



合肥大學
HEFEI UNIVERSITY



Datenbanken

5. Software und Literatur

Thomas Weise (汤卫思)
tweise@hfuu.edu.cn

Institute of Applied Optimization (IAO)
School of Artificial Intelligence and Big Data
Hefei University
Hefei, Anhui, China

应用优化研究所
人工智能与大数据学院
合肥大学
中国安徽省合肥市

Databases



Dies ist ein Kurs über Datenbanken an der Universität Hefei (合肥大学).

Die Webseite mit dem Lehrmaterial dieses Kurses ist <https://thomasweise.github.io/databases> (siehe auch den QR-Kode unten rechts). Dort können Sie das Kursbuch (in Englisch) und diese Slides finden. Das Repository mit den Beispielen finden Sie unter <https://github.com/thomasWeise/databasesCode>.



Outline



1. Einleitung
2. Open Source Relationale Datenbankmanagementsysteme
3. Proprietäre Datenbankmanagementsysteme
4. Software in diesem Kurs
5. Literatur
6. Zusammenfassung





Einleitung



Einleitung

- Heute gibt es eine große Auswahl an relationalen Datenbankmanagementsystemen.



Einleitung



- Heute gibt es eine große Auswahl an relationalen Datenbankmanagementsystemen.
- *DB-Engines Ranking of Relational DBMS (June 2025)*²³ listed 166 solche Produkte.

Einleitung

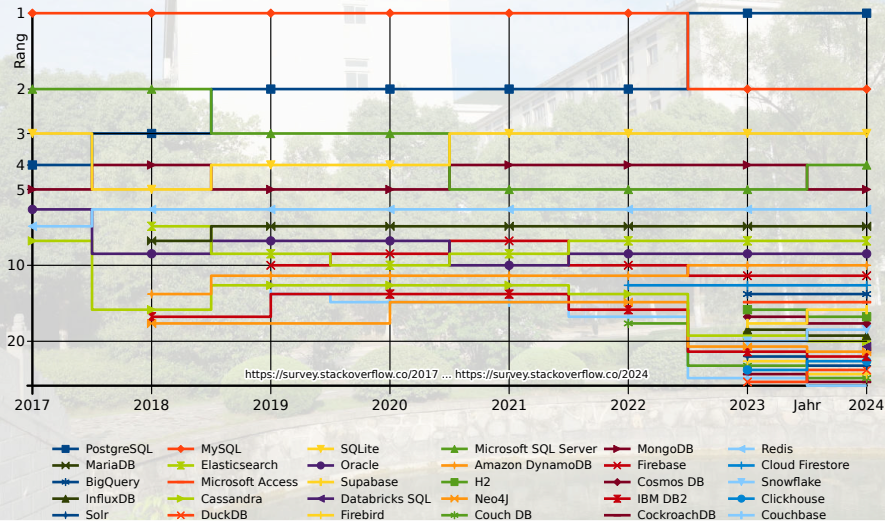


- Heute gibt es eine große Auswahl an relationalen Datenbankmanagementsystemen.
- *DB-Engines Ranking of Relational DBMS (June 2025)*²³ listed 166 solche Produkte.
- Im "Stack Overflow 2024 Developer Survey"⁷³ konnten Programmierer Feedback zu 35 Systemen geben.

Einleitung



- Eine Auswahl der Systeme der jährlichen Developer Surveys von StackOverflow seit 2017.



Open Source vs. Proprietäre Software

- Open Source Software (OSS)



Open Source vs. Proprietäre Software



- Open Source Software (OSS)
 - OSS ist Software, deren Quellcode frei verfügbar ist.

Open Source vs. Proprietäre Software



- Open Source Software (OSS)
 - OSS ist Software, deren Quellcode frei verfügbar ist.
 - Oftmals wird OSS von Freiwilligen entwickelt, manchmal mit finanzieller Unterstützung von Firmen oder Regierungen.

Open Source vs. Proprietäre Software



- Open Source Software (OSS)

- OSS ist Software, deren Quellcode frei verfügbar ist.
- Oftmals wird OSS von Freiwilligen entwickelt, manchmal mit finanzieller Unterstützung von Firmen oder Regierungen.
- OSS kostet kein Geld.

Open Source vs. Proprietäre Software



- Open Source Software (OSS)

- OSS ist Software, deren Quellcode frei verfügbar ist.
- Oftmals wird OSS von Freiwilligen entwickelt, manchmal mit finanzieller Unterstützung von Firmen oder Regierungen.
- OSS kostet kein Geld.
- OSS wird oft auf kollaborativen Plattformen wie GitHub gehostet und ihre Entwicklung wird durch VCSes wie Git gemanaged.

Open Source vs. Proprietäre Software



- Open Source Software (OSS)

- OSS ist Software, deren Quellcode frei verfügbar ist.
- Oftmals wird OSS von Freiwilligen entwickelt, manchmal mit finanzieller Unterstützung von Firmen oder Regierungen.
- OSS kostet kein Geld.
- OSS wird oft auf kollaborativen Plattformen wie GitHub gehostet und ihre Entwicklung wird durch VCSes wie Git gemanaged.
- Nach Hoffmann, Nagle und Zhou war der Wert von OSS im Jahr 2024 mehr als 8 billionen dollar⁴¹.

Open Source vs. Proprietäre Software



- Open Source Software (OSS)

- OSS ist Software, deren Quellcode frei verfügbar ist.
- Oftmals wird OSS von Freiwilligen entwickelt, manchmal mit finanzieller Unterstützung von Firmen oder Regierungen.
- OSS kostet kein Geld.
- OSS wird oft auf kollaborativen Plattformen wie GitHub gehostet und ihre Entwicklung wird durch VCSes wie Git gemanaged.
- Nach Hoffmann, Nagle und Zhou war der Wert von OSS im Jahr 2024 mehr als 8 billionen dollar⁴¹.

