



**SYSTEM INQUIRY LANGUAGE  
DEUTSCHER BENUTZERHANDBUCH,  
ERSTE AUSGABE**

**SIL Dokumentation**

**January 30, 2020**

## Contents

Einführung in SIL .....	5
Wie kann ich SIL herunterladen und installieren? .....	5
Verbindung zu RDBMS herstellen .....	11
Wie schreibe ich SIL-Anfragen? .....	20
SIL Syntax .....	21
SIL-Statements in der Sprache "SIL German" .....	22
Hinweis - SIL ist genau wie SQL .....	22
SIL SCHAFFEN DATENBANK Erklärung.....	22
SIL SCHAFFEN TABELLE Erklärung .....	22
SIL TROPFEN TABELLE Erklärung .....	23
SIL ÄNDERN TABELLE - ADDIEREN Säule .....	23
SIL SCHAFFEN INDEX Erklärung.....	24
SIL SCHAFFEN AUSSICHT Erklärung.....	24
SIL WÄHLEN Erklärung .....	25
SIL WÄHLEN DEUTLICH Erklärung .....	25
SIL WOHER Klausel .....	26
SIL UND, ODER und NICHT Operators.....	26
SIL AUFTRAG DURCH Stichwort .....	27
SIL ADDIEREN AUF_ZU Erklärung.....	29
SIL AKTUALISIEREN Erklärung .....	30
SIL LÖSCHEN Erklärung .....	30
SIL AUF Operator.....	31
SIL Aliases.....	31
SIL BEITRETEN .....	32
SIL UNION Operator .....	33
SIL UNION ALLE .....	33
SIL MINDEST() und MAX() Funktion .....	34
SIL ZÄHLUNG (), DURCHSCHNITT() und SUMME () Funktion .....	35
SIL SUMME () Funktion .....	36
The SIL ZWISCHEN Operator .....	36
SIL GRUPPE DURCH Erklärung.....	37
The SIL HABEN Klausel .....	38
SIL MÖGEN .....	39

SIL Wildcards.....	40
SIL Zwang .....	41
SIL PRIMÄR SCHLÜSSEL Zwang.....	41
SIL AUTO_ZUWACHS Field .....	42

**Wir empfehlen dringend, die virtuelle Tastatur zu installieren, um die Abfragen in einer anderen Sprache auszuführen**

## Einführung in SIL

SIL steht für System Inquiry Language.

SIL ist eine Abfragesprache, die es Benutzern ermöglicht, mit der Datenbank zu kommunizieren und Abfragen in verschiedenen Sprachen wie Arabisch, Hindi, Spanisch usw. zusätzlich zu Englisch zu schreiben.

SIL bietet eine Türöffnung, über die es Manipulationen an der internen Datenbank vornehmen kann, indem es Abfragen schreibt. Mit SIL können Sie Ihre Datenbank und alle DBMS-Operationen in Ihrer eigenen Sprache verwalten.

SIL kann Folgendes tun:

- SIL kann neue Datenbanken erstellen
- SIL kann neue Tabellen in einer Datenbank erstellen
- SIL kann Abfragen für eine Datenbank ausführen.
- SIL kann Daten aus einer Datenbank abrufen.
- SIL kann Datensätze in einer Datenbank einfügen, aktualisieren und löschen





Kurz gesagt, SIL kann alle Vorgänge in der Datenbank wie SQL ausführen, jedoch in einer benutzerfreundlichen Sprache.

## Wie kann ich SIL herunterladen und installieren?

SIL-Software ist verfügbar und kann von unserer offiziellen Website [www.sil.com](http://www.sil.com) heruntergeladen werden

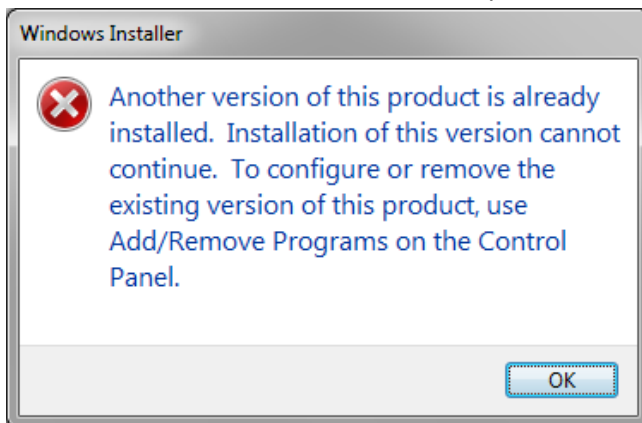
### Schritte zum Installieren von SIL:

Schritt 1: Laden Sie SIL von der Website herunter. Im Ordner befinden sich zwei Setup-Dateien wie in der Abbildung unten.

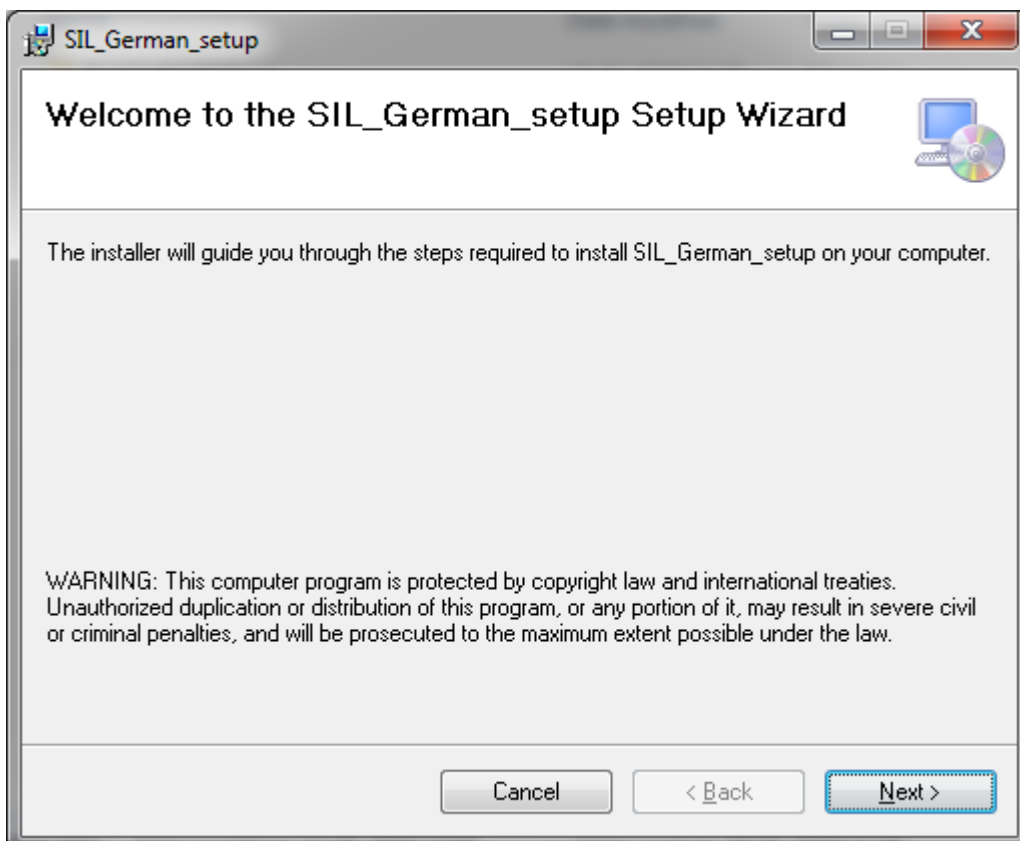
Name	Date modified	Type	Size
 DotNetFX40Client	18-02-2018 22:36	File folder	
 WindowsInstaller3_1	18-02-2018 22:36	File folder	
 setup.exe	18-02-2018 13:55	Application	418 KB
 SIL_German.msi	18-02-2018 13:55	Windows Installer ...	4,769 KB

Doppelklicken Sie auf die setup.exe und die Installation wird gestartet.

Hinweis: Wenn die Software bereits im System vorhanden ist, erscheint ein Fehlerbildschirm.



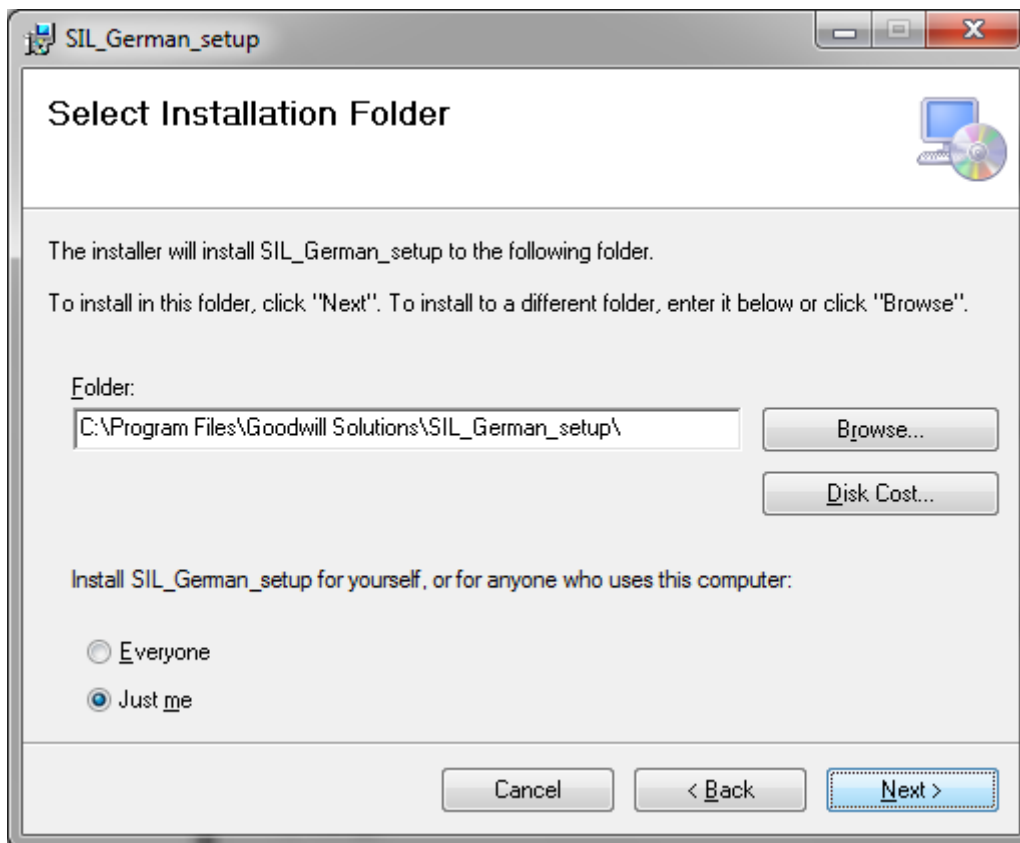
Schritt 2: Die Installation beginnt mit dem folgenden Fenster. Klicken Sie auf "Weiter", um zu beginnen.



