Bedingungen benötigt.
- Die Verwendung verknüpfter Bedingungen ist oft „komfortabel“. Am Ende dieser Vorlesung werden wir aber mit Hilfe von Beispielen auch nachweisen können, dass die Inhalte der Aussagenlogik für uns gelegentlich sogar **unverzichtbar** sind.

AND

AND (Und-Verknüpfung)

Die Und-Verknüpfung kann – anschaulich gesprochen – als „**streng**“ bezeichnet werden. Gemeint ist: Die Verknüpfung ist **nur wahr**, wenn **beide Teilaussagen** („A“ / „B“) **wahr** sind.

	A ist wahr	A ist falsch
B ist wahr		
B ist falsch		

AND (Und-Verknüpfung)

Die Und-Verknüpfung kann – anschaulich gesprochen – als „**streng**“ bezeichnet werden. Gemeint ist: Die Verknüpfung ist **nur wahr**, wenn **beide Teilaussagen wahr** sind.

	A ist wahr	A ist falsch
B ist wahr	=> (A und B) ist wahr	
B ist falsch		

AND (Und-Verknüpfung)

Die Und-Verknüpfung kann – anschaulich gesprochen – als „**streng**“ bezeichnet werden. Gemeint ist: Die Verknüpfung ist **nur wahr**, wenn **beide Teilaussagen wahr** sind.

	A ist wahr	A ist falsch
B ist wahr	=> (A und B) ist wahr	=> (A und B) ist falsch
B ist falsch		

AND (Und-Verknüpfung)

Die Und-Verknüpfung kann – anschaulich gesprochen – als „**streng**“ bezeichnet werden. Gemeint ist: Die Verknüpfung ist **nur wahr**, wenn **beide Teilaussagen wahr** sind.

	A ist wahr	A ist falsch
B ist wahr	=> (A und B) ist wahr	=> (A und B) ist falsch
B ist falsch	=> (A und B) ist falsch	

AND (Und-Verknüpfung)

Die Und-Verknüpfung kann – anschaulich gesprochen – als „**streng**“ bezeichnet werden. Gemeint ist: Die Verknüpfung ist **nur wahr**, wenn **beide Teilaussagen wahr** sind.

	A ist wahr	A ist falsch
B ist wahr	=> (A und B) ist wahr	=> (A und B) ist falsch
B ist falsch	=> (A und B) ist falsch	=> (A und B) ist falsch

OR

OR (Oder-Verknüpfung)

Die Oder-Verknüpfung kann – anschaulich gesprochen – als „**bescheiden**“ bezeichnet werden. Gemeint ist: Die Verknüpfung ist **nur falsch**, wenn **beide Teilaussagen** („A“ / „B“) **falsch** sind.

	A ist wahr	A ist falsch
B ist wahr		
B ist falsch		

OR (Oder-Verknüpfung)

Die Oder-Verknüpfung kann – anschaulich gesprochen – als „**bescheiden**“ bezeichnet werden. Gemeint ist: Die Verknüpfung ist **nur falsch**, wenn **beide Teilaussagen** („A“ / „B“) **falsch** sind.

	A ist wahr	A ist falsch
B ist wahr	=> (A oder B) ist wahr	
B ist falsch		

OR (Oder-Verknüpfung)

Die Oder-Verknüpfung kann – anschaulich gesprochen – als „**bescheiden**“ bezeichnet werden. Gemeint ist: Die Verknüpfung ist **nur falsch**, wenn **beide Teilaussagen** („A“ / „B“) **falsch** sind.

	A ist wahr	A ist falsch
B ist wahr	=> (A oder B) ist wahr	=> (A oder B) ist wahr
B ist falsch		

OR (Oder-Verknüpfung)

Die Oder-Verknüpfung kann – anschaulich gesprochen – als „**bescheiden**“ bezeichnet werden. Gemeint ist: Die Verknüpfung ist **nur falsch**, wenn **beide Teilaussagen** („A“ / „B“) **falsch** sind.