- Proprietäre Software

Open Source vs. Proprietäre Software



- Open Source Software (OSS)

- OSS ist Software, deren Quellcode frei verfügbar ist.
- Oftmals wird OSS von Freiwilligen entwickelt, manchmal mit finanzieller Unterstützung von Firmen oder Regierungen.
- OSS kostet kein Geld.
- OSS wird oft auf kollaborativen Plattformen wie GitHub gehostet und ihre Entwicklung wird durch VCSes wie Git gemanaged.
- Nach Hoffmann, Nagle und Zhou war der Wert von OSS im Jahr 2024 mehr als 8 billionen dollar⁴¹.

- Proprietäre Software

- Proprietäre Software ist kommerzielle Software, die von einem Anbieter entwickelt wird und deren Quellcode normalerweise nicht frei zugänglich ist.

Open Source vs. Proprietäre Software



- Open Source Software (OSS)

- OSS ist Software, deren Quellcode frei verfügbar ist.
- Oftmals wird OSS von Freiwilligen entwickelt, manchmal mit finanzieller Unterstützung von Firmen oder Regierungen.
- OSS kostet kein Geld.
- OSS wird oft auf kollaborativen Plattformen wie GitHub gehostet und ihre Entwicklung wird durch VCSes wie Git gemanaged.
- Nach Hoffmann, Nagle und Zhou war der Wert von OSS im Jahr 2024 mehr als 8 billionen dollar⁴¹.

- Proprietäre Software

- Proprietäre Software ist kommerzielle Software, die von einem Anbieter entwickelt wird und deren Quellcode normalerweise nicht frei zugänglich ist.
- Solche Software muss gekauft oder gemietet werden.



Open Source Relationale Datenbankmanagementsysteme



Open Source Relationale Datenbankmanagementsysteme



- Ein großer Teil der relationalen Datenbanken wird von Open Source DBMS gemanaged.

Open Source Relationale Datenbankmanagementsysteme



- Ein großer Teil der relationalen Datenbanken wird von Open Source DBMS gemanaged.
- Diese kosten kein Geld.

Open Source Relationale Datenbankmanagementsysteme



- Ein großer Teil der relationalen Datenbanken wird von Open Source DBMS gemanaged.
- Diese kosten kein Geld.
- Große Gemeinschaften von Entwicklern existieren um sie herum, entwickeln sie gemeinsam, und können Hilfe leisten.

Open Source Relationale Datenbankmanagementsysteme



- Ein großer Teil der relationalen Datenbanken wird von Open Source DBMS gemanaged.
- Diese kosten kein Geld.
- Große Gemeinschaften von Entwicklern existieren um sie herum, entwickeln sie gemeinsam, und können Hilfe leisten.
- Einige dieser Systeme gibt es seit den 1990ern.

Open Source Relationale Datenbankmanagementsysteme



- Ein großer Teil der relationalen Datenbanken wird von Open Source DBMS gemanaged.
- Diese kosten kein Geld.
- Große Gemeinschaften von Entwicklern existieren um sie herum, entwickeln sie gemeinsam, und können Hilfe leisten.
- Einige dieser Systeme gibt es seit den 1990ern.
- In den frühen 2000ern began ihr großer Aufstieg⁵⁹.

Open Source Relationale Datenbankmanagementsysteme



- Ein großer Teil der relationalen Datenbanken wird von Open Source DBMS gemanaged.
- Diese kosten kein Geld.
- Große Gemeinschaften von Entwicklern existieren um sie herum, entwickeln sie gemeinsam, und können Hilfe leisten.
- Einige dieser Systeme gibt es seit den 1990ern.
- In den frühen 2000ern began ihr großer Aufstieg⁵⁹.
- Nun sind sie weitverbreitete und gut getestete Produkte¹⁶.

MySQL



- MySQL ist so ein Open Source DBMS für relationale Datenbanken^{11,26,67,78,87}.

MySQL



- MySQL ist so ein Open Source DBMS für relationale Datenbanken^{11,26,67,78,87}.
- Es wurde ursprünglich von Michael Widenius und David Axmark in der schwedischen Firma MySQL AB entwickelt und kam 1996 heraus¹⁶.

MySQL



- MySQL ist so ein Open Source DBMS für relationale Datenbanken^{11,26,67,78,87}.
- Es wurde ursprünglich von Michael Widenius und David Axmark in der schwedischen Firma MySQL AB entwickelt und kam 1996 heraus¹⁶.
- Es wird ein wichtiger Teil des weitverbreiteten LAMP Stack, also einem Systemsetup für Web-Application der auf dem Linux Betriebssystem, dem Apache webserver, der MySQL Datenbank, und der Server-seitigen Skriptsprache PHP basiert^{14,37}.

MySQL



- MySQL ist so ein Open Source DBMS für relationale Datenbanken^{11,26,67,78,87}.
- Es wurde ursprünglich von Michael Widenius und David Axmark in der schwedischen Firma MySQL AB entwickelt und kam 1996 heraus¹⁶.
- Es wird ein wichtiger Teil des weitverbreiteten LAMP Stack, also einem Systemsetup für Web-Application der auf dem Linux Betriebssystem, dem Apache webserver, der MySQL Datenbank, und der Server-seitigen Skriptsprache PHP basiert^{14,37}.
- Es stand bis ins Jahr 2022 an erster Stelle als beliebtestes DBMS in den Stack Overflow Developer Surveys⁷³.

MySQL



- MySQL ist so ein Open Source DBMS für relationale Datenbanken^{11,26,67,78,87}.
- Es wurde ursprünglich von Michael Widenius und David Axmark in der schwedischen Forma MySQL AB entwickelt und kam 1996 heraus¹⁶.
- Es wird ein wichtiger Teil des weitverbreiteten LAMP Stack, also einem Systemsetup für Web-Application der auf dem Linux Betriebssystem, dem Apache webserver, der MySQL Datenbank, und der Server-seitigen Skriptsprache PHP basiert^{14,37}.
- Es stand bis ins Jahr 2022 an erster Stelle als beliebtestes DBMS in den Stack Overflow Developer Surveys⁷³.
- In dem Survey⁵⁷ des Quellkodes von 371 Java Open Source Projekten auf GitHub wurde MySQL auch am häufigsten als DBMS gefunden.