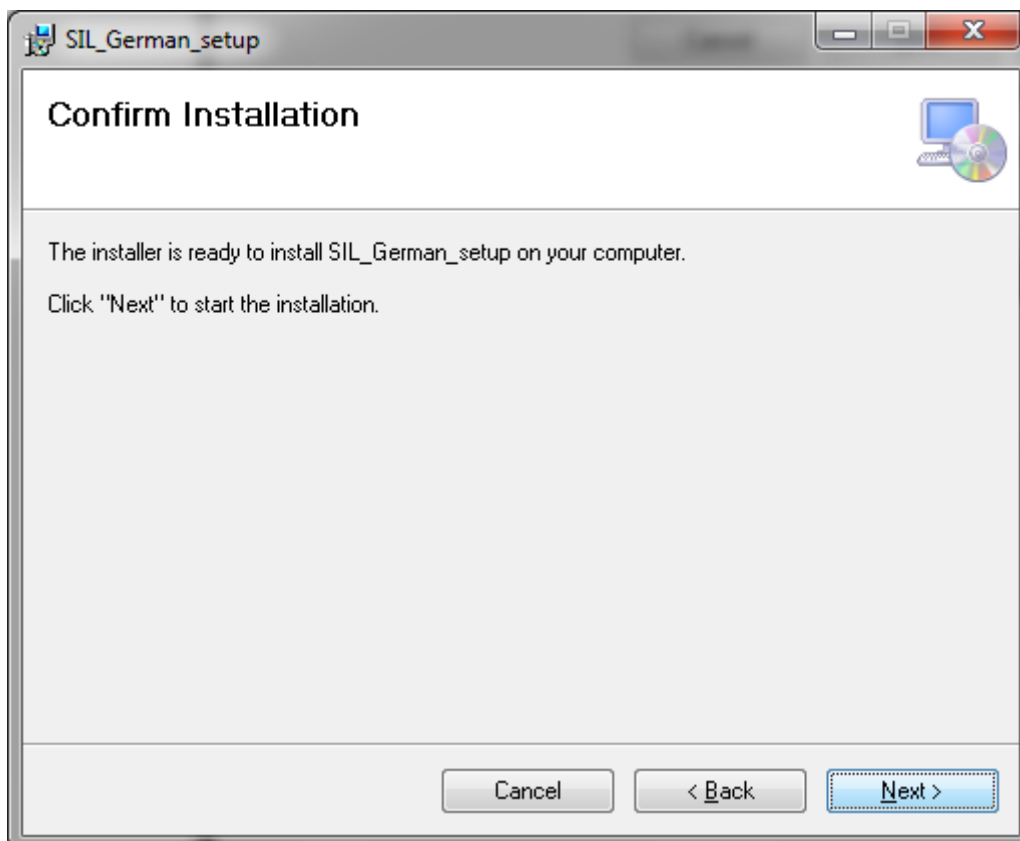
**Schritt 3:** Nachdem Sie auf "Weiter" geklickt haben, erhalten Sie den folgenden Bildschirm. Um SIL auf Ihrem System einzurichten, müssen Sie die Lizenzvereinbarung lesen und akzeptieren. Klicken Sie dann auf "Weiter".



**Schritt 4:** In diesem Schritt können Sie einen beliebigen Standort auf Ihrem System auswählen oder mit dem Standardstandort für die Einrichtung fortfahren. Klicken Sie dann auf "Weiter".

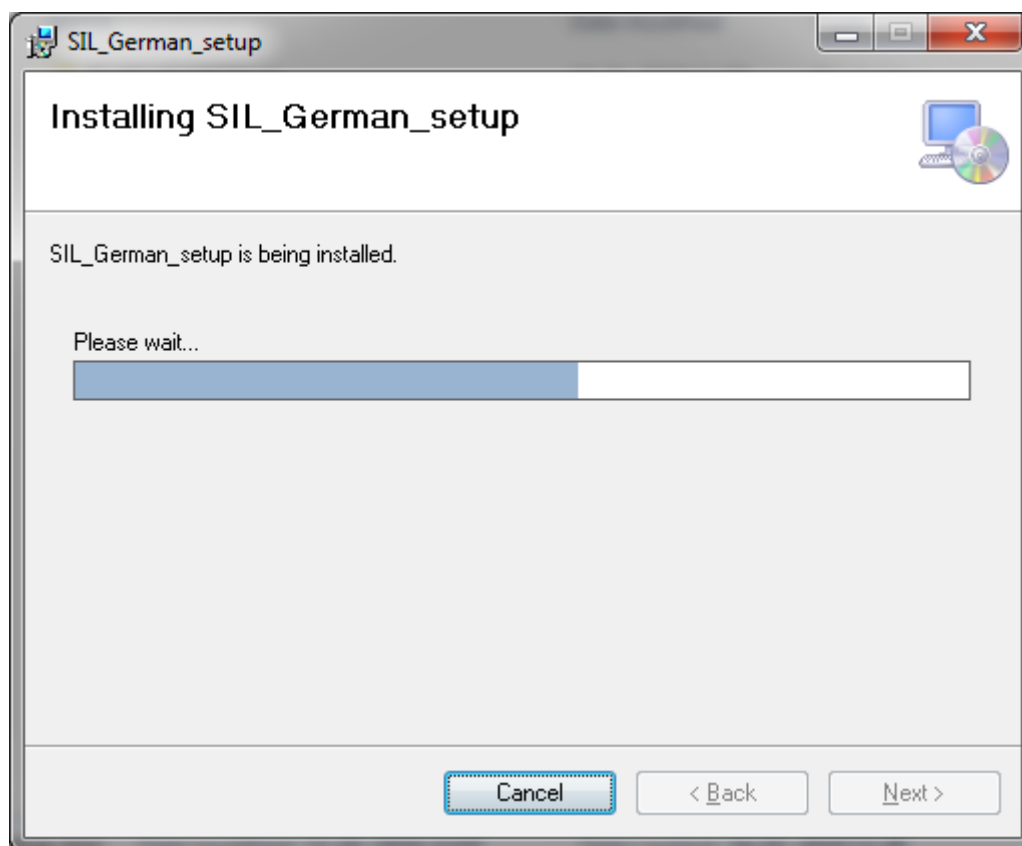


**Schritt 5:** Der nächste Schritt besteht darin, die Installation zu bestätigen. Und dann klicken Sie auf "Weiter".

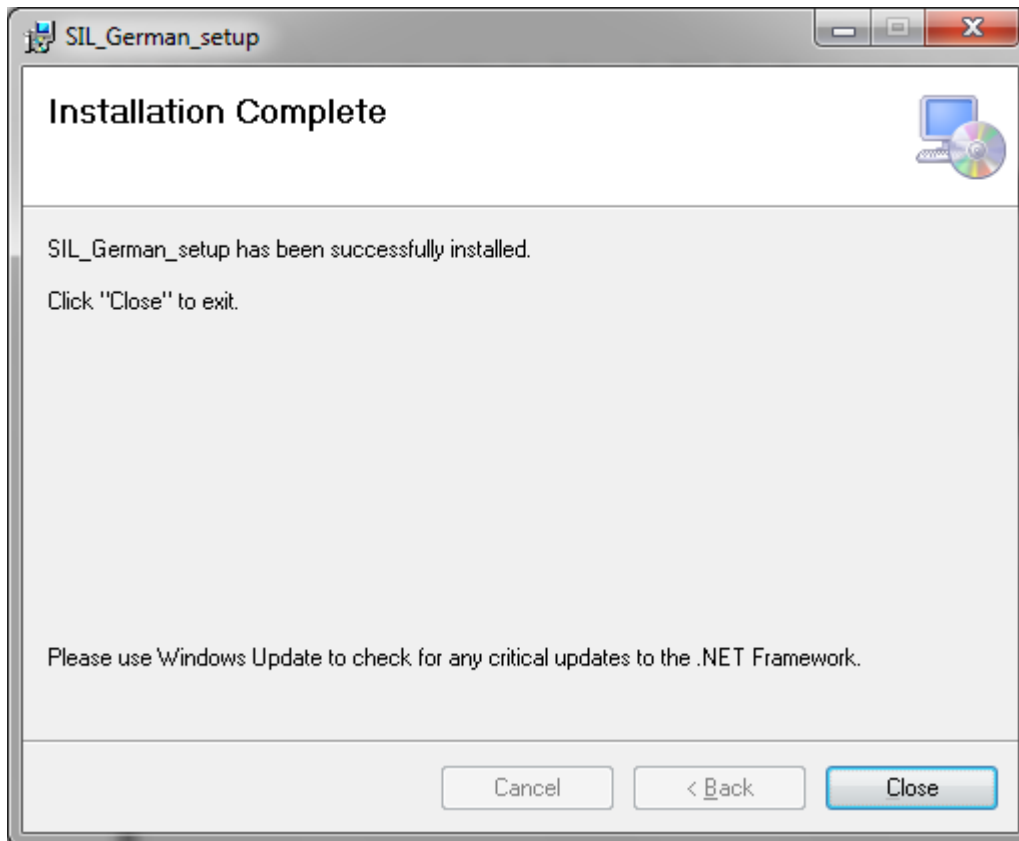




**Schritt 6:** Als nächstes sehen Sie das folgende Bild, das den Installationsfortschritt zeigt. Warten Sie einige Sekunden, um den Vorgang abzuschließen.



**Step 7:** The installation is complete



Nach Abschluss der Installation wird ein Shortcut-Symbol auf Ihrem Desktop erstellt.

Oder Sie können es von Start öffnen → Alle Programme → Systemanfrage Sprache

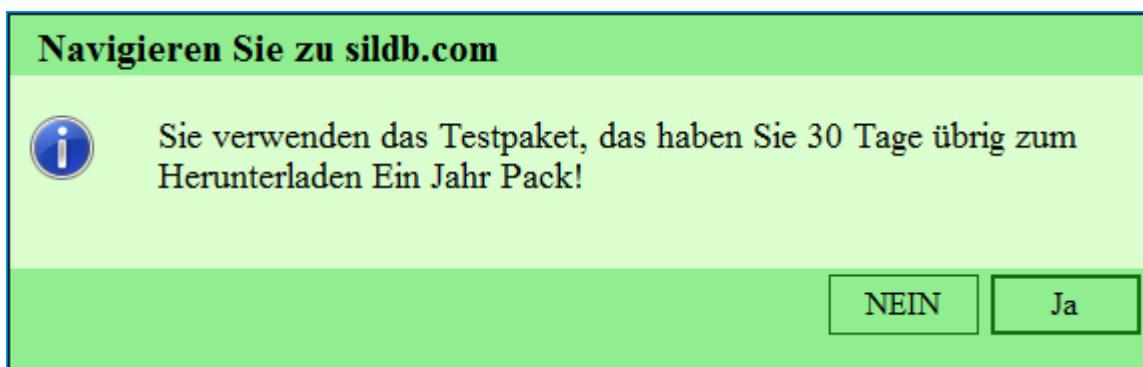
→SIL German

## Verbindung zu RDBMS herstellen

Durch SIL können wir uns mit verschiedenen RDBMS wie SQL Server, Oracle, MySQL Workbench und PostgreSQL verbinden. Zu einem Zeitpunkt kann irgendeines der obigen RDBMS verbunden sein und wir können Datenbankoperationen durchführen.