	A ist wahr	A ist falsch
B ist wahr	=> (A oder B) ist wahr	=> (A oder B) ist wahr
B ist falsch	=> (A oder B) ist wahr	

OR (Oder-Verknüpfung)

Die Oder-Verknüpfung kann – anschaulich gesprochen – als „**bescheiden**“ bezeichnet werden. Gemeint ist: Die Verknüpfung ist **nur falsch**, wenn **beide Teilaussagen** („A“ / „B“) **falsch** sind.

	A ist wahr	A ist falsch
B ist wahr	=> (A oder B) ist wahr	=> (A oder B) ist wahr
B ist falsch	=> (A oder B) ist wahr	=> (A oder B) ist falsch

NOT

NOT (NICHT-Operator)

Der Nicht-Operator ist keine Verknüpfung zwischen zwei Aussagen, sondern bezieht sich auf eine einzige Aussage, die negiert wird. (Der Wahrheitswert wird quasi „ausgetauscht“)

	A ist wahr	A ist falsch
NOT A		

NOT (NICHT-Operator)

Der Nicht-Operator ist keine Verknüpfung zwischen zwei Aussagen, sondern bezieht sich auf eine einzige Aussage, die negiert wird. (Der Wahrheitswert wird quasi „ausgetauscht“)

	A ist wahr	A ist falsch
NOT A	falsch	

NOT (NICHT-Operator)

Der Nicht-Operator ist keine Verknüpfung zwischen zwei Aussagen, sondern bezieht sich auf eine einzige Aussage, die negiert wird. (Der Wahrheitswert wird quasi „ausgetauscht“)

	A ist wahr	A ist falsch
NOT A	falsch	wahr

Erläuterung:

Das mag auf den ersten Blick erstaunlich wirken, entspricht aber exakt unserer „Alltags-Logik“:
Wer feststellt, dass eine falsche Aussage „NICHT gilt“, der sagt etwas wahres.

XOR

XOR (Entweder-Oder-Verknüpfung)

Die XOR-Verknüpfung ist **nur dann wahr**, wenn **genau 1 Teilaussage wahr** ist.

	A ist wahr	A ist falsch
B ist wahr		
B ist falsch		

XOR (Entweder-Oder-Verknüpfung)

Die XOR-Verknüpfung ist **nur dann wahr**, wenn **genau 1 Teilaussage wahr** ist.

	A ist wahr	A ist falsch
B ist wahr	=> (A XOR B) ist falsch	
B ist falsch		

XOR (Entweder-Oder-Verknüpfung)

Die XOR-Verknüpfung ist **nur dann wahr**, wenn **genau 1 Teilaussage wahr** ist.

	A ist wahr	A ist falsch
B ist wahr	=> (A XOR B) ist falsch	=> (A XOR B) ist wahr
B ist falsch		

XOR (Entweder-Oder-Verknüpfung)

Die XOR-Verknüpfung ist **nur dann wahr**, wenn **genau 1 Teilaussage wahr** ist.

	A ist wahr	A ist falsch
B ist wahr	=> (A XOR B) ist falsch	=> (A XOR B) ist wahr
B ist falsch	=> (A XOR B) ist wahr	

XOR (Entweder-Oder-Verknüpfung)

Die XOR-Verknüpfung ist **nur dann wahr**, wenn **genau 1 Teilaussage wahr** ist.

