MySQL Einführung

* Eine der weltweit meistens verbreitende relationale Datenbankverwaltungssysteme
* Open-Source Software
* Aktuellste Version 5.7.21 (15.Januar 2018)
* Verfügbar für mehrere Betriebssysteme (Windows, macOS, Linux, Unix, Oracle Solaris, FreeBSD)
* Name MySQL kommt von My (Tochter des Mysql-AB Gründers Michiael Widenius) und SQL (Strucutured Query Language)
* MySQL Delphin Logo leitet sich Sakila dem Namen einer Stadt in Tansanias
* MySQL wird heutzutage durch MariaDB abgelöst
* MariaDB ist zu Mysql Datenbanken kompatibel, welches Umstieg sehr vereinfacht

Open Source Software

* Open Source bedeutet, dass es für jeden möglich ist die Software zu nutzen und diese frei zu verändern/verbessern
* Jeder kann sich die Software gratis vom Internet herunterladen

Entwicklung

* 1994 entwickelt von MySQL AB (schwedisches Softwareunternehmen)
* 2008 wurde MySQL von Sun Microsystems übernommen (kaliforniescher Computer und Software Enwickler
* 2010 wurde Sun Microsystems dann von Oracle, einem Weltführer in Softwareherstellung (entwickelt Java laufend weiter, Netbeans, Eclipse…)

Verwendung MySQL

* MySQL wird bevorzugt für die Datenspeicherung von Webservices eingesetzt
* Häufig werden je nach Größe des Webservers eine Vielzahl von Datenbanken genutzt und viele Nutzer gleichzeitig können auf diese zugreifen  
  Bekannte Bsp.: Youtube, Facebook, Twitter
* MySQL wird häufig mit dem Webserver Apache und der Skriptsprache PHP eingesetzt
* Wird oft auch als eingebettetes Datenbanksystem eingesetzt

Eingebettetes Datenbanksystem

Ein eingebettetes Datenbanksystem ist ein in der Anwendungssoftware eingebettetes Datenbanksystem, das nach außen nicht sichtbar in Erscheinung tritt und speziell für die Anwendung angepasst ist. Das eingebettete Datenbanksystem kann nicht von Fremdsystem zur Datenspeicherung verwendet werden

Relationale Datenbank

* Weitest verbreitetes Datenbankmodell
* Konzept: Basiert auf der Speicherung von Informationen in verschiedenen Tabellen, welche über eine Beziehung (Relation) miteinander verknüpft sind
* Datenbankmanagesystem von rationalen Datenbanken heißt RDBMS (Relational Database Management System)
* Mit RDBMS lassen sich rationale Datenbanken erstellen, pflegen und administrieren

MySQL Aufbau

* Datenbank 🡪 Ein Ort um Daten zu speichern
* Datenbank besteht aus mehreren Tabellen
* Tabelle ist Teil der Datenbank, welcher die Daten speichert
* Tabellen wiederum bestehen aus Zeilen und Spalten
* In den Spalten werden die Attribute dargestellt
* Attribute müssen immer einen Datentyp besitzen (Varchar, Integer, Date,..)
* Attribute können neben dem Datentyp auch Contraints besitzen
* Constraints werden verwendet, um Regeln für Dateneinträge in einer Tabelle zu definieren
* In den Zeilen werden die Dateneinträge mit dem im Attribut definierten Datentyp hineingespeichert (Zeile für Attribut Name „Martin Eller“)
* Tabellen werden mit sogenannten Schlüsseln(PRIMARY und FOREIGN KEYS) miteinander verbunden

Verwendung von Constraints

* Constraints können verwendet werden, um eine erhöhte Genauigkeit und Zuverlässigkeit der Daten in einer Tabelle zu erhalten
* Bei Verstoß wird ein Fehler ausgegeben
* Constraints können auf ganze Tabellen oder Spalten angewendet werden
* Wichtige Constraints sind:

1. NOT NULL 🡪 stellt sicher, dass Wert in der Spalte keinen Nullwert haben darf
2. UNIQUE 🡪 stellt sicher, dass alle Werte in einer Spalte unterschiedlich sind
3. PRIMARY KEY 🡪 identifiziert Dateneinträge in einer Zeile als eindeutig  
   (Kombination aus NOT NULL und Unique)
4. FOREIGN KEY 🡪 identifiziert Dateneinträge in einer anderen (fremden) Tabelle als eindeutig
5. DEFAULT 🡪 legt einen Standartwert für eine Spalte fest, falls kein Dateneintrag gemacht wurde

SQL

* SQL oder auch Structured query language ist die Datenbanksprache
* SQL ermöglicht es mit speziellen SQL Statements( in DDL 🡪 Data Definition Language definiert) an die Datenbank das Erstellen, Löschen, Verändern von Tabellen und das Abfragen der in denen sich befindlichen Daten
* Wichtigsten Befehle in SQL sind CREATE, ALTER, DELETE, DROP, UPDATE, INSERT, SELECT, SHOW

Create Statements

* Für das Erstellen einer Datenbank 🡪 create database <name>;
* Für das Erstellen einer Tabelle 🡪 create table <name> (<column1> datatype,  
   <column2> datatype, …);

Drop Statement

* Um eine bestehende Datenbank mit Inhalt zu löschen 🡪 drop database <name>;
* Um eine bestehende Tabelle mit Inhalt zu löschen 🡪 drop table <name>;

Alter Statement

* Um in einer bestehenden Tabelle Spalten hinzuzufügen 🡪 alter table <name>   
   add <column> datatype;
* Um aus einer bestehenden Tabelle Spalten zu löschen 🡪 alter table <name>  
   drop column <column>;
* Um den Datentyp bestimmter Spalten zu ändern 🡪 alter table <name>   
   modify column <column> datatype;

Insert Statement

* Um Elemente in Tabellen hinzuzufügen 🡪 insert into <name>   
   (column1 , column2, column3, …)

Values (value1, value2, value3,….);

Update Statement

* Um bestimmten Tabelleneintrag zu ändern 🡪 update <table> set <cloumn1>=value  
   where <column>=othervalue;

Delete Statement

* Um einen bestimmten Tabelleneintrag zu löschen 🡪 delete from <table>   
   where <column>=value;
* Um alle Tabelleneinträge in einer Tabelle zu löschen 🡪 delete from <table>;

Select Statement

* Um alle Tabelleneinträge einer Tabelle anzeigen zu können   
  🡪 Select \* from <table>;

Show Statement

* Um Spalten von einer Tabelle anzuzeigen 🡪 show columns from <table>;
* Um bestehende Datenbanken anzuzeigen 🡪 show databases;
* Um bestehende Tabellen in einer Datenbank anzuzeigen 🡪 show tables;