Im Folgenden finden Sie die Schritte zum Öffnen und Herstellen der Verbindung mit Datenbanken.

**Schritt 1:** Öffnen Sie die SIL-Software über das Desktop-Verknüpfungssymbol SIL English oder über das Startmenü 'System Inquiry Language folder'.

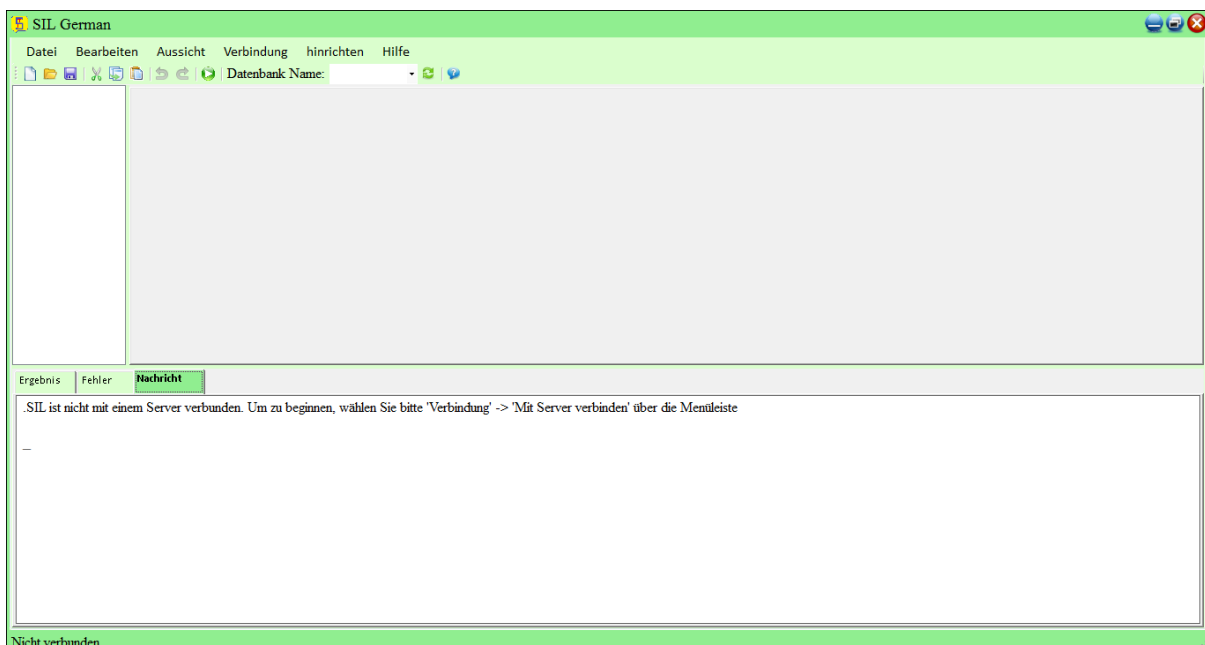


Click - YES to register SIL application

NEIN, um die 30-Tage-Trail-Version von SIL English Language zu verwenden

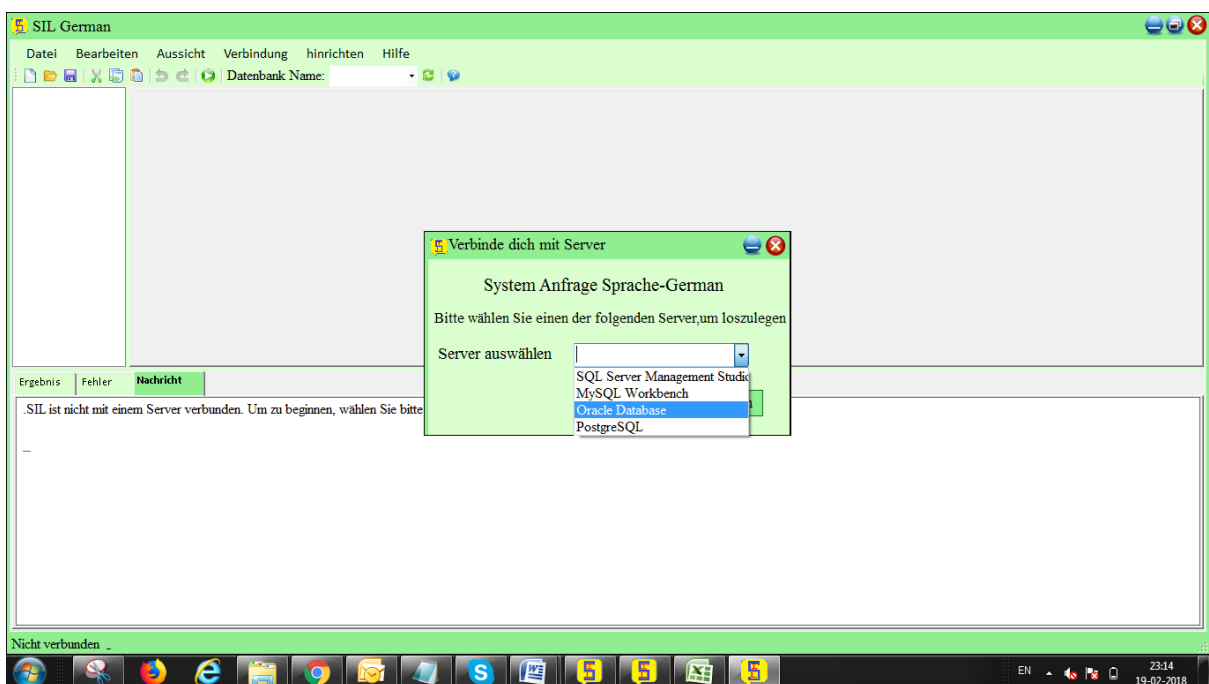
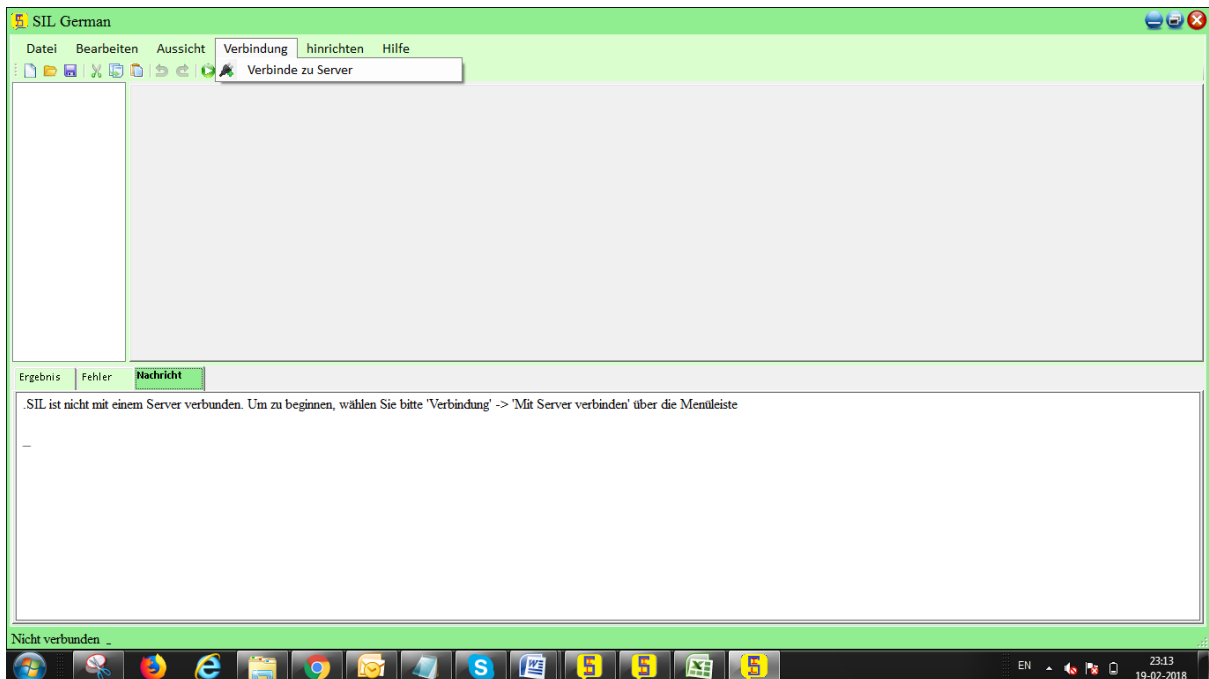


Nach dem Laden des SIL Editors öffnet sich Window. In der Statusleiste können Sie die Nachricht als "nicht verbunden" zusammen mit den Standortdetails für die Verbindung sehen. Es wird wie in der Abbildung unten aussehen.



**Schritt 2:** Stellen Sie eine Verbindung zu Datenbanken her, wählen Sie aus dem Menü 'Verbindung'

Dann erhalten Sie das folgende Fenster, in dem Sie eine der Datenbanken auswählen müssen.