MySQL



- MySQL ist so ein Open Source DBMS für relationale Datenbanken^{11,26,67,78,87}.
- Es wurde ursprünglich von Michael Widenius und David Axmark in der schwedischen Forma MySQL AB entwickelt und kam 1996 heraus¹⁶.
- Es wird ein wichtiger Teil des weitverbreiteten LAMP Stack, also einem Systemsetup für Web-Application der auf dem Linux Betriebssystem, dem Apache webserver, der MySQL Datenbank, und der Server-seitigen Skriptsprache PHP basiert^{14,37}.
- Es stand bis ins Jahr 2022 an erster Stelle als beliebtestes DBMS in den Stack Overflow Developer Surveys⁷³.
- In dem Survey⁵⁷ des Quellkodes von 371 Java Open Source Projekten auf GitHub wurde MySQL auch am häufigsten als DBMS gefunden.
- MySQL AB wurde im Jahr 2000 von Sun Microsystems gekauft, die wiederum 2010 von Oracle gekauft wurden¹⁶.

MySQL



- MySQL ist so ein Open Source DBMS für relationale Datenbanken^{11,26,67,78,87}.
- Es wurde ursprünglich von Michael Widenius und David Axmark in der schwedischen Forma MySQL AB entwickelt und kam 1996 heraus¹⁶.
- Es wird ein wichtiger Teil des weitverbreiteten LAMP Stack, also einem Systemsetup für Web-Application der auf dem Linux Betriebssystem, dem Apache webserver, der MySQL Datenbank, und der Server-seitigen Skriptsprache PHP basiert^{14,37}.
- Es stand bis ins Jahr 2022 an erster Stelle als beliebtestes DBMS in den Stack Overflow Developer Surveys⁷³.
- In dem Survey⁵⁷ des Quellkodes von 371 Java Open Source Projekten auf GitHub wurde MySQL auch am häufigsten als DBMS gefunden.
- MySQL AB wurde im Jahr 2000 von Sun Microsystems gekauft, die wiederum 2010 von Oracle gekauft wurden¹⁶.
- MySQL blieb jedoch OSS.



MariaDB

The MariaDB logo is under the copyright of its owners.

- Nach dem Aufkauf von Sun durch Oracle gründeten einige der ursprünglichen MySQL-Entwickler, auch Michael Widenius, den MySQL-Fork MariaDB^{4,5,8,26,66}.



MariaDB

The MariaDB logo is under the copyright of its owners.

- Nach dem Aufkauf von Sun durch Oracle gründeten einige der ursprünglichen MySQL-Entwickler, auch Michael Widenius, den MySQL-Fork MariaDB^{4,5,8,26,66}.
- Sie versprachen, dass MariaDB für immer OSS bleiben wird.



MariaDB

The MariaDB logo is under the copyright of its owners.

- Nach dem Aufkauf von Sun durch Oracle gründeten einige der ursprünglichen MySQL-Entwickler, auch Michael Widenius, den MySQL-Fork MariaDB^{4,5,8,26,66}.
- Sie versprachen, dass MariaDB für immer OSS bleiben wird.
- In der Studie⁵⁷ über Datenbanknutzung von Java OSS Programmen war MariaDB schon das fünft-populäre relationale DBMS.

PostgreSQL

- PostgreSQL^{29,54,63,78} ist ein Objekt-relationales DBMS, das auch Konzepte aus der Object-Orientierten Programmierung (OOP) wie Vererbung unter Tabellen unterstützt.



PostgreSQL



- PostgreSQL^{29,54,63,78} ist ein Objekt-relationales DBMS, das auch Konzepte aus der Object-Orientierten Programmierung (OOP) wie Vererbung unter Tabellen unterstützt.
- Es unterstützt auch zusätzliche Datentypen, wie JSON-Objekte und geometrische Daten.



PostgreSQL



- PostgreSQL^{29,54,63,78} ist ein Objekt-relationales DBMS, das auch Konzepte aus der Object-Orientierten Programmierung (OOP) wie Vererbung unter Tabellen unterstützt.
- Es unterstützt auch zusätzliche Datentypen, wie JSON-Objekte und geometrische Daten.
- Es stammt aus dem POSTGRES-Projekt, dem Nachfolger des berühmten INGRES-Projekts an der University of Berkeley in Kalifornien¹⁶.



PostgreSQL



- PostgreSQL^{29,54,63,78} ist ein Objekt-relationales DBMS, das auch Konzepte aus der Object-Orientierten Programmierung (OOP) wie Vererbung unter Tabellen unterstützt.
- Es unterstützt auch zusätzliche Datentypen, wie JSON-Objekte und geometrische Daten.
- Es stammt aus dem POSTGRES-Projekt, dem Nachfolger des berühmten INGRES-Projekts an der University of Berkeley in Kalifornien¹⁶.
- Es ist vielleicht das OSS DBMS mit den meisten Features.



PostgreSQL



- PostgreSQL^{29,54,63,78} ist ein Objekt-relationales DBMS, das auch Konzepte aus der Object-Orientierten Programmierung (OOP) wie Vererbung unter Tabellen unterstützt.
- Es unterstützt auch zusätzliche Datentypen, wie JSON-Objekte und geometrische Daten.
- Es stammt aus dem POSTGRES-Projekt, dem Nachfolger des berühmten INGRES-Projekts an der University of Berkeley in Kalifornien¹⁶.
- Es ist vielleicht das OSS DBMS mit den meisten Features.
- Im "Stack Overflow 2024 Developer Survey" war es das populärste DBMS⁷³.



PostgreSQL



- PostgreSQL^{29,54,63,78} ist ein Objekt-relationales DBMS, das auch Konzepte aus der Object-Orientierten Programmierung (OOP) wie Vererbung unter Tabellen unterstützt.
- Es unterstützt auch zusätzliche Datentypen, wie JSON-Objekte und geometrische Daten.
- Es stammt aus dem POSTGRES-Projekt, dem Nachfolger des berühmten INGRES-Projekts an der University of Berkeley in Kalifornien¹⁶.
- Es ist vielleicht das OSS DBMS mit den meisten Features.
- Im "Stack Overflow 2024 Developer Survey" war es das populärste DBMS⁷³.
- Im Open Source Quellcode-Survey⁵⁷ war es auf Rang 2.



PostgreSQL

- die Structured Query Language (SQL)-Datenbank mit der weitesten Verbreitung ist jedoch SQLite^{16,30,39,88}.