	A ist wahr	A ist falsch
B ist wahr	=> (A XOR B) ist falsch	=> (A XOR B) ist wahr
B ist falsch	=> (A XOR B) ist wahr	=> (A XOR B) ist falsch

Gründe, warum XOR **selten gebraucht** wird

- Wir fordern oft Bedingungen, die entweder **beide gemeinsam** erfüllt sein müssen, oder bei denen es uns reicht, wenn **mindestens 1 Bedingung** zutrifft, Beispiele:
 - Man besteht die Fahrprüfung, wenn man die schriftliche UND die praktische Prüfung bestanden hat.
 - Ein Fußballtrainer gewinnt einen Titel, wenn er Deutscher Meister wird ODER den Pokal gewinnt.
- Eher ungewöhnlich wäre es hingegen, wenn wir behaupten, dass ein Trainer keinen Titel verdient hat, „weil“ er Deutscher Meister **UND** Pokalsieger wurde!?
- Ein anderer Grund für den seltenen Bedarf der XOR-Verknüpfung ist, dass wir nicht selten Bedingungen betrachten, die sich **gegenseitig ausschließen**. In diesem Fall ist dann aber die Verwendung von OR und XOR ohne Unterschied.

Nachweis der **Redundanz** von XOR

- Wie können die Verknüpfung **A XOR B** gleichwertig durch **(A AND NOT B) OR (NOT A AND B)** ersetzen. Aus dieser Sicht ist also **XOR** aus AND, OR und NOT herleitbar und daher **redundant**.
- Dennoch ist es natürlich deutlich **übersichtlicher**, mit XOR zu arbeiten.
- Wir werden bei den heutigen Übungsaufgaben allerdings Problemstellungen kennen lernen, die deutlich machen, dass die alternative Schreibweise gelegentlich **unverzichtbar** ist.

Stolperfallen

Unterschiedliche Verwendung von **UND** und **ODER** in Aufgabenstellung und Lösung

Beispiel: Der Kunde mit ID=1 **und** der Kunde mit ID=3 sollen gelöscht werden:

DELETE FROM Kunde WHERE Kunde_ID = 1 **OR Kunde_ID = 3;**

Erläuterung:

In der Aufgabenstellung werden mit „und“ nicht Bedingungen, sondern **Aufgaben** verknüpft:
Der eine Kunde soll **gelöscht** werden UND der andere auch.

In der Lösung werden hingegen zwei Bedingungen verknüpft:

Wenn wir einen Datensatz finden, der die Bedingung erfüllt, dass die Kunde_ID=1 ist
ODER

wenn wir einen Datensatz finden, der die Bedingung erfüllt, dass die Kunde_ID=5 ist, ...

... so ist die „Gesamt-Bedingung“ erfüllt und der entsprechende Datensatz wird gelöscht.

Regel versus Ausnahme

- Gelegentlich werden in einer Aufgabenstellung nicht die Bedingungen (bzw. „Regeln“) genannt, die erfüllt sein müssen, sondern gerade umgekehrt die Ausnahmen, die eben **NICHT** erfüllt sein dürfen.
- **Beispiel:** Es sollen fast alle Abrechnungen gelöscht werden, **Ausnahme** sind jedoch alle Abrechnungen, die von Kunden(3) stammen, oder solche Abrechnungen, die nach dem 01.01.2000 eingereicht wurden.
- Die Lösung besteht dann darin, zunächst die Ausnahme zu formulieren, diese in Klammern zu setzen und den gesamten Klammersausdruck zu negieren:

DELETE FROM Abrechnung **WHERE NOT**(Kunde_ID=3 **OR** Datum>"2000-01-01");

Zusatzbemerkung:

Die obige Bemerkung gilt im Prinzip auch für die Formulierung „**weder noch**“, die man aber der Einfachheit halber in „**Das eine NICHT UND das andere auch NICHT**“ übersetzt.