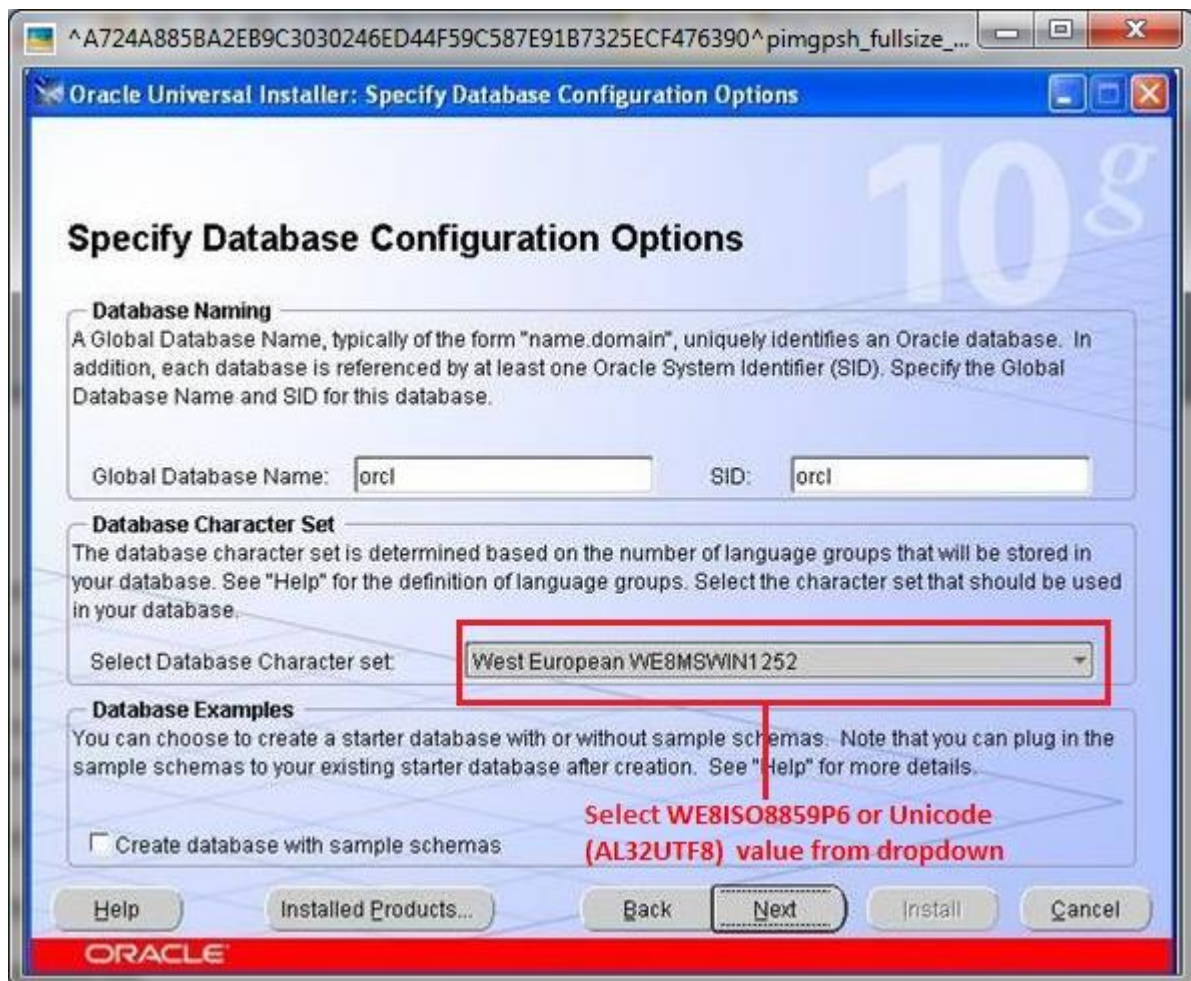
Anmerkung1: Wenn wir eine MYSQL-Workbench auswählen, müssen wir einen DATENBANK-Namen nur in ENGLISCHER Sprache und nicht in einer anderen Sprache

erstellen. Wenn wir DATENBANK in anderen Sprachen erstellen, werden die Tabellen nicht in einer anderen Sprache erstellt

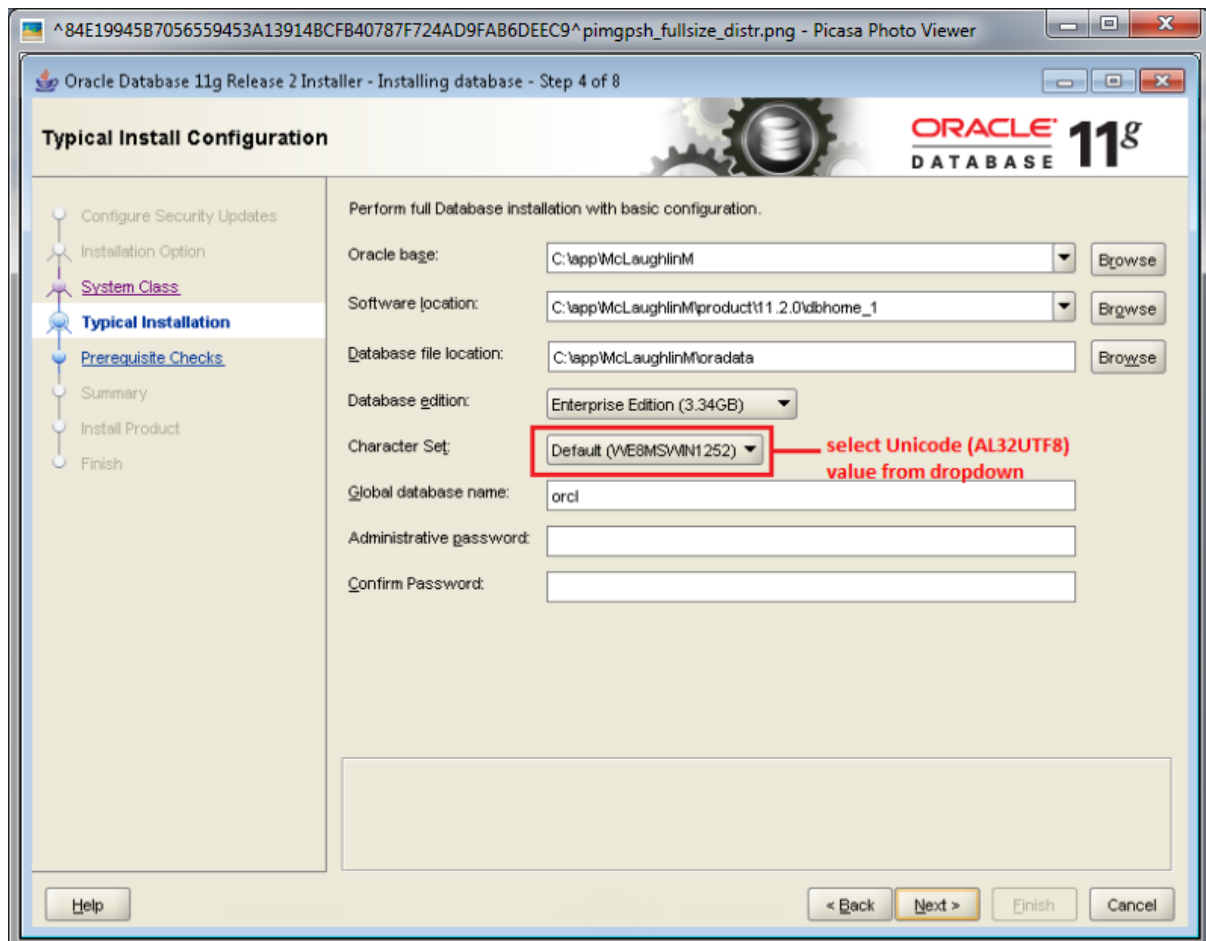
Anmerkung2: In SIL wird der "DELIMITER" in den gespeicherten Prozeduren und Funktionen nicht unterstützt

Anmerkung 3: Wenn wir uns für SQL Server entscheiden, sind einige Grundregeln zu beachten. Beim Speichern von Unicode-Daten muss die Spalte den Unicode-Datentyp aufweisen (nvarchar, nvarchar, ntext). Eine andere Regel ist, dass dem Wert während der Einfügung N vorangestellt werden muss

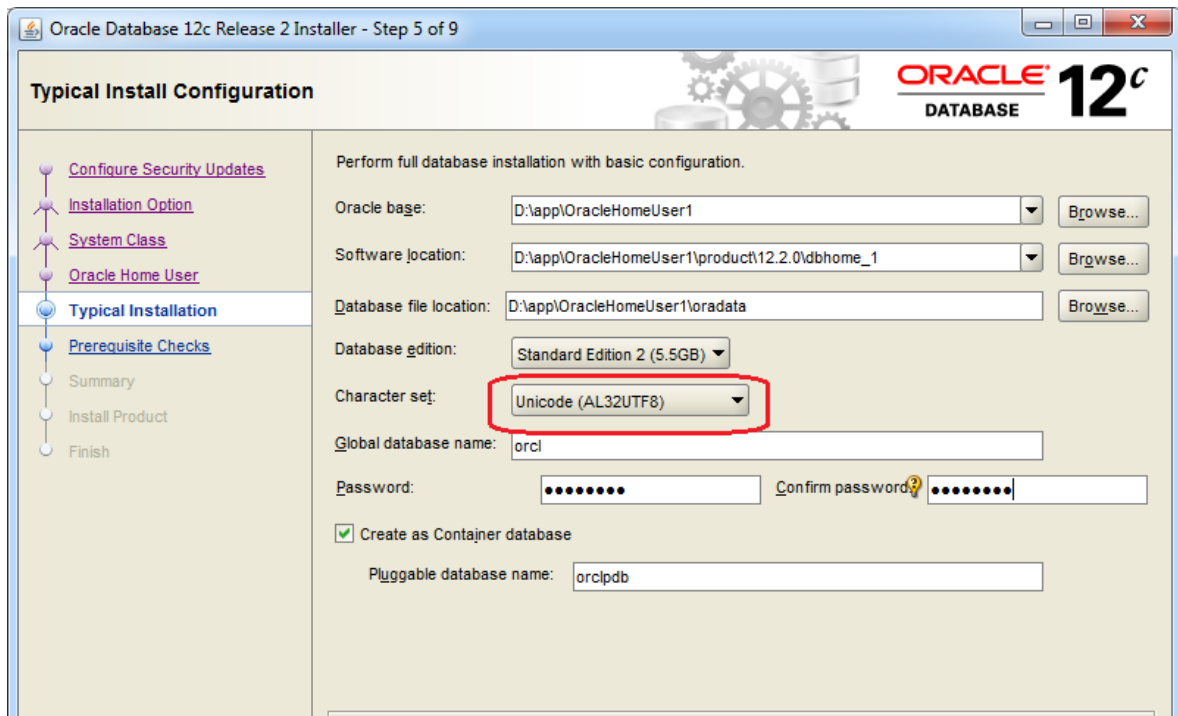
**Hinweis3:** Während der Installation von Oracle 10g ändern Sie den TROPFENdown-Wert für den Datenbankzeichensatz auf WE8ISO8859P6 oderUnicode (AL32UTF8), um arabische Zeichen in Oracle 10g wie unten gezeigt zu akzeptieren.



Wie im folgenden Screenshot des Fensters Oracle 11g Typische Installationskonfiguration gezeigt, ändern Sie den Tropfenwert für den Zeichensatz in Unicode (AL32UTF8), um arabische Zeichen in Oracle 11g zu akzeptieren.



Wie im folgenden Screenshot des Oracle 12c Typical Install Configuration-Fensters gezeigt, ändern Sie den Zeichensatz-Tropfendown-Wert in Unicode (AL32UTF8), um arabische Zeichen in Oracle 12c zu akzeptieren.



Ansonsten können wir die Einstellungen durch einen manuellen Prozess ändern, d. H.