The SQLite logo is under the copyright of its owners.

PostgreSQL



- die SQL-Datenbank mit der weitesten Verbreitung ist jedoch SQLite^{16,30,39,88}.
- Anders als die vorher genannten System unterstützt es *nicht* die client-server architecture.



The SQLite logo is under the copyright of its owners.

PostgreSQL



- die SQL-Datenbank mit der weitesten Verbreitung ist jedoch SQLite^{16,30,39,88}.
- Anders als die vorher genannten System unterstützt es *nicht* die client-server architecture.
- Stattdessen wird es als Bibliothek direkt in den Prozess geladen, der es nutzt.



The SQLite logo is under the copyright of its owners.

PostgreSQL



- die SQL-Datenbank mit der weitesten Verbreitung ist jedoch SQLite^{16,30,39,88}.
- Anders als die vorher genannten System unterstützt es *nicht* die client-server architecture.
- Stattdessen wird es als Bibliothek direkt in den Prozess geladen, der es nutzt.
- Heute ist es auf fast jedem Computer, Smartphone, Fernseher, und Automobil installiert und Teil von fast jedem Web Browser.



The SQLite logo is under the copyright of its owners.

PostgreSQL



- die SQL-Datenbank mit der weitesten Verbreitung ist jedoch SQLite^{16,30,39,88}.
- Anders als die vorher genannten System unterstützt es *nicht* die client-server architecture.
- Stattdessen wird es als Bibliothek direkt in den Prozess geladen, der es nutzt.
- Heute ist es auf fast jedem Computer, Smartphone, Fernseher, und Automobil installiert und Teil von fast jedem Web Browser.
- Es wurde ursprünglich im Jahr 2000 veröffentlicht und sein Kernentwickler Richard Hipp hat dafür den SIGMOD Systems Award 2017 bekommen¹⁶.



The SQLite logo is under the copyright of its owners.

LibreOffice Base



- Ein DBMS welches mit einzelnen Dateien arbeitet ist auch in der Open Source Bürosoftware LibreOffice^{31,51,70} implementiert: Als LibreOffice Base^{28,70}.



LibreOffice®

The LibreOffice logo is under the copyright of its owners.

LibreOffice Base



- Ein DBMS welches mit einzelnen Dateien arbeitet ist auch in der Open Source Bürosoftware LibreOffice^{31,51,70} implementiert: Als LibreOffice Base^{28,70}.
- LibreOffice Base ist allerdings mehr als ein DBMS.



LibreOffice®

The LibreOffice logo is under the copyright of its owners.

LibreOffice Base



- Ein DBMS welches mit einzelnen Dateien arbeitet ist auch in der Open Source Bürosoftware LibreOffice^{31,51,70} implementiert: Als LibreOffice Base^{28,70}.
- LibreOffice Base ist allerdings mehr als ein DBMS.
- Es bietet eine vielseitige graphische Benutzeroberfläche mit dem man sich auch auf Datenbanken wie MySQL, MariaDB und PostgreSQL verbinden kann.



LibreOffice®

The LibreOffice logo is under the copyright of its owners.

LibreOffice Base



- Ein DBMS welches mit einzelnen Dateien arbeitet ist auch in der Open Source Bürosoftware LibreOffice^{31,51,70} implementiert: Als LibreOffice Base^{28,70}.
- LibreOffice Base ist allerdings mehr als ein DBMS.
- Es bietet eine vielseitige graphische Benutzeroberfläche mit dem man sich auch auf Datenbanken wie MySQL, MariaDB und PostgreSQL verbinden kann.
- Mit der Benutzeroberfläche kann man Tabellen, Sichte, Anfragen, Formulare, und Berichte für die Datenbanken erstellen.



LibreOffice®

The LibreOffice logo is under the copyright of its owners.

LibreOffice Base



- Ein DBMS welches mit einzelnen Dateien arbeitet ist auch in der Open Source Bürosoftware LibreOffice^{31,51,70} implementiert: Als LibreOffice Base^{28,70}.
- LibreOffice Base ist allerdings mehr als ein DBMS.
- Es bietet eine vielseitige graphische Benutzeroberfläche mit dem man sich auch auf Datenbanken wie MySQL, MariaDB und PostgreSQL verbinden kann.
- Mit der Benutzeroberfläche kann man Tabellen, Sichte, Anfragen, Formulare, und Berichte für die Datenbanken erstellen.
- LibreOffice Base ist eine freie Alternative zu dem kommerziellen Produkt Microsoft Access^{9,19,81}.



LibreOffice®

The LibreOffice logo is under the copyright of its owners.



Proprietäre Datenbankmanagementsysteme



- Die erste kommerzielle SQL-Datenbank war die Oracle Database^{16,55}.

Oracle Database



- Die erste kommerzielle SQL-Datenbank war die Oracle Database^{16,55}.
- Sie ist noch heute eine der erfolgreichsten und weitverbreiteten kommerziellen Datenbanken, mit vielen hochentwickelten Features^{10,47}.

- Die erste kommerzielle SQL-Datenbank war die Oracle Database^{16,55}.
- Sie ist noch heute eine der erfolgreichsten und weitverbreiteten kommerziellen Datenbanken, mit vielen hochentwickelten Features^{10,47}.
- Im Juni 2025 war sie das populärste DMBS überhaupt im Survey²³.

- Die erste kommerzielle SQL-Datenbank war die Oracle Database^{16,55}.
- Sie ist noch heute eine der erfolgreichsten und weitverbreiteten kommerziellen Datenbanken, mit vielen hochentwickelten Features^{10,47}.
- Im Juni 2025 war sie das populärste DMBS überhaupt im Survey²³.
- In dem Java Quellkodesurvey⁵⁷ war sie auf Rang 4.

- Die erste kommerzielle SQL-Datenbank war die Oracle Database^{16,55}.
- Sie ist noch heute eine der erfolgreichsten und weitverbreiteten kommerziellen Datenbanken, mit vielen hochentwickelten Features^{10,47}.
- Im Juni 2025 war sie das populärste DMBS überhaupt im Survey²³.
- In dem Java Quellkodesurvey⁵⁷ war sie auf Rang 4.
- Die Migration von Oracle Database nach PostgreSQL ist in⁴⁸ diskutiert.

