**SGL mit Manoor Tag 3**

**Modul 6: Einführung in SQL Tag 3**

**Themen heute:**

* **UPDATE**: Die UPDATE-Anweisung bearbeitet eine Zeile in einer Tabelle. Sie können die UPDATE-Anweisung verwenden, wenn Sie vorhandene Datensätze ändern möchten.
* **DELETE**: Die DELETE FROM-Anweisung löscht eine oder mehrere Zeilen aus einer Tabelle. Sie können die Anweisung verwenden, wenn Sie vorhandene Datensätze löschen möchten
* **LIKE**: Der LIKE-Operator wird in einer WHERE-Klausel verwendet, um nach einem bestimmten Muster in einer Spalte zu suchen.

|  |  |
| --- | --- |
| LIKE Operator | Beschreibung |
| WHERE Kundenname LIKE ‘a%’ | Findet alle Werte, die mit “a” beginnen |
| WHERE Kundenname LIKE ‚%a‘ | Findet alle Werte, die mit „a“ enden |
| WHERE Kundenname LIKE ‚%or%‘ | Findet alle Werte, die „or“ an beliebiger Position haben. |
| WHERE Kundenname LIKE ‚\_r%‘ | Findet alle Werte, die „r“ an zweiter Position haben. |
| WHERE Kundenname LIKE ‚a\_%‘ | Findet alle Werte, die mit „a“ beginnen und mindestens 2 Zeichen lang sind. |
| WHERE Kundenname LIKE ‚a\_\_%‘ | Findet alle Werte, die mit „a“ beginnen und mindestens 3 Zeichen lang sind. |
| WHERE Kundenname LIKE ‚a&o‘ | Findet alle Werte, die mit “a” beginnen und mit “o” enden. |

* **WILDCARD**: Ein Platzhalterzeichen wird verwendet, um ein oder mehrere Zeichen in einer Zeichenfolge zu ersetzen.

|  |  |  |
| --- | --- | --- |
| Symbol | Beschreibung | Beispiel |
| % | Stellt Null oder mehr Zeichen dar | Bl% findet Bl, Black, Blue, und Blob |
| \_ | Stellt ein einzelnes Zeichen dar | H\_t findet hot, hat, und hit |

**Material:**

Sehen Sie sich die Videos 21-25, 27, 28 der folgenden Playlist an:

[*https://www.youtube.com/playlist?list=PL\_RGaFnxSHWr\_6xTfF2FrIw-NAOo3iWMy*](https://www.youtube.com/playlist?list=PL_RGaFnxSHWr_6xTfF2FrIw-NAOo3iWMy)

Gehen Sie das folgende Schulungsmaterial durch – „SQL Update“ – „SQL Aliases“:

[*https://www.w3schools.com/sql/sql\_update.asp*](https://www.w3schools.com/sql/sql_update.asp)

**Fragen:**

1. **Schreiben Sie eine Abfrage, um alle Datensätze mit "bar" in ihrem Namen auszuwählen?**
   * SELECT \* FROM Kunden  
     WHERE Name LIKE '%bar%';
2. **In welcher Reihenfolge werden die Ergebnisse standardmäßig angezeigt, wenn der ASC- oder DESC-Parameter nicht mit dem ORDER BY-Befehl angegeben wird?**
   * Ältester Eintrag > Jüngster Eintrag
   * From oldest entry to youngest entry
3. **Was ist der Unterschied zwischen IN- und BETWEEN-Operatoren?**
   * Der IN-Operator wählt Werte „in“ einem Bereich aus.
   * Der BETWEEN-Operator wählt Werte „innerhalb“ eines Bereiches aus.
   * The IN operator selects values „in“ a range.
   * The BETWEEN operator selects values “within” a range.
4. **Welcher Operator wird in der Abfrage für den Musterabgleich verwendet?**
   * LIKE.
5. **Schreiben Sie eine SQL-Abfrage, um alle Datensätze aus der Tabelle „Personen“ mit „Bar“ in ihrer Spalte „Name“ auszuwählen?**
   * SELECT \* FROM Personen

WHERE Name LIKE=‘%Bar%‘;

1. **Eine SQL-Abfrage, um eine Tabelle aus der Datenbank und dem Speicher zu löschen, während die Struktur der Tabelle intakt bleibt?**
   * TRUNCATE FROMTabellenname;
2. **Welches SQL-Wildcard-Zeichen steht für null oder mehr Zeichen?**
   * %
   * Prozent
3. **Welcher SQL-Wildcard steht für ein einzelnes Zeichen?**
   * \_
   * Underscore