Schritte zum Ändern des Zeichensatzwerts auf AL32UTF8 in oracle11g:

Zuerst können Sie das überprüfen NLS\_CHARACTERSET parameter Ihrer Datenbank mit dem

SQL\*PLUS command:-

Wählen \* von v\$nls\_parameters wobei Parameter = 'NLS\_CHARACTERSET';

Das Ergebnis sollte sein

PARAMETER

WERTE

NLS\_CHARACTERSET

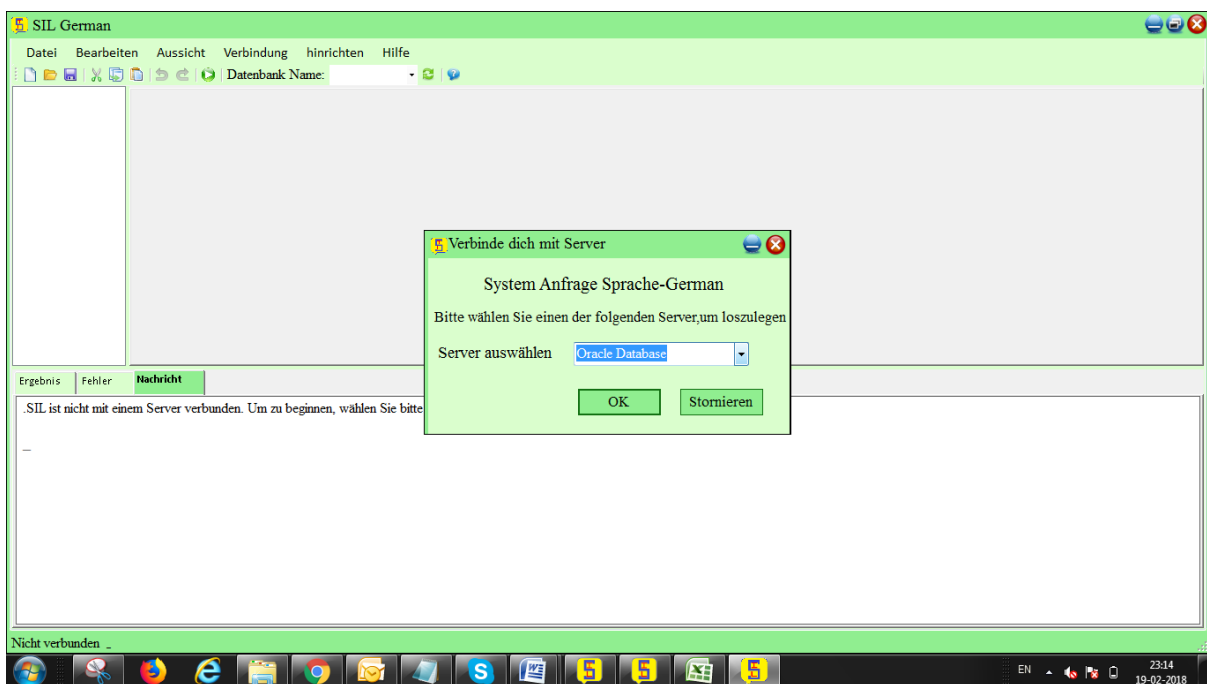
AR8MSWIN1256

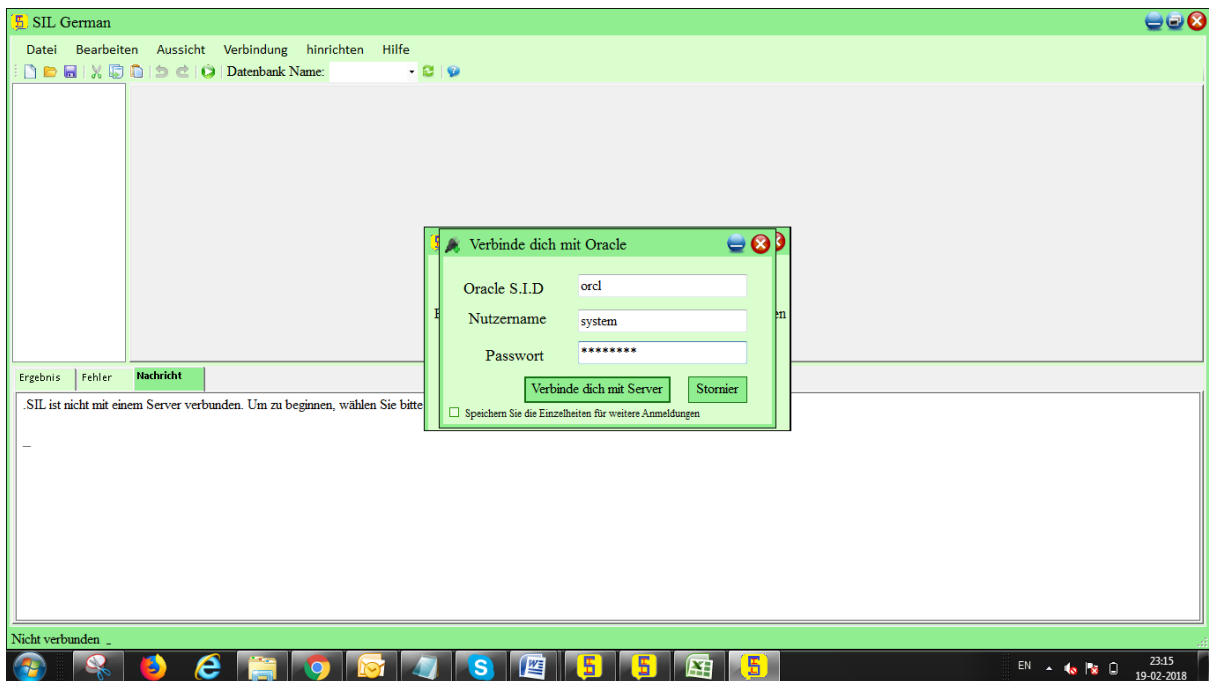
Ist dies nicht der Fall, müssen Sie den Wert dieses Parameters ändern, indem Sie:



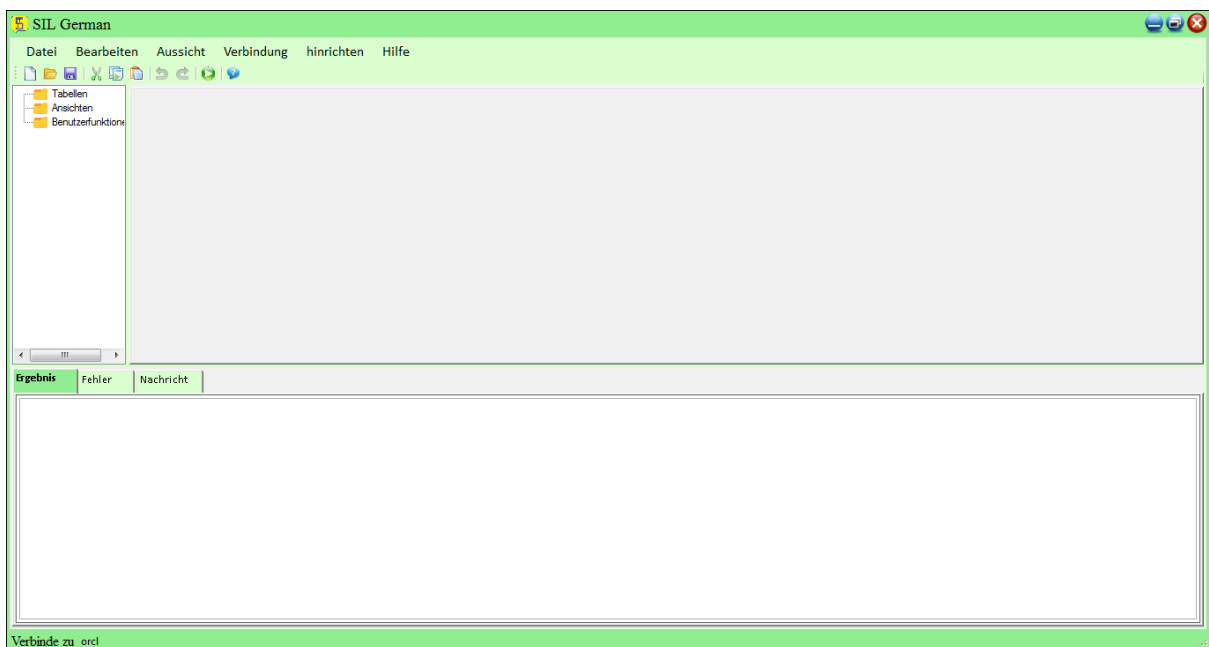
- Drücken Sie WINDOWS KEY + r auf Ihrer Tastatur
- Schreiben :- SQLPLUS sys as sysdba
- Drücken Sie die Eingabetaste und geben Sie das Passwort ein oder drücken Sie einfach eine andere Eingabetaste
- geben Sie die folgenden Befehle aus:
- SOFORT ABSCHALTEN
- STARTBESCHRÄNKUNG
- ÄNDERN DATENBANK CHARACTER SET INTERNAL\_USE AL32UTF8;
- ÄNDERN DATENBANK CHARACTER SET AL32UTF8;
- SOFORT ABSCHALTEN
- ANFANG
- Ausgang

Wählen Sie eine davon aus und Sie erhalten ein entsprechendes Anmeldefenster, in das Sie ihre Anmeldeinformationen eingeben müssen. Hier stellen wir eine Verbindung zu Oracle her Um Oracle zu verwenden, müssen Sie es installieren.



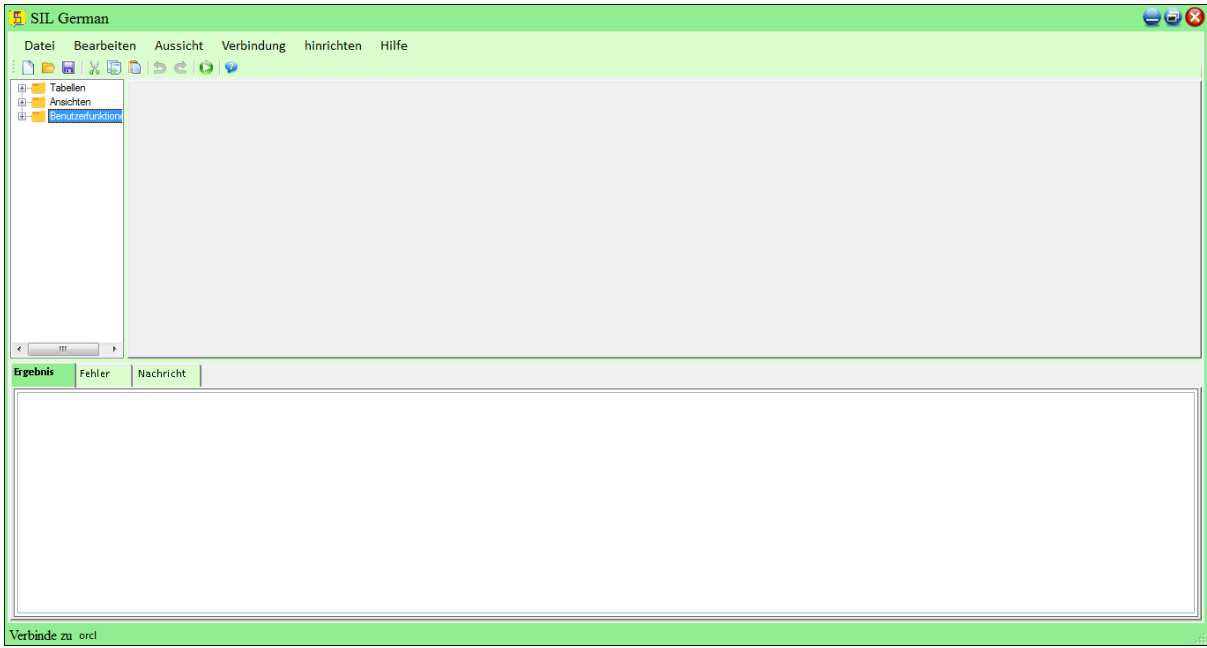


Nach dem obigen Schritt sind Sie mit dem ausgewählten RDBMS verbunden. Dann sieht das SIL Editor Fenster wie folgt aus:



In der Statusleiste wird der Servername der verbundenen Datenbank angezeigt. Daher wird die Verbindung erfolgreich hergestellt.

