

## 1. Erstelle einen Datenbank:

```
CREATE DATABASE db_name;
```

. kann keinen anderen Datenbank mit dem selben Name erstellt wurde!

### 1.2 Erstelle einen Datenbank, falls diese nicht bereits existiert:

```
CREATE DATABASE IF NOT EXISTS db_name;
```

## 2. lösche einen Datenbank:

```
DROP DATABASE db_name;
```

### 2.2 lösche einen Datenbank, falls diese bereits existiert:

```
DROP DATABASE IF EXISTS db_name;
```

## 3. Tabellen in Datenbanken erstellen:

```
CREATE TABLE table_name (  
    column_name_1 datatype,  
    column_name_2 datatype,  
    column_name_3 datatype,  
    ....  
);
```

- مكان الكلمة `table_name` نضع الإسم الذي سيتم إعطاؤه للجدول.
- مكان كل كلمة `column_name` نضع إسم العمود الذي سيتم إنشاؤه في الجدول.
- مكان كل كلمة `datatype` نضع نوع البيانات التي يمكن إدخالها في العمود.

### إنشاء جدول جديد في قاعدة البيانات

الإستعلام التالي ينشئ جدول جديد إسمه `books` يتألف من أربع أعمدة.

مثال

```
CREATE TABLE books (  
    title VARCHAR(200),  
    number_of_pages INT,  
    author_name VARCHAR(50),  
    publishing_date DATE  
);
```

-- العمود الأول إسمه title و نوعه VARCHAR و يمكنه أن يحتوي على 200 حرف --  
-- العمود الثاني إسمه number\_of\_pages و نوعه INT, أي يمكن وضع أرقام صحيحة فيه --  
-- العمود الثالث إسمه author\_name و نوعه VARCHAR و يمكنه أن يحتوي على 50 حرف --  
-- العمود الرابع إسمه publishing\_date و نوعه DATE, أي يمكن وضع تواريخ فيه فقط --

- . before wir die Tabelle erstellen, sollen wir erst einen Datenbank auswählen.
- . kann keine andere Tabelle mit dem selben Name erstellt wurde!

3.2 Erstelle eine Tabelle, falls sie nicht bereits existiert:

```
CREATE TABLE IF NOT EXISTS table_name;
```

---

#### 4. Lösche eine Tabelle:

```
DROP TABLE table_name;
```

4.1 lösche eine Tabelle, falls sie bereits existiert:

```
DROP TABLE IF EXISTS books;
```

- . before wir die Tabelle löschen, sollen wir erst einen Datenbank auswählen.
- 

#### 5. wähle einen Datenbank mit dem Name der Datenbank:

```
USE db_name;
```

---

## 6. füge eine neue Linie ( Zeile) in der Tabelle:

### 6.1 neue Zeile für jede Spalte:

#### طريقة استخدامه

في حال لم تكن ستمرر قيم لكل الحقول الموجودة في السطر، يجب أن تحدد اسم كل عمود ستمرر قيمة له.

```
INSERT INTO table_name (column1, column2, column3, ...)  
VALUES (value1, value2, value3, ...);
```



#### Erklärung:

- مكان الكلمة `table_name` نضع اسم الجدول الذي سنضيف فيه السطر.
- بداخل القوسين اللذين نضعهما بعد اسم الجدول نحدد أسماء الأعمدة التي نريد أن نضع فيها القيم.
- بداخل القوسين اللذين نضعهما بعد كلمة `VALUES` نمرر القيم التي سيتم وضعها في الأعمدة بنفس الترتيب.

### 6.2 neue Zeile für alle Spalten:

في حال كنت تنوي إدخال قيم لكل الحقول الموجودة في السطر، عندها لا داعي لذكر أسماء الأعمدة و بالتالي يمكن كتابة الإستعلام كالتالي.

```
INSERT INTO table_name  
VALUES (value1, value2, value3, ...);
```



## 7. Automatische Nummerierung:

أي حقل يتم إضافته في الجدول بهدف أن يتم استخدامه للتمييز بين الأسطر يتم تعريفه كحقل تلقائي التعداد ( `Auto Increment` ) و كفتاح أساسي ( `Primary Key` ).

### الحصول على ترقيم تلقائي

طريقة إنشاء عمود في الجدول يتم إعطاء ترقيم تلقائي لكل سطر فيه تختلف من قاعدة بيانات لأخرى كما يلي:

- في قواعد بيانات `MySQL` نستخدم `AUTO_INCREMENT`.

### 7.1 Beispiel:

#### إنشاء مفتاح يترقم تلقائياً

بهدف أن يتم جعل قيمة الحقل تزداد بشكل تلقائي في كل سطر و ضمان أن تكون قيمته موحدة سنقوم دائماً بتعريفه على النحو التالي.

```
column_name INT NOT NULL PRIMARY KEY AUTO_INCREMENT
```



- مكان الكلمة `column_name` نضع اسم العمود الذي سيتم إنشاؤه في الجدول.
- `INT` تعني أن العمود سيوضع فيه أعداد صحيحة.
- `NOT NULL` تعني أن العمود لا يمكن أن يكون فارغ و هذا الأمر منطقي لأنه سيتم وضع عدد بشكل تلقائي لكل سطر يتم إضافته.
- `PRIMARY KEY` تعني أن هذا الحقل لا يمكنه أن يحتوي على قيم مكررة و لهذا يعتبر المفتاح الرئيسي للتمييز و الوصول لأي سطر موجود في الجدول.
- `AUTO_INCREMENT` تعني أن القيم في العمود ستزداد بشكل تلقائي في كل سطر جديد.

## 8. Abrufen von Daten aus der Datenbank:

### طريقة استخدامه

عند جلب البيانات من جدول يمكنك تحديد إسم كل عمود تريد جلب البيانات منه كما يلي.

```
SELECT column1, column2, ...  
FROM table_name;
```



أيضاً، تستطيع استخدام الرمز \* لتحديد جميع أعمدة الجدول بشكل تلقائي كما يلي.

```
SELECT * FROM table_name;
```



### 8.1 Beispiel:

```
SELECT * FROM employees;
```



النتيجة

id	first_name	last_name	is_married	salary	birthday
1	Ahmad	Alhazem	0	800.00	1990-10-23
2	Rami	Algharib	0	750.00	1992-05-08
3	Said	Alnaja	1	900.00	1988-07-12
4	Noura	Jamali	0	845.00	1994-12-04
5	Amani	Sarrouf	1	1000.00	1991-10-27
6	Mhamad	Harmush	0	700.00	1993-11-12
7	Ahmad	Asaadi	1	850.00	1995-03-10
8	Mostafa	Shawki	1	920.00	1990-02-14
9	Jana	Alkassem	0	780.00	1991-08-12
10	Houssam	Zahrani	1	880.00	1985-09-20

### 8.2 Spaltennamen ändern, wenn sie aus der Tabelle abgerufen werden: Mit (AS)

Beispiel:

original:

first_name	last_name	salary
------------	-----------	--------

gändert:

مثال

```
SELECT first_name AS 'First Name', last_name AS 'Last Name', salary AS 'Salary'  
FROM employees;
```



النتيجة

First Name	Last Name	Salary
------------	-----------	--------