**Aufgaben:**

1. **Schreiben Sie eine Abfrage, um die an die Mitarbeiter zu zahlenden Gesamtgehälter zu erhalten.**
   * SELECT Gehalt, sum(Gehalt) FROM Mitarbeitertabelle;
2. **Schreiben Sie eine Abfrage, um alle Vornamen aus der Mitarbeitertabelle in Großbuchstaben zu erhalten**
   * SELECT UPPER(Vorname) FROM Mitarbeitertabelle;
3. **Schreiben Sie eine Abfrage, um die ersten 3 Zeichen des Vornamens aus der Mitarbeitertabelle zu erhalten.**
   * SELECT SUBSTER(Vorname, 1, 3) FROM Mitarbeitertabelle;
   * SELECT \* FROM Mitarbeitertabelle (Employee table)

LEFT(Vorname; 3);

1. **Schreiben Sie eine Abfrage, um den Vornamen aus der Mitarbeitertabelle zu erhalten, nachdem Sie die Leerzeichen auf beiden Seiten entfernt haben.**
   * SELECT Mitarbeiter-ID, Vorname FROM Mitarbeitertabelle

UPDATE Mitarbeiter set Vorname=trim(Vorname) Where Mitarbeiter-ID=102;

1. **Schreiben Sie eine Abfrage, um die Länge der Mitarbeiternamen (Vorname, Nachname) aus der Mitarbeitertabelle abzurufen**
   * SELECT length(concat(Vorname, Nachname)) FROM Mitarbeitertabelle; (Employee table)
   * SELECT length(Vorname)+length(Nachname) FROM Mitarbeitertabelle;

1. **Schreiben Sie eine Abfrage, um die ersten 10 Datensätze aus einer Tabelle auszuwählen.**
   * SELECT \* FROM Tabellenname

LIMIT 10;

1. **Schreiben Sie eine Abfrage, um das monatliche Gehalt (runden 2 Dezimalstellen) von jedem einzelnen Mitarbeiter zu erhalten. Hinweis: Angenommen, das Gehaltsfeld enthält die Informationen zum Jahresgehalt.**
   * SELECT ROUND(Gehalt/12,2) as Monatliches GehaltMiterabeiter-ID, Vorname, Nachname
2. **Schreiben Sie eine Abfrage, um das Mindestgehalt aus der Mitarbeitertabelle abzurufen.**
   * SELECT Gehalt, min(Gehalt) FROM Mitarbeitertabelle; (Employee table)
   * Gehalt = Salary
3. **Schreiben Sie eine Abfrage, um das maximale Gehalt eines Mitarbeiters zu erhalten, der als „IT\_PROG“ arbeitet.**
   * SELECT \*, max(Gehalt) FROM Mitarbeitertabelle (Employee table)

WHERE Job-ID= ‘IT\_PROG’;

1. **Schreiben Sie eine Abfrage, um das durchschnittliche Gehalt und die Anzahl der Mitarbeiter in der Abteilung 90 zu erhalten.**
   * SELECT \*, avg(Gehalt), COUNT(Mitarbeiter-ID) FROM Mitarbeitertabelle

WHERE department-ID=90;

1. **Schreiben Sie eine Abfrage, um das höchste, niedrigste, summierte und durchschnittliche Gehalt aller Mitarbeiter zu erhalten.**
   * SELECT \*, max(Gehalt), min(Gehalt), sum(Gehalt), avg(Gehalt) FROM Mitarbeitertabelle;
2. **Schreiben Sie eine Abfrage, um die Anzahl der Mitarbeiter mit demselben Job zu erhalten.**
   * SELECT Job-ID, COUNT(Mitarbeiter-ID) FROM Mitarbeiter group by JOB\_ID;
3. **Schreiben Sie eine Abfrage, um die Differenz zwischen dem höchsten und dem niedrigsten Gehalt zu ermitteln.**
4. **Schreiben Sie eine SQL-Anweisung, um die E-Mail-Spalte der Mitarbeitertabelle für alle Mitarbeiter auf „nicht verfügbar“ zu ändern**
   * UPDATE Mitarbeitertabelle

SET E-Mail = ‚nicht verfügbar‘;

1. **Schreiben Sie eine SQL-Anweisung, um die E-Mail- und provisions\_pct-Spalte der Mitarbeitertabelle mit „NA“ und 0,10 für alle Mitarbeiter zu ändern.**
   * UPDATE Mitarbeitertabelle

SET E-Mail = ‘NA’, provisions\_pct = 0,10;