Und wir finden die Tabellen, Benutzerfunktionen und Ansichten auf der linken Seite in der TreeView-Art. Wie in der folgenden Screenshot gezeigt



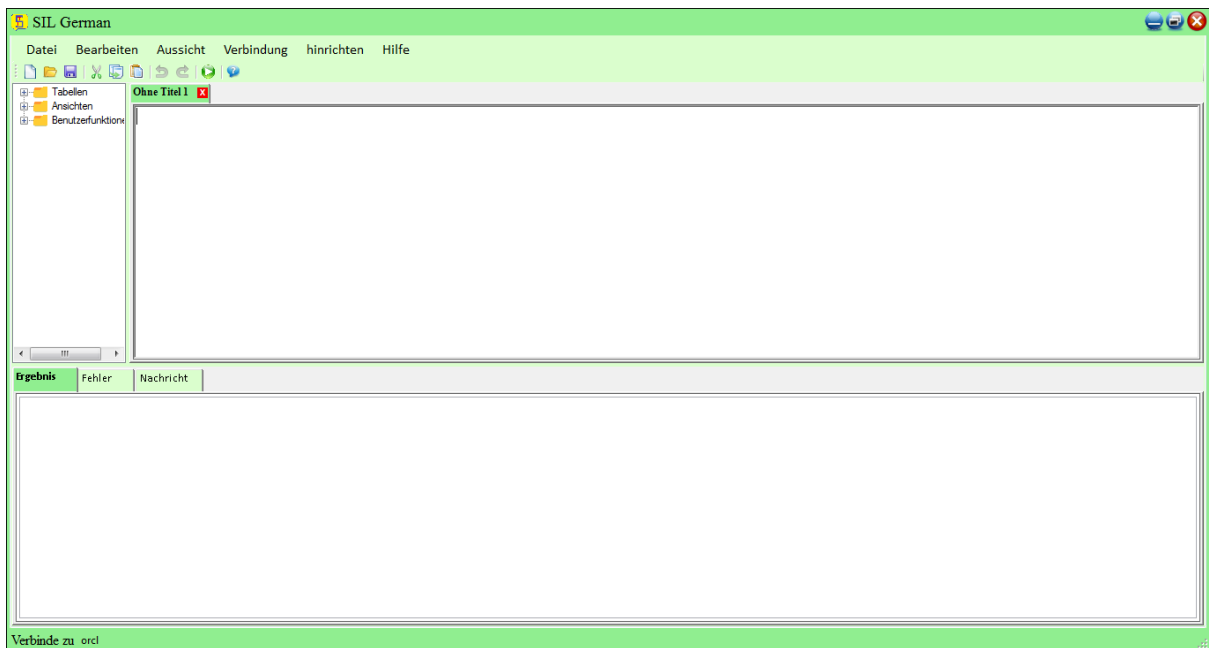
## Wie schreibe ich SIL-Anfragen?

Sobald die Verbindung hergestellt ist, können Sie Abfragen ausführen und Datenbankoperationen ausführen.

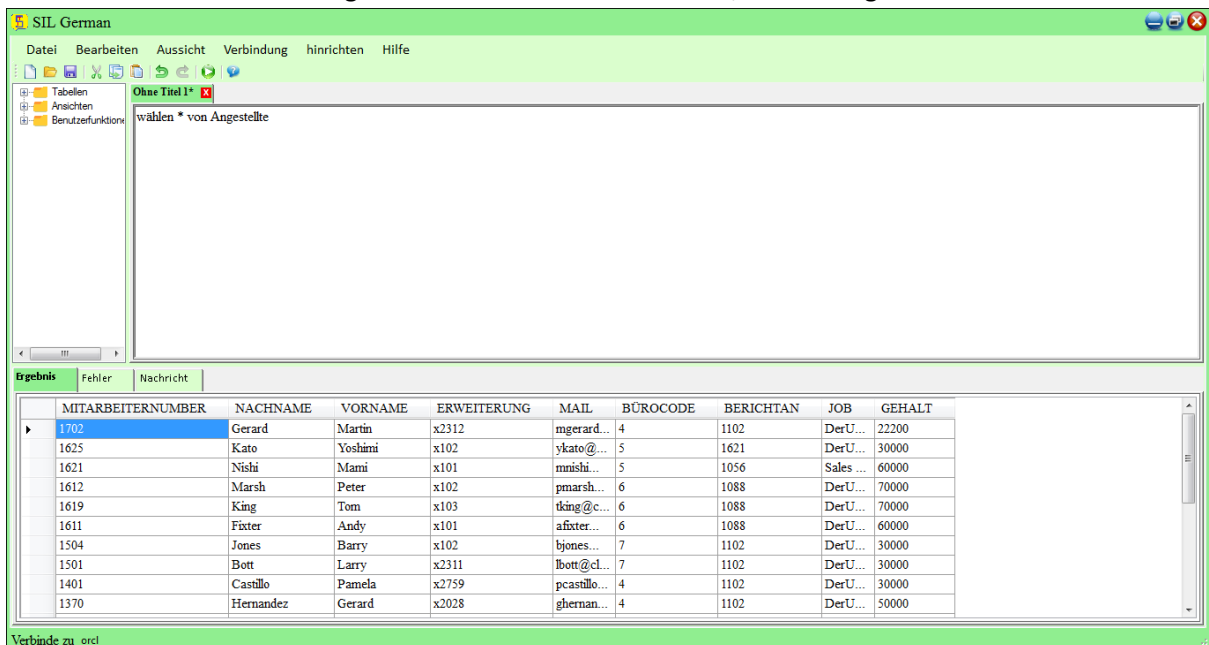
Ein Beispiel für die Ausführung wird in den folgenden Bildern gezeigt.

### BEISPIEL:

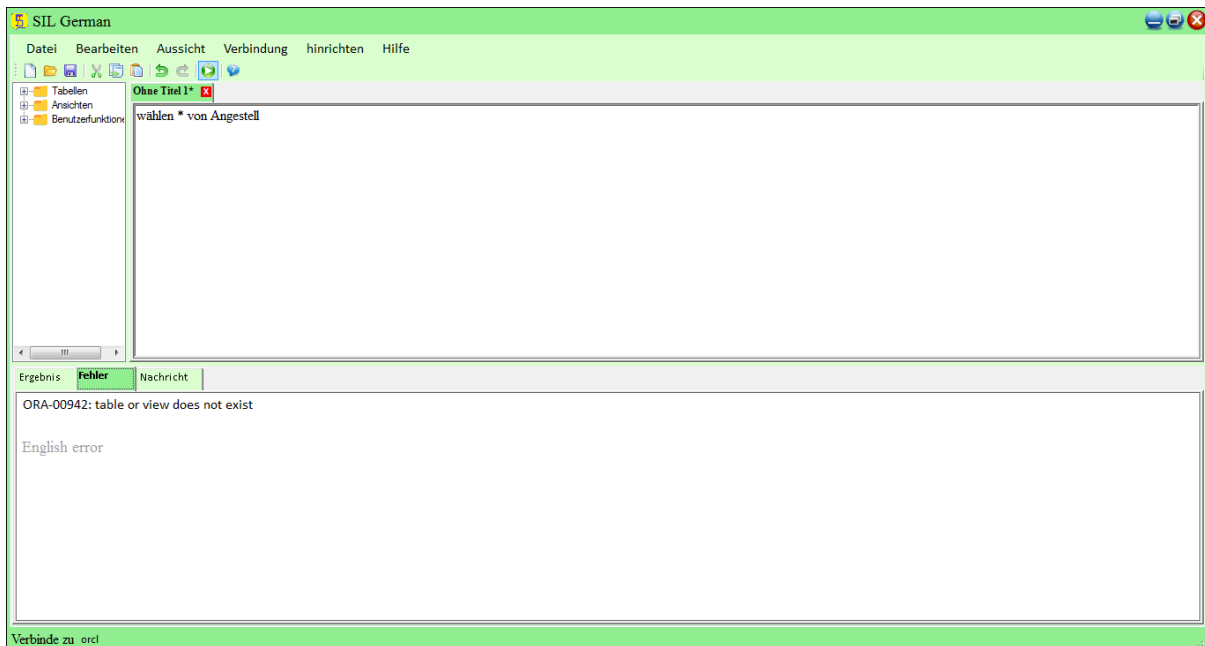
Wählen Sie zuerst die neue Seite aus, um die Abfragen auszuführen



Schreiben Sie dann die Abfrage und klicken Sie auf Ausführen, um das Ergebnis zu erhalten.



Im Falle eines Fehlers erhalten Sie einen Fehler in der Fehler-Registerkarte, wie unten gezeigt.



## SIL Syntax

Die Syntax von SIL ist identisch mit SQL

Hinweis: Alle in diesem Dokument gezeigten Syntaxen stammen aus PostgreSQL. Die Syntax für andere Datenbanken kann wenig variieren.

## SIL-Statements in der Sprache "SIL German"

### Hinweis - SIL ist genau wie SQL

#### SIL SCHAFFEN DATENBANK Erklärung

Hinweis - Die SCHAFFEN DATENBANK-Anweisung wird zum Erstellen einer neuen SQL-Datenbank verwendet.

Syntax:

```
SCHAFFEN DATENBANK Datenbank name;
```

Beispiel:

```
SCHAFFEN DATENBANK testDB;
```

#### SIL SCHAFFEN TABELLE Erklärung

Hinweis - Die Anweisung SCHAFFEN tabelle wird verwendet, um eine neue Tabelle in einer Datenbank zu erstellen.