Microsoft SQL Server



- Auf Rang 3 im Survey²³ stand der Microsoft SQL Server^{1,60,86}.

Microsoft SQL Server



- Auf Rang 3 im Survey²³ stand der Microsoft SQL Server^{1,60,86}.
- Dieses DBMS hat seine Wurzeln im Jahr 1988, als Microsoft, Ashton-Tate, und Sybase zusammen eine Variante des Sybase SQL-Server für das Betriebssystem IBM OS/2 entwickelten⁷⁹.

Microsoft SQL Server



- Auf Rang 3 im Survey²³ stand der Microsoft SQL Server^{1,60,86}.
- Dieses DBMS hat seine Wurzeln im Jahr 1988, als Microsoft, Ashton-Tate, und Sybase zusammen eine Variante des Sybase SQL-Server für das Betriebssystem IBM OS/2 entwickelten⁷⁹.
- Die erste Version dieses DBMS kam dann 1989 heraus.

Microsoft SQL Server



- Auf Rang 3 im Survey²³ stand der Microsoft SQL Server^{1,60,86}.
- Dieses DBMS hat seine Wurzeln im Jahr 1988, als Microsoft, Ashton-Tate, und Sybase zusammen eine Variante des Sybase SQL-Server für das Betriebssystem IBM OS/2 entwickelten⁷⁹.
- Die erste Version dieses DBMS kam dann 1989 heraus.
- Spätere Versionen unterstützten Microsoft Windows NT.

Microsoft SQL Server



- Auf Rang 3 im Survey²³ stand der Microsoft SQL Server^{1,60,86}.
- Dieses DBMS hat seine Wurzeln im Jahr 1988, als Microsoft, Ashton-Tate, und Sybase zusammen eine Variante des Sybase SQL-Server für das Betriebssystem IBM OS/2 entwickelten⁷⁹.
- Die erste Version dieses DBMS kam dann 1989 heraus.
- Spätere Versionen unterstützten Microsoft Windows NT.
- Heute läuft der Microsoft SQL Server auch auf Linux (genau wie die Oracle Database).

Microsoft Access



- Mit Microsoft Access⁴⁹ bietet Microsoft auch ein DBMS an, das hauptsächlich für einzelne Benutzer auf einem einzelnen Computer gedacht ist.

Microsoft Access



- Mit Microsoft Access⁴⁹ bietet Microsoft auch ein DBMS an, das hauptsächlich für einzelne Benutzer auf einem einzelnen Computer gedacht ist.
- Es ist ein unglaublich nützliches und bequemes Werkzeug, das die Features eines relationen DBMS mit Werkzeugen zum Entwicklen von Formularen und Berichten vereint^{9,19,52,81}.

Microsoft Access



- Mit Microsoft Access⁴⁹ bietet Microsoft auch ein DBMS an, das hauptsächlich für einzelne Benutzer auf einem einzelnen Computer gedacht ist.
- Es ist ein unglaublich nützliches und bequemes Werkzeug, das die Features eines relationen DBMS mit Werkzeugen zum Entwicklen von Formularen und Berichten vereint^{9,19,52,81}.
- Es war sicherlich die Vorlage für LibreOffice Base.

- Mit Microsoft Access⁴⁹ bietet Microsoft auch ein DBMS an, das hauptsächlich für einzelne Benutzer auf einem einzelnen Computer gedacht ist.
- Es ist ein unglaublich nützliches und bequemes Werkzeug, das die Features eines relationen DBMS mit Werkzeugen zum Entwicklen von Formularen und Berichten vereint^{9,19,52,81}.
- Es war sicherlich die Vorlage für LibreOffice Base.
- Die meisten Dinge, die LibreOffice Base kann, kann man auch mit Microsoft Access machen, nur oftmals bequemer ... aber LibreOffice Base ist kostenfrei...

IBM DB2



- IBM's DB2^{7,18} ist ein weiteres frühes relationales DBMS³⁴.



IBM DB2



- IBM's DB2^{7,18} ist ein weiteres frühes relationales DBMS³⁴.
- Es entstand als Software für mächtige Mainframe-Computer, die IBM selbst hergestellt hat.



IBM DB2



- IBM's DB2^{7,18} ist ein weiteres frühes relationales DBMS³⁴.
- Es entstand als Software für mächtige Mainframe-Computer, die IBM selbst hergestellt hat.
- Mainframes sind Computer mit sehr viel Rechenleistung, die oft zentral in großen Organisationen oder Firmen genutzt werden.

IBM DB2



- IBM's DB2^{7,18} ist ein weiteres frühes relationales DBMS³⁴.
- Es entstand als Software für mächtige Mainframe-Computer, die IBM selbst hergestellt hat.
- Mainframes sind Computer mit sehr viel Rechenleistung, die oft zentral in großen Organisationen oder Firmen genutzt werden.
- Während Oracle also Datenbanken für normale Computer hergestellt hat, hat sich IBM auf starke Server konzentriert.

IBM DB2



- IBM's DB2^{7,18} ist ein weiteres frühes relationales DBMS³⁴.
- Es entstand als Software für mächtige Mainframe-Computer, die IBM selbst hergestellt hat.
- Mainframes sind Computer mit sehr viel Rechenleistung, die oft zentral in großen Organisationen oder Firmen genutzt werden.
- Während Oracle also Datenbanken für normale Computer hergestellt hat, hat sich IBM auf starke Server konzentriert.
- 1989 hatte die Hälfte aller Mainframe-Kunden DB2 installiert³⁴.

- IBM's DB2^{7,18} ist ein weiteres frühes relationales DBMS³⁴.
- Es entstand als Software für mächtige Mainframe-Computer, die IBM selbst hergestellt hat.
- Mainframes sind Computer mit sehr viel Rechenleistung, die oft zentral in großen Organisationen oder Firmen genutzt werden.
- Während Oracle also Datenbanken für normale Computer hergestellt hat, hat sich IBM auf starke Server konzentriert.
- 1989 hatte die Hälfte aller Mainframe-Kunden DB2 installiert³⁴.
- Die DB2 ist auf Rang 6 im *DB-Engines Ranking of Relational DBMS (June 2025)*²³ und auf Rang 23 im "Stack Overflow 2024 Developer Survey"⁷³.