Beispiel: Löschen Sie bitte alle Kunden, die weder mit Vornamen „Peter“ noch mit Nachnamen „Müller“ heißen:

DELETE FROM Kunde **WHERE NOT** Vorname="Peter" **AND NOT** Nachname="Müller";

Ein „etwas unseriöser“ Tipp 😊

- Oft steht man lediglich vor der Frage, ob die Verknüpfung „AND“ oder „OR“ lauten soll. Falls man diesbezüglich (z.B. im Prüfungsstress) unsicher ist, so hilft es gelegentlich, sich zu überlegen, was das Gegenteil von der vermuteten Lösung bedeuten würde.
- **Beispiel:** Es sollen alle Produkte gelöscht werden, die billiger als 10 Euro oder teurer als 100 Euro sind. Man vermutet spontan, die folgende Lösung, ist sich aber nicht sicher:

DELETE FROM Produkt **WHERE** Euro_Preis < 10 **OR** Euro_Preis > 100;

- Was aber würde es bedeuten, wenn man sich für die folgende Lösung entschieden hätte?

DELETE FROM Produkt **WHERE** Euro_Preis < 10 **AND** Euro_Preis > 100;

Hinweis:

Falls Sie jemals ein Produkt finden sollten, dass billiger als 10 Euro **UND** teurer als 100 Euro, ist ...
... dann schlafen Sie sich am Besten erst einmal richtig aus ;-)

Motivation (Nachtrag)

- Verknüpfte Aussagen sind nicht immer notwendig, sondern manchmal nur **komfortabel**:

Anstelle von ...

DELETE FROM Kunde WHERE Kunde_ID=2 OR Kunde_ID=3;

... könnte man auch die beiden folgenden Befehle absenden:

DELETE FROM Kunde WHERE Kunde_ID=2;

DELETE FROM Kunde WHERE Kunde_ID=3;

- Gelegentlich ist aber die Verknüpfung auch **unverzichtbar**:

Mit dem folgenden Befehl ...

DELETE FROM Produkt WHERE Euro_Preis >= 10 AND Euro_Preis <= 100;


... werden alle Produkte gelöscht, deren Preis zwischen 10 und 100 Euro liegen.

Die beiden folgenden Befehle löschen hingegen **die gesamte Tabelle** Produkt(!):

DELETE FROM Produkt WHERE Euro_Preis >= 10 ;

DELETE FROM Produkt WHERE Euro_Preis <= 100;

Gemeinsame Übung („Live-Coding“) -> A_02_04_01



Aufgabe_02_04_01

Ausschnitt eines Datenbankschemas:

Kuchen
id
hersteller_id
kalorien
gewicht_gramm
preis_euro
volumen_cm3

Formulieren Sie bitte für alle folgenden Lösch-Aufträge eine entsprechende WHERE-Klausel:


DELETE FROM Kuchen WHERE(...)

- a) Alle Kuchen mit einer id größer 3 und kleiner 9.
- b) Alle Kuchen von Hersteller (3), die teurer als 20,- € sind.
- c) Alle Kuchen mit mehr als 900 Kalorien, falls diese weniger als 300 Gramm wiegen.
- d) Nur Kuchen, die weder 1 kg wiegen, noch 1000 Kalorien haben.
- e) Fast alle Kuchen, Ausnahmen sind allerdings Kuchen von Hersteller 3 und solche, deren Volumen unter 700 cm³ liegen.
- f) Alle Kuchen, die entweder eine ID kleiner 12 haben, oder einen Preis größer 30 €.
- g) Alle Kuchen, die von Hersteller 1 stammen, ebenfalls alle Kuchen, die genau 1 Kalorie haben, und ferner alle Kuchen, die genau 1 Gramm wiegen.

WBS TRAINING AG
Lorenzweg 5
D-12099 Berlin
Amtsgericht Berlin HRB 68531
Sitz der Gesellschaft: Berlin

Vorstand:
Hennrich Kronbichler,
Joachim Giese
Aufsichtsrat (Vorsitz): Dr. Daniel Stadler
USt-IDNr.: DE 209 768 248

GLS Gemeinschaftsbank eG
IBAN: DE18 4306 0967 1146 1814 00
BIC: GENODEM33GLS



GLS zertifiziert nach
ISO 9001:2015 und ISO 14001:2015
Zertifizierung nach AENOR Reg. Nr. 0110006-K0002

Vielen Dank für Ihre Aufmerksamkeit!