Syntax:

```
SCHAFFEN TABELLE tabelle_name (
```

```
säule1datenart,
```

```
säule2datenart,
```

```
säule3datenart,....);
```

Beispiel:

```
schaffen tabelle Zahlungen (
```

```
KundeNummer int nicht null,
```

```
Kunde_Name varchar(50),
```

```
ZahlungDatum datum nicht null,
```

Menge dezimal nicht null,  
primär schlüssel (KundeNummer )  
);

KUNDENUMMER	KUNDE_NAME	ZAHLUNGSDATUM	MENGE
12	Jon	23-04-2017 00:00:00	500
11	Rob	23-05-2016 00:00:00	516
10	Ali	23-05-2016 00:00:00	516
122	Ahamad	23-05-2017 00:00:00	501
125	Martin	23-05-2014 00:00:00	501
1	Yoshimi	24-02-2014 00:00:00	501
2	Gerard	14-01-2014 00:00:00	501
4	Kato	19-01-2012 00:00:00	501
5	Kat	19-01-2011 00:00:00	501
6	Nishi	18-02-2011 00:00:00	501
7	Mami	18-02-2013 00:00:00	501
3	Tom	14-01-2012 00:00:00	501
8	Marsh	18-09-2013 00:00:00	509
9	Peter	09-10-2013 00:00	502

## SIL TROPFEN TABELLE Erklärung

Hinweis - Die Anweisung TROPFEN tabelle wird verwendet, um eine vorhandene Tabelle in einer Datenbank zu löschen.

Syntax:

```
TROPFEN TABELLE tabelle_name;
```

Beispiel:

```
TROPFEN TABELLE Angestellte;
```

## SIL ÄNDERN TABELLE - ADDIEREN Säule

Hinweis - Die Anweisung ÄNDERN TABELLE wird zum Hinzufügen, Löschen oder Ändern von Spalten in einer vorhandenen Tabelle verwendet.

Syntax:

ÄNDERN TABELLE tabelle\_name

ADDIEREN säule\_namedatenart;

Beispiel:

ändern tabelle Zahlungen

addieren Nachname varchar(50)

### **SIL SCHAFFEN INDEX Erklärung**

Hinweis - Die SCHAFFEN INDEX-Anweisung wird zum Erstellen von Indizes in Tabellen verwendet.

Syntax:

SCHAFFEN INDEX index\_name

AUF1 tabelle\_name (säule1, säule2, ...);

Beispiel:

schaffen index idex\_zahlungen

auf1 Zahlungen ( menge )

### **SIL SCHAFFEN AUSSICHT Erklärung**

Hinweis - In SIL ist eine Sicht eine virtuelle Tabelle, die auf dem Ergebnis einer SIL-Anweisung basiert.

Syntax:

SCHAFFEN AUSSICHT aussicht\_name WIE

WÄHLEN säule1, säule2, ...

VON tabelle\_name

WOHER bedingung;

Beispiel:

schaffen aussicht aussicht\_zahlungen wie



wählen \* von Zahlungen

## SIL WÄHLEN Erklärung

Hinweis - Die WÄHLEN-Anweisung wird verwendet, um Daten aus einer Datenbank auszuwählen.

Syntax1:

WÄHLEN säule1, säule2, ...

VON tabelle\_name;

Beispiel:

wählen nachname,vorname von Angestellte;

(Dies liefert nur Nachname und Vorname von Angestellten)

Syntax2:

WÄHLEN \* VON tabelle\_name;

Beispiel:

wählen \* von Angestellte;

(Dadurch werden alle Spalten von Angestellten zurückgegeben)

## SIL WÄHLEN DEUTLICH Erklärung

Hinweis - Die Anweisung WÄHLEN DEUTLICH wird verwendet, um nur bestimmte (unterschiedliche) Werte zurückzugeben.

Syntax:

WÄHLEN DEUTLICH säule1, säule2, ...

VON tabelle\_name;

Beispiel:

wählen deutlich vorname von Angestellte

## SIL WOHER Klausel

Hinweis - Die WOHER-Klausel wird nur zum Extrahieren der Datensätze verwendet, die eine bestimmte Bedingung erfüllen.

Syntax:

WÄHLEN \*VON

tabelle\_name

WOHER bedingung;

Beispiel:

wählen \* von Angestellte

woher Vorname ='Gerard' ;

## SIL UND, ODER und NICHT Operators

Hinweis - Die WOHER-Klausel kann mit UND-, ODER- und NICHT-Operatoren kombiniert werden.

Die Operatoren UND und ODER werden verwendet, um Datensätze basierend auf mehr als einer Bedingung zu filtern:

**UND** Syntax:

WÄHLEN säule1, säule2, ...

VON tabelle\_name

WOHER bedingung1 UND bedingung2 UND bedingung3 ...;

**UND** Beispiel:

wählen \* von Angestellte

woher nachname = 'Kato' und vorname = 'Yoshimi'

**ODER** Syntax:

WÄHLEN säule1, säule2, ...

VON tabelle\_name

WOHER bedingung1 ODER bedingung2 ODER bedingung3 ...;

**ODER** Beispiel:

wählen \* von Angestellte

woher nachname = 'Kato' oder vorname = 'Yoshimi'

**NICHT** Syntax:

WÄHLEN säule1, säule2, ...

VON tabelle\_name

WOHER NICHT bedingung;

**NICHT** Beispiel:

wählen \* von ANGESTELLTE

woher nicht NachName = 'Nishi'

## SIL AUFTRAG DURCH Stichwort

Hinweis - Das AUFTRAG DURCH-Schlüsselwort sortiert die Datensätze standardmäßig in aufsteigender Reihenfolge. Verwenden Sie das DESC-Schlüsselwort, um die Datensätze in absteigender Reihenfolge zu sortieren.

Syntax:

WÄHLEN säule1, säule2, ...

VON tabelle\_name

AUFTRAG DURCH säule1, säule2, ... ASC|DESC;

Beispiel:

wählen \* von Angestellte

auftrag durch VORNAME



## SIL ADDIEREN AUF\_ZU Erklärung

Hinweis - Die ADDIEREN AUF\_ZU-Anweisung wird verwendet, um neue Datensätze in eine Tabelle einzufügen.

Syntax1: (Addierening Werte to specific säules)

```
ADDIEREN AUF_ZU tabelle_name (säule1, säule2, säule3, ...)
```

```
WERTE (werte1, werte2, werte3, ...);
```

Beispiel1:

```
ADDIEREN AUF_ZU EMP_TABELLE (EMP_ID, ContactName, City,)
```

```
WERTE ('102', 'MARIYAM', 'Sheffield');
```

Syntax2: (inserting Werte tall the säules of the tabelle)

```
ADDIEREN AUF_ZU tabelle_name
```

```
WERTE (werte1, werte2, werte3, ...);
```

Beispiel2:

```
einfügen auf_zu Angestellte
```

```
werte (1702,'Gerard','Martin','x2312','mgerard@classicmodelcars.com','4',1102,'DerUmsatz Rep')
```

## SIL AKTUALISIEREN Erklärung

Hinweis - Die AKTUALISIEREN-Anweisung wird verwendet, um die vorhandenen Datensätze in einer Tabelle zu ändern.

Syntax:

AKTUALISIEREN tabelle\_name

EINSTELLEN säule1 = werte1, säule2 = werte2, ...

WOHER bedingung;

Beispiel:

aktualisieren Zahlungen

einstellen Kunde\_Name = 'ahamad'

woher KundeNummer = 122

## SIL LÖSCHEN Erklärung

Hinweis - Die LÖSCHEN-Anweisung wird verwendet, um vorhandene Datensätze in einer Tabelle zu löschen.

Syntax:

LÖSCHEN VON tabelle\_name

WOHER bedingung;

Beispiel:

löschen von Zahlungen1

woher KundeNummer =12

**To Löschen All Records**

Syntax:

LÖSCHEN VON tabelle\_name;

Beispiel: löschen von Zahlungen1

## SIL AUF Operator

Hinweis - Der Operator AUF ist eine Kurzform für mehrere ODER-Bedingungen.

Syntax:

WÄHLEN säule\_name(s)

VON tabelle\_name

WOHER säule\_name AUF (werte1, werte2, ...);

Beispiel:

wählen \* von Angestellte

woher nachname auf ('Kato')

## SIL Aliases

Hinweis - SIL-Aliase werden verwendet, um einer Tabelle oder einer Spalte in einer Tabelle einen temporären Namen zu geben.

Syntax1:

WÄHLEN säule\_name WIE alias\_name

VON tabelle\_name;

Beispiel1:

wählen nachname

wie name von Angestellte

## SIL BEITRETEN

Hinweis - Eine BEITRETEN-Klausel wird verwendet, um Zeilen aus zwei oder mehr Tabellen basierend auf einer verwandten Spalte zwischen ihnen zu kombinieren.

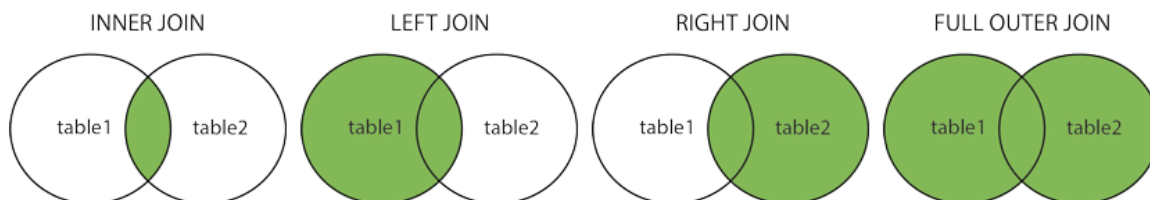
### Types of SIL BEITRETEN

(INNERE) BEITRETEN : Gibt Datensätze mit übereinstimmenden Werten in beiden Tabellen zurück

LINKS (ÄUßERE1) BEITRETEN : Gibt alle Datensätze aus der linken Tabelle und die übereinstimmenden Datensätze aus der rechten Tabelle zurück

RECHTS (ÄUßERE1) BEITRETEN : Gibt alle Datensätze aus der rechten Tabelle und die übereinstimmenden Datensätze aus der linken Tabelle zurück

VOLL (ÄUßERE1) BEITRETEN : Gibt alle Datensätze zurück, wenn eine Übereinstimmung in der linken oder rechten Tabelle vorhanden ist





Syntax:

WÄHLEN säule\_name(s)

VON tabelle1

INNERE BEITRETEN tabelle2 ON tabelle1.säule\_name = tabelle2.säule\_name;

Beispiel:wählen Kunden.Nachname, Kunden.Kunde\_Name, Zahlungen.ZahlungDatum

von Zahlungen

innere beitreten Kunden auf1 Kunden.KundeNummer=Zahlungen.KundeNummer

## SIL UNION Operator

Hinweis - Der UNION-Operator wird verwendet, um die Ergebnismenge von zwei oder mehr WÄHLEN zu kombinieren Anweisungen, gibt nur eindeutige Werte zurück

Syntax:

WÄHLEN säule\_name(s) VON tabelle1

UNION

WÄHLEN säule\_name(s) VON tabelle2;

Beispiel:

wählen Kunde\_Name von Kunden

union

wählen Kunde\_Name von Zahlungen

## SIL UNION ALLE

Hinweis - Gibt auch doppelte Werte zurück

Syntax:

WÄHLEN säule\_name(s) VON tabelle1

UNION ALLE

WÄHLEN säule\_name(s) VON tabelle2;

Beispiel:

wählen Kunde\_Name von Kunden

union alle

wählen Kunde\_Name von Zahlungen

## SIL MINDEST() und MAX() Funktion

Hinweis: Diese Funktion gibt den kleinsten Wert der ausgewählten Spalte zurück.