Software in diesem Kurs



Dieser Kurs



- Das Ziel dieses Kurses ist eine *praktische* Einführung in Datenbanken.

Dieser Kurs



- Das Ziel dieses Kurses ist eine *praktische* Einführung in Datenbanken.
- Daher werden wir mit praktischen Beispielen arbeiten und (mehr oder weniger) realistische Datenbanken auf echten Datenbankmanagementsystemen erstellen.

Dieser Kurs



- Das Ziel dieses Kurses ist eine *praktische* Einführung in Datenbanken.
- Daher werden wir mit praktischen Beispielen arbeiten und (mehr oder weniger) realistische Datenbanken auf echten Datenbankmanagementsystemen erstellen.
- Das bedeutet, dass wir verschiedene interessante Themen behandeln werden.

Dieser Kurs



- Das Ziel dieses Kurses ist eine *praktische* Einführung in Datenbanken.
- Daher werden wir mit praktischen Beispielen arbeiten und (mehr oder weniger) realistische Datenbanken auf echten Datenbankmanagementsystemen erstellen.
- Das bedeutet, dass wir verschiedene interessante Themen behandeln werden.
- Und in allen Themen werden wir echte praktische Erfahrung sammeln.

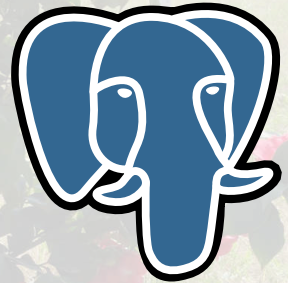
Dieser Kurs



- Das Ziel dieses Kurses ist eine *praktische* Einführung in Datenbanken.
- Daher werden wir mit praktischen Beispielen arbeiten und (mehr oder weniger) realistische Datenbanken auf echten Datenbankmanagementsystemen erstellen.
- Das bedeutet, dass wir verschiedene interessante Themen behandeln werden.
- Und in allen Themen werden wir echte praktische Erfahrung sammeln.
- Und für die meisten Themen werden wir ein passendes Softwarewerkzeug verwenden.

Datenbankmanagementsystem: PostgreSQL

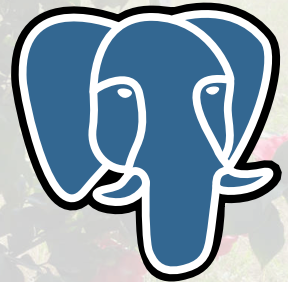
- Wir werden mit einem echten DBMS arbeiten.



The PostgreSQL logo is under the copyright of its owners.

Datenbankmanagementsystem: PostgreSQL

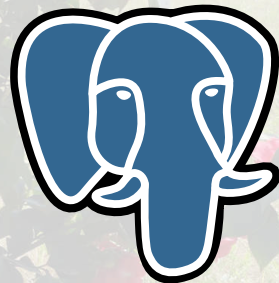
- Wir werden mit einem echten DBMS arbeiten.
- Wir haben uns für PostgreSQL^{29,54,63,78} entschieden.



Datenbankmanagementsystem: PostgreSQL



- Wir werden mit einem echten DBMS arbeiten.
- Wir haben uns für PostgreSQL^{29,54,63,78} entschieden.
- DBMSs sind normalerweise Server an die man SQL-Befehle mit Hilfe eines Clientprogramms vom Terminal aus schicken kann.

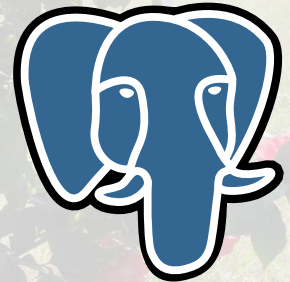


Datenbankmanagementsystem: PostgreSQL



- Wir werden mit einem echten DBMS arbeiten.
- Wir haben uns für PostgreSQL^{29,54,63,78} entschieden.
- DBMSs sind normalerweise Server an die man SQL-Befehle mit Hilfe eines Clientprogramms vom Terminal aus schicken kann.
- Dafür nehmen wir den psql-Klient, der zu PostgreSQL dazu gehört.

```
tweise@weise-laptop: ~  
tweise@weise-laptop: $ psql "postgres://postgres:[REDACTED]@localhost:5432/test"  
psql (16.9 (Ubuntu16.9-0ubuntu0.24.04.1))  
SSL connection (protocol: TLSv1.3, cipher: TLS_AES_256_GCM_SHA384, compression: off)  
Type "help" for help.  
  
test=# CREATE TABLE names (name VARCHAR(100) PRIMARY KEY);  
CREATE TABLE  
test=# INSERT INTO names (name) VALUES ('Bibbo'), ('Bebbo'), ('Bebba');  
INSERT 0 3  
test=# SELECT * FROM names;  
name  
-----  
Bibbo  
Bebbo  
Bebba  
(3rows)  
  
test=# \q  
tweise@weise-laptop: $
```



The PostgreSQL logo is under the copyright of its owners.

GUI: LibreOffice Base

- DBMS werden oft als Server installiert, dem man mit einem Klienten über die Kommandozeile SQL-Kommandos schicken kann.



LibreOffice®

The LibreOffice logo is under the copyright of its owners.

GUI: LibreOffice Base



- DBMS werden oft als Server installiert, dem man mit einem Klienten über die Kommandozeile SQL-Kommandos schicken kann.
- Es gibt aber auch Werkzeuge, die eine GUI anbieten, mit denen man Formulare und Berichte entwerfen kann.



LibreOffice®

The LibreOffice logo is under the copyright of its owners.

GUI: LibreOffice Base



- DBMS werden oft als Server installiert, dem man mit einem Klienten über die Kommandozeile SQL-Kommandos schicken kann.
- Es gibt aber auch Werkzeuge, die eine GUI anbieten, mit denen man Formulare und Berichte entwerfen kann.
- Formulare sind strukturierte Fenster / Dialoge / Masken zur Dateneingabe.



LibreOffice®

The LibreOffice logo is under the copyright of its owners.

