

MOTIVATION

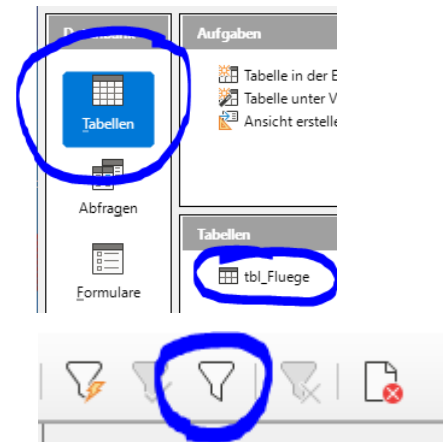
Die Firma "5 Stars" bewertet und vermarktet privat vermietete Ferienwohnung in Deutschland. Dazu reisen Mitarbeiter zu den Wohnungen, erstellen ein Exposee und bewerten die Wohnungen. Für die anfallenden Inlandsflüge hält die Firma eine Tabelle bereit, in der Flüge verzeichnet sind, jeden Tag durchgeführt werden.

Öffne die Datenbank "Flugdaten" und klicke auf „Tabellen“ --> „tbl_Fluege

A) Finde den billigsten Flug von Hannover nach Nürnberg. Benutze den Standardfilter.

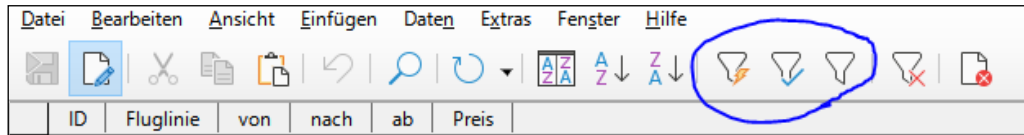
B) Finde den spätesten Flug von Düsseldorf nach Münster mithilfe des Standardfilters.

C) Kann man die Anzahl aller Flüge ermitteln, die von "Quick Lift" von Bremen aus durchgeführt werden.



ÜBERSICHT

In der Tabellenansicht der Daten sind "Autofilter" und "Standardfilter" nützliche Werkzeuge, um in einer Tabelle Daten zu filtern.

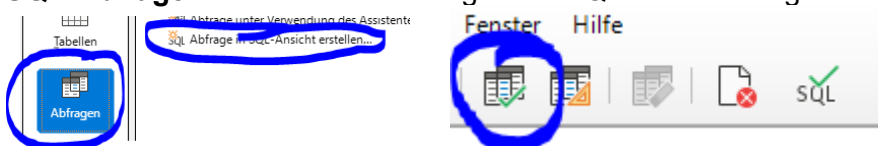


Der **Standardfilter** erlaubt mehrere Suchkriterien über eine **logische Verknüpfung**. Der **Autofilter** sucht alle Datensätze gemäß der aktuell markierten Zelle. Falls die Filter nicht sichtbar sind: Ansicht --> Symbolleiste --> Tabellenansicht.

Der **Bearbeitungsmodus** verschafft einen Überblick über die Datentypen in der Tabelle. Man öffnet eine Tabelle im Bearbeitungsmodus, indem man entweder die Tabelle markiert und dann das Icon "**Bearbeiten**" klickt oder einen Rechtsklick auf das Tabellensymbol in den Entwurfsmodus macht und "Bearbeiten" wählt.

Eine Tabellenzeile nennt sich **Datensatz**, der aus mehreren **Datenfeldern** besteht. Jede Spalte wird durch ein **Attribut** beschrieben (Bsp.: Fluglinie oder von oder nach)

SQL-Abfragen können unter Abfragen --> SQL-Ansicht ausgeführt werden.



Eine **SQL-Anweisung** enthält immer

- 1.) Die **Tabellenspalten**, die später angezeigt werden,
- 2.) die **Tabellennamen**, aus denen ausgewählt wird,
- 3.) die **Abfragekriterien**.

	Datenbanken mit OpenOffice/Libre Office
	Daten in Tabellen

SQL	Beispiel 1	Beispiel 2
SELECT Spalten	SELECT *	SELECT "tbl_Fluege"."Fluglinie"
FROM Tabellen	FROM "tbl_Fluege"	FROM "tbl_Fluege"
WHERE Bedingung	WHERE „von“ = ‚Bremen‘	WHERE "Preis" > '20'
	AND "nach" = 'Berlin'	OR "Preis" < '50'

Beachte, dass bei OpenOffice Base, die **Tabellenangaben** in **Doppelhochkommas** und die **Kriterien** in **Einfachhochkommas** stehen.