MINDEST () Syntax:

WÄHLEN MINDEST (säule\_name)VON tabelle\_name

Beispiel:

wählen mindest (kundennummer) von Kunden

Nichte- Diese Funktion gibt die größten Werte der gewählten Spalte zurück.

MAX () Syntax:

WÄHLEN MAX (säule\_name)VON tabelle\_name;

Beispiel:

wählen max (kundennummer) von Kunden

## SIL ZÄHLUNG (), DURCHSCHNITT() und SUMME () Funktion

Hinweis: Die Funktion SIL ZÄHLUNG () wird verwendet, um die Anzahl der Zeilen in einer Tabelle zu finden

ZÄHLUNG () Syntax1:

WÄHLEN ZÄHLUNG (säule\_name)VON tabelle\_name

ZÄHLUNG () Syntax2:

WÄHLEN ZÄHLUNG (säule\_name)VON tabelle\_name

WOHER bedingung;

Beispiel:

wählen zählung (kundennummer) von Kunden

Hinweis: Die Funktion DURCHSCHNITT () gibt den Durchschnittswert einer numerischen Spalte zurück

DURCHSCHNITT () Syntax1:

WÄHLEN DURCHSCHNITT(säule\_name)VON tabelle\_name

DURCHSCHNITT () Syntax2:

WÄHLEN DURCHSCHNITT (säule\_name)VON tabelle\_name

WOHER bedingung;

Beispiel:

wählen durchschnitt (kundennummer) von Kunden

## SIL SUMME () Funktion

Hinweis: Mit der Funktion SIL SUMME () wird die Gesamtzahl der Spalten zurückgegeben

SUMME () Syntax1:

WÄHLEN SUMME (säule\_name)VON tabelle\_name

SUMME () Syntax2:

WÄHLEN SUMME (säule\_name)VON tabelle\_name

WOHER bedingung;

Beispiel:

wählen summe (kundennummer) von Kunden

## The SIL ZWISCHEN Operator

Hinweis: Der ZWISCHEN-Operator wählt Werte innerhalb eines bestimmten Bereichs aus

ZWISCHEN Syntax:

WÄHLEN säule\_name(s)VON tabelle\_name

WOHER säule\_name ZWISCHEN werte1 UND werte2;

Beispiel :

wählen \* von ANGESTELLTE

woher Gehalt zwischen 50000 und 70000

## SIL GRUPPE DURCH Erklärung

Hinweis: Die GRUPPE DURCH-Anweisung wird mit Aggregatfunktionen wie ZÄHLUNG, MAX, MINDEST, SUMME , DURCHSCHNITT verwendet, indem mehr als eine Spalte kombiniert wird, um das Ergebnis zu erhalten

GRUPPE DURCH Syntax:

WÄHLEN säule\_name(s)VON tabelle\_name

WOHER bedingung

GRUPPE DURCH säule\_name(s)

AUFTRAG DURCH säule\_name(s);

Beispiel1:

wählen zählung (Gehalt) ,NachName von ANGESTELLTE

gruppe durch NachName

Beispiel2:

wählen zählung (Gehalt) ,NachName von ANGESTELLTE

gruppe durch NachName auftrag durch NachName

## The SQL HAVING Clause

Anmerkung: Die HAVING-Klausel wird mit der WHERE-Klausel verwendet, weil die WHERE-Klausel nicht mit Aggregationsfunktionen verwendet werden konnte

HAVING Syntax:

WÄHLEN säule\_name(s) VON tabelle\_name WOHER bedingung

GRUPPE DURCH säule\_name(s)

HAVING bedingung

AUFTRAG DURCH säule\_name(s);

Beispiel1:

wählen zahlung (Gehalt) ,NachName von ANGESTELLTE

gruppe durch NachName having zahlung (NachName) > 1

## SIL MÖGEN

Hinweis: Der MÖGEN-Operator wird in einer WOHER-Klausel verwendet, um nach einem bestimmten Muster in einer Spalte zu suchen.

In Verbindung mit dem MÖGEN-Operator werden zwei Platzhalter verwendet:

% - Das Prozentzeichen steht für null, eins oder mehrere Zeichen

\_ - Der Unterstrich repräsentiert ein einzelnes Zeichen

MÖGEN Syntax:

WÄHLEN säule1, säule2, ...

VON tabelle\_name

WOHER säuleN MÖGEN pattern;

Hier sind einige Beispiele, die verschiedene MÖGEN-Operatoren mit '%' und '\_' Wildcards zeigen:

WIE Operator Beschreibung

WO CustomerName MÖGEN 'a%'	Findet alle Werte, die mit beginnen "a"
WO CustomerName MÖGEN '%a'	Findet alle Werte, die mit enden "a"
WO CustomerName MÖGEN '%or%'	Findet alle Werte mit "or" in einer beliebigen Position
WO CustomerName MÖGEN '_r%'	Findet alle Werte mit "r" an der zweiten Position
WO CustomerName MÖGEN 'a_%_%'	Sucht nach Werten, die mit "a" beginnen und mindestens 3 Sind Zeichen in der Länge
WO CustomerName MÖGEN 'a%o'	Findet alle Werte, die mit "a" beginnen und mit "o" enden

## SIL Wildcards

SIL Wildcard Figuren:

Hinweis: Ein Platzhalterzeichen wird verwendet, um andere Zeichen in einer Zeichenfolge zu ersetzen.

In Verbindung mit dem MÖGEN-Operator werden zwei Platzhalter verwendet:

% - Das Prozentzeichen steht für null, eins oder mehrere Zeichen

\_ - Der Unterstrich repräsentiert ein einzelnes Zeichen

Die Platzhalter können auch in Kombinationen verwendet werden!

Hier sind einige Beispiele, die verschiedene MÖGEN-Operatoren mit '%' und '\_' Wildcards zeigen:

WIE Operator Beschreibung

WO CustomerName MÖGEN 'a%'	Findet alle Werte, die mit beginnen "a"
WO CustomerName MÖGEN '%a'	Findet alle Werte, die mit enden "a"
WO CustomerName MÖGEN '%or%'	Findet alle Werte mit "or" in einer beliebigen Position
WO CustomerName MÖGEN '_r%'	Findet alle Werte mit "r" an der zweiten Position
WO CustomerName MÖGEN 'a_%_%'	Sucht nach Werten, die mit "a" beginnen und mindestens 3 Sind Zeichen in der Länge
WO CustomerName MÖGEN 'a%o'	Findet alle Werte, die mit "a" beginnen und mit "o" enden



## SIL Zwang

Hinweis-SIL-Einschränkungen werden verwendet, um Regeln für Daten in einer Tabelle anzugeben.

Syntax:

```
SCHAFFEN TABELLE tabelle_name (  
säule1datenart zwang,  
säule2datenart zwang,  
säule3datenart zwang,  
....  
);
```

Die folgenden Einschränkungen werden normalerweise in SIL verwendet:

**NICHT NULL** - Stellt sicher, dass eine Spalte keinen NULL-Wert haben darf

**EINZIGARTIG** - Stellt sicher, dass alle Werte in einer Spalte unterschiedlich sind

**PRIMÄR SCHLÜSSEL** - Eine Kombination aus NICHT NULL und EINZIGARTIG. Identifiziert jede Zeile in einer Tabelle

eindeutig

**FOREIGN SCHLÜSSEL** - Identifiziert eine Zeile / einen Datensatz in einer anderen Tabelle eindeutig

**PRÜFEN** - Stellt sicher, dass alle Werte in einer Spalte eine bestimmte Bedingung erfüllen

**VOREINSTELLUNG** - Legt einen Standardwert für eine Spalte fest, wenn kein Wert angegeben ist

**INDEX** - Verwenden Sie diese Option zum schnellen Erstellen und Abrufen von Daten aus der

Datenbank

## SIL PRIMÄR SCHLÜSSEL Zwang

Hinweis: Die PRIMÄR SCHLÜSSEL-Einschränkung identifiziert jeden Datensatz in einer Datenbanktabelle eindeutig.

## SIL PRIMÄR SCHLÜSSEL auf SCHAFFEN TABELLE

Der folgende SIL erstellt einen PRIMÄRSCHLÜSSEL für die Spalte "ID", wenn die Tabelle "Personen" erstellt wird:

SQL Server / Oracle / MS Access:

SCHAFFEN TABELLE Persons (

    ID int NICHT NULL PRIMÄR SCHLÜSSEL,

    LastNamevarchar(255) NICHT NULL,

    FirstNamevarchar(255),

    Age int

);

MySQL / SQL Server / Oracle / MS Access:

schaffen tabelle Zahlungen

(

    KundeNummer int nicht null,

    Kunde\_Name varchar(50),

    ZahlungDatum datum nicht null,

    Menge dezimal nicht null,

    primär schlüssel (KundeNummer )

);

## SIL AUTO\_ZUWACHS Field

Hinweis: Auto\_zuwachs ermöglicht die automatische Generierung einer eindeutigen Nummer, wenn ein neuer Datensatz in eine Tabelle eingefügt wird.

Syntax for MySQL:

```
SCHAFFEN TABELLE Persons (  
    ID int NICHT NULL AUTO_ZUWACHS,  
    LastNamevarchar(255) NICHT NULL,  
    FirstNamevarchar(255),  
    Age int,  
    PRIMÄR SCHLÜSSEL (ID)  
);
```

Syntax for SQL Server:

```
SCHAFFEN TABELLE Persons (  
    ID intIDENTITY(1,1) PRIMÄR SCHLÜSSEL,  
    LastNamevarchar(255) NICHT NULL,  
    FirstNamevarchar(255),  
    Age int  
);
```

Syntax for Access:

```
SCHAFFEN TABELLE Persons (  
    ID Integer PRIMÄR SCHLÜSSEL AUTO_ZUWACHS,  
    LastNamevarchar(255) NICHT NULL,  
    FirstNamevarchar(255),  
    Age int  
);
```

Syntax for Oracle:

```
schaffen sequenz seq_emp
```

```
minwerte 1
```

```
anfang mit 1
```

```
zuwachs durch 1
```

```
zwischenspeicher 10;
```