1. **Schreiben Sie eine SQL-Anweisung, um die E-Mail- und provisions\_pct-Spalte der Mitarbeitertabelle mit „na“ und 0,15 für die Mitarbeiter zu ändern, deren „department\_id“ 110 ist.**
   * UPDATE Mitarbeitertabelle

SET E-Mail= ‘na’, provisions\_pct = 0,15

WHERE department\_id = 110;

1. **Schreiben Sie eine SQL-Anweisung, um die E-Mail-Spalte der Mitarbeitertabelle mit „n.a.“ zu ändern. für diejenigen Mitarbeiter, deren Department\_id 80 ist und eine Provision von weniger als 0,20 % erhält.**
   * UPDATE Mitarbeitertabelle

SET E-Mail = ‚n.a.‘

WHERE department\_id = 80 AND Provision < 0,20%;

1. **Schreiben Sie eine SQL-Anweisung, um das Gehalt des Mitarbeiters auf 8000 zu ändern, dessen ID 105 ist, wenn das vorhandene Gehalt weniger als 5000 beträgt.**
   * UPDATE Mitarbeitertabelle

SET Gehalt = 8000

WHERE id = 105 AND Gehalt < 5000;

1. **Schreiben Sie eine SQL-Anweisung, um die Job-ID des Mitarbeiters mit der ID 118 in SH\_CLERK zu ändern, wenn der Mitarbeiter zur Abteilung gehört, deren ID 30 ist und die vorhandene Job-ID nicht mit SH beginnt.**
   * UPDATE Mitarbeitertabelle

SET Job-ID = ‚SH\_CLERK‘

WHERE id = 118 AND Abteilungs-ID = 30 AND Job-ID NOT LIKE ‘SH%’;

1. **Schreiben Sie eine SQL-Anweisung, um das Gehalt der Mitarbeiter in den Abteilungen 40, 90 und 110 gemäß den Unternehmensregeln zu erhöhen, wonach das Gehalt für die Abteilung 40 um 25 %, für die Abteilung 90 um 15 % und für die Abteilung 110 um 10 % erhöht wird Die restlichen Abteilungen bleiben gleich.**
   * ?
2. **Schreiben Sie eine SQL-Anweisung, um das Gehalt von Mitarbeitern, die als PU\_CLERK arbeiten, um 20 % und die Provisionsprozente um 0,10 zu erhöhen**
   * UPDATE Mitarbeiter SET Gehalt=Gehalt+Gehalt\*.20, commission\_pct+.10 WHERE Job-ID= ‚PU\_CLERK‘;
3. **Schreiben Sie eine Abfrage, um die an die Mitarbeiter zu zahlenden Gesamtgehälter zu erhalten**

**W3resource**

**SQL Wildcard & Special Operator**

1. Schreibe aus der Tabelle Verkäufer eine SQL-Abfrage, um die Details der Verkäufer zu finden, die aus der Stadt „Paris“ oder „Rom“ kommen.
   * SELECT \* FROM Verkäufer

WHERE Stadt = ‚Paris‘ OR Stadt = ‚Rom‘;

1. Schreibe aus der Tabelle Verkäufer eine SQL-Abfrage, um die Details der Verkäufer zu finden, die entweder aus „Paris“ oder „Rom“ kommen.
   * SELECT \* FROM Verkäufer

WHERE Stadt IN(‚Paris‘, ‚Rom‘);

1. Schreibe eine SQL-Abfrage, um die Details der Verkäufer zu ermitteln, die in anderen Städten als Paris und Rom leben.
   * SELECT \* FROM Verkäufer

WHERE Stadt NOT IN(‚Paris‘, ‚Rom‘);

Oder

* + SELECT \* FROM Verkäufer

WHERE NOT Stadt IN(‘Paris’, Rom’);

1. Schreibe eine SQL-Abfrage aus der Tabelle Kunde, um die Details aller Kunden abzurufen, deren ID zu einem der Werte 3007, 3008 oder 3009 gehört.
   * SELECT \* FROM Kunde

WHERE Kunden-ID IN (3007, 3008, 3009);

1. Schreibe aus der Tabelle Verkäufer eine SQL-Abfrage, um Vertriebsmitarbeiter zu finden, die Provisionen zwischen 0,12 und 0,14 erhalten.
   * SELECT \* FROM Verkäufer

WHERE Provision BETWEEN 0,12 AND 0,14;

1. Schreibe aus der Tabelle Bestellungen eine SQL-Abfrage, um Bestellungen zwischen 500 und 4000 auszuwählen (Anfangs- und Endwerte sind enthalten). Ohe Auftragsbestellungen Betrag 948,50 und 1983,43.
   * SELECT \*FROM Bestellungen