ANWENDEN

Aufgabe 1

Wende den **Standardfilter** an und schreibe jeweils die **Verknüpfungskriterien** des Filters auf, um alle entsprechenden Flüge anzeigen zu lassen. Ermittle durch Abzählen:

- a) den billigsten Flug von Hannover nach Nürnberg,
- b) den spätesten Flug von Düsseldorf nach Münster,
- c) die Anzahl aller Flüge, die "Quick Lift" von Bremen aus durchführt,
- d) die Anzahl der Flüge von Berlin aus nach 20:00 Uhr,
- e) die Anzahl aller Flüge aller Städte, die das Nachtflugverbot (23:00-06:00) verletzen (→ werden Uhrzeiten korrekt verglichen?),

HA

- f) die Anzahl der Flüge unter 20 Euro,
- g) die Anzahl der Flüge von Nürnberg aus unter 20 Euro,
- h) Gibt es einen Flug von Leipzig nach Berlin nach 19:00 Uhr?

Aufgabe 2

Erstelle zu den folgenden Fällen die **SQL-Anweisung**, um alle entsprechenden Flüge anzeigen zu lassen. Ermittle dann durch Abzählen. Die Abfragen sind dieselben wie bei Aufgabe 1

- a) den billigsten Flug von Hannover nach Nürnberg,
- b) den spätesten Flug von Düsseldorf nach Münster,
- c) die Anzahl aller Flüge, die "Quick Lift" von Bremen aus durchführt,
- d) die Anzahl der Flüge von Berlin aus nach 20:00 Uhr,
- e) die Anzahl aller Flüge aller Städte, die das Nachtflugverbot (23:00-06:00) verletzen (→ werden Uhrzeiten korrekt verglichen?),

HA

- f) die Anzahl der Flüge unter 20 Euro,
- g) die Anzahl der Flüge von Nürnberg aus unter 20 Euro.
- h) Gibt es einen Flug von Leipzig nach Berlin nach 19:00 Uhr?

Aufgabe 3

Ermittle die folgenden Fragestellungen mithilfe von SQL-Anweisungen und schreibe die SQL-Anweisungen auf. Benutze dabei: ORDER BY *Kriterium* ASC / ORDER BY *Kriterium* DESC (aufsteigend / absteigend). Lies dann das Ergebnis aus der Tabelle ab.

- a) Ermittle die Stadt mit dem frühesten Start,
- b) die Airline mit dem teuersten Flug,
- c) die Airline mit dem billigsten Flug der in Berlin startet,

	Datenbanken mit OpenOffice/Libre Office
	Daten in Tabellen

d) Ermittle den billigsten Flug von Berlin nach München.

HA

- e) Ermittle die Stadt mit dem spätesten Start,
- f) die Airline mit dem billigsten Flug,
- g) die Airline mit dem billigsten Flug der in Bremen startet,
- h) ermittle den billigsten Flug von Dresden nach Bremen

Aufgabe 4

- a) Überprüfe mit einer einzigen SQL-Abfrage, ob es Verbindungen "Berlin - München" mit Hin- und Rückflug an einem Tag gibt.
- b) Überprüfe mit einer einzigen SQL-Abfrage, ob es möglich ist, am selben Tag von Bremen nach Köln/Bonn und weiter nach Dresden zu fliegen.
- c) Überprüfe mit einer einzigen SQL-Abfrage, ob es möglich ist, am selben Tag von Berlin nach München und wieder zurück mit derselben Airline zu fliegen.

HA

- d) Überprüfe mit einer einzigen SQL-Abfrage, ob es möglich ist, am selben Tag von Berlin nach Köln/Bonn und weiter nach Münster zu fliegen.
- e) Überprüfe mit einer einzigen SQL-Abfrage, ob es möglich ist, am selben Tag von Stuttgart nach Bremen und wieder zurück mit derselben Airline zu fliegen.

Aufgabe 5

Öffne "StreamingDaten" und sieh die Daten in der Tabelle an. Die Anzahl der Datensätze kann mit dem Zusatz SELECT COUNT(*) ermittelt werden. Erstelle jeweils eine SQL-Anweisung:

- a) der Anzahl der "Premium"-Abonnenten,
- b) der Anzahl der "Mini"- oder "Test"-Abonnenten,
- c) der Anzahl der Abonnenten mit 50 Jahren oder darüber,
- d) der Anzahl der Abonnenten über 30 Jahren, die ein "Test"-Abo haben,

HA

- e) Anzahl der Abonnenten, die zwischen 30 und 50 sind und ein XL Abo haben,
- f) die Abonnementform mit den meisten Teilnehmern an. Schreibe für jedes Abo eine neue SQL-Abfrage.

Aufgabe 6

AVG, MIN und MAX berechnen den Durchschnitt, das Minimum und das Maximum.

- a) Berechne das Durchschnittsalter eines Premiumabonnenten,
- b) Berechne das geringste Alter aller Basic-Abonnenten,
- c) Finde das Alter des ältesten Pro-Abonnenten.

HA

- d) Berechne die Durchschnittsrestlaufzeit der Test-Abos.