GUI: LibreOffice Base



- DBMS werden oft als Server installiert, dem man mit einem Klienten über die Kommandozeile SQL-Kommandos schicken kann.
- Es gibt aber auch Werkzeuge, die eine GUI anbieten, mit denen man Formulare und Berichte entwerfen kann.
- Formulare sind strukturierte Fenster / Dialoge / Masken zur Dateneingabe.
- Berichte sind Dokumente, die automatisch mit den Daten aus einer Datenbank gefüllt werden.



LibreOffice®

The LibreOffice logo is under the copyright of its owners.

GUI: LibreOffice Base



- DBMS werden oft als Server installiert, dem man mit einem Klienten über die Kommandozeile SQL-Kommandos schicken kann.
- Es gibt aber auch Werkzeuge, die eine GUI anbieten, mit denen man Formulare und Berichte entwerfen kann.
- Formulare sind strukturierte Fenster / Dialoge / Masken zur Dateneingabe.
- Berichte sind Dokumente, die automatisch mit den Daten aus einer Datenbank gefüllt werden.
- Wir werden auch so ein Werkzeug ausprobieren: LibreOffice Base^{28,70}.



LibreOffice[®]

The LibreOffice logo is under the copyright of its owners.



The Python programming language logo is under the copyright of its owners.



Copyright © Gabriella Albano and the Psycog team

- Ein DBMS ist normalerweise das Backend der Infrastruktur einer Organisation.

Zugriff über Programmiersprache



The Python programming language logo is under the copyright of its owners.



Copyright © Gabriella Albano and the Psycogp team

- Ein DBMS ist normalerweise das Backend der Infrastruktur einer Organisation.
- Nutzer arbeiten nur sehr selten direkt mit einem DBMS.



pythonTM

The Python programming language logo is under the copyright of its owners.



Copyright © Gabriella Albano and the Psycog team

- Ein DBMS ist normalerweise das Backend der Infrastruktur einer Organisation.
- Nutzer arbeiten nur sehr selten direkt mit einem DBMS.
- Stattdessen werden oft mehrere Applikationen entwickelt, die auf die Datenbank mit Hilfe einer API zugreifen.



The Python programming language logo is under the copyright of its owners.



Copyright © Gabriella Albano and the Psycogp team

- Ein DBMS ist normalerweise das Backend der Infrastruktur einer Organisation.
- Nutzer arbeiten nur sehr selten direkt mit einem DBMS.
- Stattdessen werden oft mehrere Applikationen entwickelt, die auf die Datenbank mit Hilfe einer API zugreifen.
- Wir wollen auch das ausprobieren.



pythonTM

The Python programming language logo is under the copyright of its owners.



Copyright © Gabriella Albano and the Psycogp team

- Ein DBMS ist normalerweise das Backend der Infrastruktur einer Organisation.
- Nutzer arbeiten nur sehr selten direkt mit einem DBMS.
- Stattdessen werden oft mehrere Applikationen entwickelt, die auf die Datenbank mit Hilfe einer API zugreifen.
- Wir wollen auch das ausprobieren.
- Dafür werden wir die Programmiersprache Python^{3,42,45,50,85} verwenden.



The Python programming language logo is under the copyright of its owners.



Copyright © Gabriella Albano and the Psycog team

- Ein DBMS ist normalerweise das Backend der Infrastruktur einer Organisation.
- Nutzer arbeiten nur sehr selten direkt mit einem DBMS.
- Stattdessen werden oft mehrere Applikationen entwickelt, die auf die Datenbank mit Hilfe einer API zugreifen.
- Wir wollen auch das ausprobieren.
- Dafür werden wir die Programmiersprache Python^{3,42,45,50,85} verwenden.
- Dazu gibt es eine Bibliothek für den Zugriff auf PostgreSQL, nämlich psycopg⁸³.

Konzeptuelle Modelle



yEd

The logo of the yEd graph editor. The yEd logo is protected by copyright. yEd is a registered trademark of yWorks GmbH. Unauthorized use, reproduction, or distribution is strictly prohibited.

- Ein wichtiger Schritt im Datenbankentwicklungsprozess ist das Erstellen von abstrakten konzeptuellen Modelle der Problemdomäne.

Konzeptuelle Modelle



yEd

The logo of the yEd graph editor. The yEd logo is protected by copyright. yEd is a registered trademark of yWorks GmbH. Unauthorized use, reproduction, or distribution is strictly prohibited.

- Ein wichtiger Schritt im Datenbankentwicklungsprozess ist das Erstellen von abstrakten konzeptuellen Modelle der Problemdomäne.
- Dazu werden graphische Diagramme, sogenannte entity relationship diagrams (ERDs) gezeichnet, die die Objekte und deren Beziehungen aus der realen Welt darstellen, die wir in der Datenbank modellieren wollen.

Konzeptuelle Modelle



yEd

The logo of the yEd graph editor. The yEd logo is protected by copyright. yEd is a registered trademark of yWorks GmbH. Unauthorized use, reproduction, or distribution is strictly prohibited.

- Ein wichtiger Schritt im Datenbankentwicklungsprozess ist das Erstellen von abstrakten konzeptuellen Modelle der Problemdomäne.
- Dazu werden graphische Diagramme, sogenannte entity relationship diagrams (ERDs) gezeichnet, die die Objekte und deren Beziehungen aus der realen Welt darstellen, die wir in der Datenbank modellieren wollen.
- Diese konzeptuellen Modelle sind unabhängig von der letztendlich verwendeten DBMS-Technologie.

Konzeptuelle Modelle

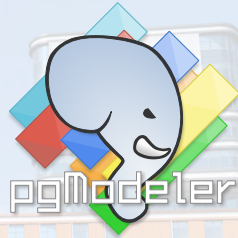


yEd

The logo of the yEd graph editor. The yEd logo is protected by copyright. yEd is a registered trademark of yWorks GmbH. Unauthorized use, reproduction, or distribution is strictly prohibited.