WHERE (Kaufpreis BETWEEN 500 AND 4000)

AND NOT Kaufpreis IN(948.50,1983.43);

1. Schreibe aus der Tabelle Verkäufer eine SQL-Abfrage, um die Details der Verkäufer abzurufen, deren Namen mit einem beliebigen Buchstaben zwischen „A“ und „L“ beginnen.
   * SELECT \* FROM Verkäufer

WHERE Name BETWEEN ‚A‘ AND ‚L‘;

1. Schreibe aus der Tabelle Verkäufer eine SQL-Abfrage, um die Details aller Vertriebsmitarbeiter zu finden, mit Ausnahme derjenigen, deren Namen mit einem beliebigen Buchstaben zwischen “A” und “L” beginnen.
   * SELECT \* FROM Verkäufer

WHERE Name NOT BETWEEN ‚A‘ AND ‚L‘;

1. Schreibe aus der Tabelle Kunde eine SQL-Abfrage, um die Details der Kunden abzufrufen, deren Namen mit dem Buchstaben “B” beginnen.
   * SELECT \* FROM Kunde

WHERE Name LIKE ‚B%‘;

1. Schreibe aus der Tabelle Kunde eine SQL-Abfrage, um die Details der Kunden zu ermitteln, deren Namen mit dem Buchstaben „n“ enden.
   * SELECT \* FROM Kunde

WHERE Name LIKE ‚%n‘;

1. Schreibe aus der Tabelle Verkäufer eine SQL-Abfrage, um die Detials der Verkäufer zu finden, deren Namen mit „N“ beginnen und das vierte Zeichen „I“ ist.
   * SELECT \* FROM Verkäufer

WHERE Name LIKE ‚N\_l%‘;

1. Schreibe aus der Tabelle Testtabelle eine SQL-Abfrage, um die Zeilen zu finden, in denen col1 den Unterstrich (\_) enthält.
   * SELECT \* FROM Testtabelle

WHERE col1 LIKE ‚%/\_%‘ ESCAPE ‚/‘;

1. Schreibe aus der Tabelle Testtabelle eine SQL-Abfrage, um die Zeilen zu identifizieren, in denen col1 nicht den Unterstrich (\_) enthält.
   * SELECT \* FROM Testtabelle

WHERE col1 LIKE NOT ‚%/\_%‘ ESCAPE ‚/‘;

1. Schreibe aus der Tabelle Testtabelle eine SQL-Abfrage, um Zeilen zu finden, in denen col1 den Schrägstrick (/) enthält.
   * SELECT \* FROM testtable

WHERE col1 LIKE '%//%' ESCAPE '/';

1. Schreibe aus der Tabelle Testtabelle eine SQL-Abfrage, um Zeilen zu finden, in denen col1 den Schrägstrick (/) nicht enthält.
   * SELECT \* FROM testtable

WHERE col1 LIKE NOT '%//%' ESCAPE '/';

1. Schreibe aus der Tabelle Testtabelle eine SQL-Abfrage, um Zeilen zu finden, in denen col1 die Zeichenfolge ( \_/ ) enthält.
   * SELECT \* FROM testtable

WHERE col1 LIKE '%/\_//%' ESCAPE '/';

1. Schreibe aus der Tabelle Testtabelle eine SQL-Abfrage, um Zeilen zu finden, in denen col1 die Zeichenfolge ( \_/ ) nicht enthält.
   * SELECT \* FROM testtable

WHERE col1 LIKE NOT '%/\_//%' ESCAPE '/';

1. Schreibe aus der Tabelle Testtabelle eine SQL-Abfrage, die Zeilen zu finden, in denen col1 das Zeichen Prozent ( % ) enthält.
   * SELECT \* FROM testtable

WHERE col1 LIKE '%/%%' ESCAPE '/';

1. Schreibe aus der Tabelle Kunde eine SQL-Abfrage, um alle Kunden zu finden, die keine Note haben.
   * SELECT \* FROM Kunde

WHERE Note IS NULL

1. Schreibe aus der Tabelle Kunde eine SQL-Abfrage, um alle Kunden zu finden, die einen Notenwert haben.
   * SELECT \* FROM Kunde

WHERE Note IS NOT NULL

**W3resource**

**SQL booleschen und relationalen Operatoren**

1. Schreibe aus der Tabelle Kunde eine SQL-Abfrage, um die Details von Kunden mit Notenwerten über 100 zu suchen.