- Ein wichtiger Schritt im Datenbankentwicklungsprozess ist das Erstellen von abstrakten konzeptuellen Modelle der Problemdomäne.
- Dazu werden graphische Diagramme, sogenannte entity relationship diagrams (ERDs) gezeichnet, die die Objekte und deren Beziehungen aus der realen Welt darstellen, die wir in der Datenbank modellieren wollen.
- Diese konzeptuellen Modelle sind unabhängig von der letztendlich verwendeten DBMS-Technologie.
- Als Werkzeug um solche Diagramme zu zeichnen verwenden wir yEd^{69,89}.

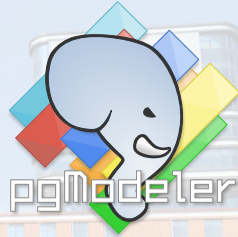
Logische Modelle



The PgModeler logo is under the copyright of Raphael Araújo e Silva.

- Die konzeptuellen Modelle müssen letztendlich auch logische Schemas abgebildet werden, die auf einem spezifischen Datenmodell und einer spezifischen Datenbanktechnologie beruhen.

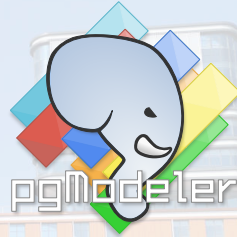
Logische Modelle



The PgModeler logo is under the copyright of Raphael Araújo e Silva.

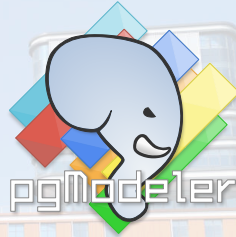
- Die konzeptuellen Modelle müssen letztendlich auch logische Schemas abgebildet werden, die auf einem spezifischen Datenmodell und einer spezifischen Datenbanktechnologie beruhen.
- Die logischen Modelle beschreiben, wie Anwendungen und Nutzer die Daten sehen.

Logische Modelle



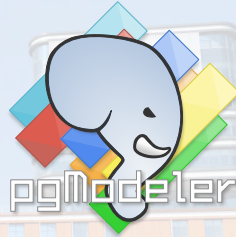
The PgModeler logo is under the copyright of Raphael Araújo e Silva.

- Die konzeptuellen Modelle müssen letztendlich auch logische Schemas abgebildet werden, die auf einem spezifischen Datenmodell und einer spezifischen Datenbanktechnologie beruhen.
- Die logischen Modelle beschreiben, wie Anwendungen und Nutzer die Daten sehen.
- Sie können z.B. in SQL implementiert werden.



The PgModeler logo is under the copyright of Raphael Araújo e Silva.

- Die konzeptuellen Modelle müssen letztendlich auch logische Schemas abgebildet werden, die auf einem spezifischen Datenmodell und einer spezifischen Datenbanktechnologie beruhen.
- Die logischen Modelle beschreiben, wie Anwendungen und Nutzer die Daten sehen.
- Sie können z.B. in SQL implementiert werden.
- Sie können auch visuell mit Werkzeugen modelliert werden, als Diagramme die so ähnlich wie ERDs aussehen, die aber an eine bestimmte Datenbanktechnologie gebunden sind.



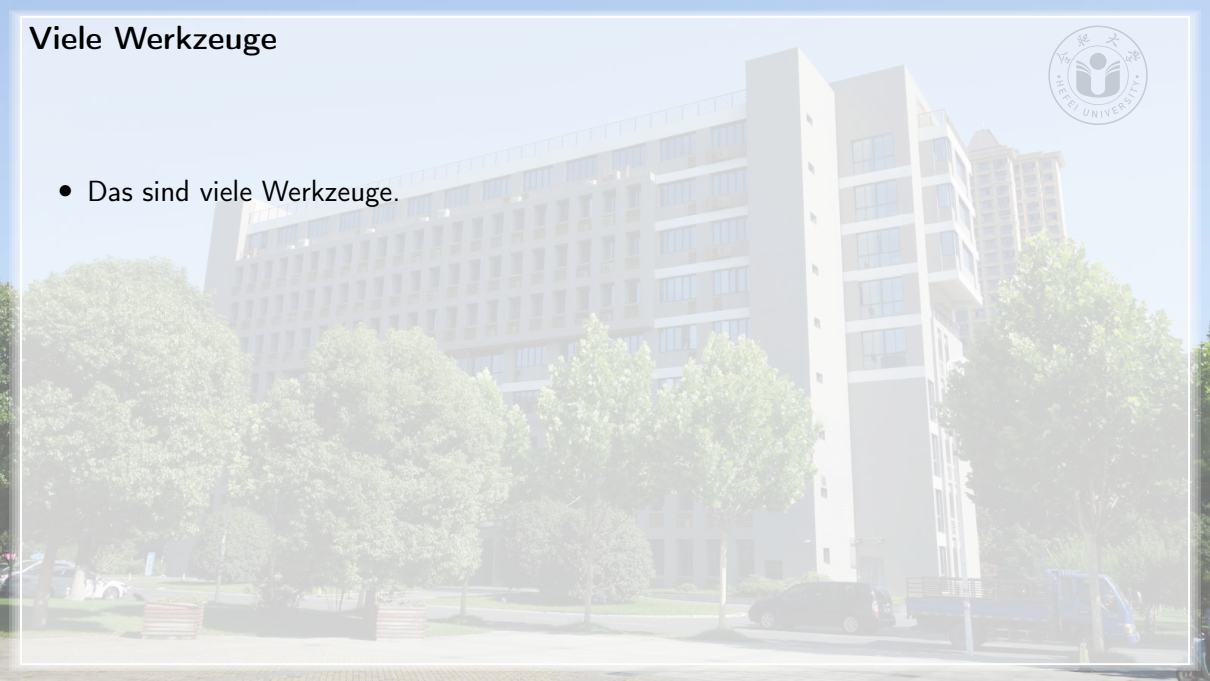
The PgModeler logo is under the copyright of Raphael Araújo e Silva.

- Die konzeptuellen Modelle müssen letztendlich auch logische Schemas abgebildet werden, die auf einem spezifischen Datenmodell und einer spezifischen Datenbanktechnologie beruhen.
- Die logischen Modelle beschreiben, wie Anwendungen und Nutzer die Daten sehen.
- Sie können z.B. in SQL implementiert werden.
- Sie können auch visuell mit Werkzeugen modelliert werden, als Diagramme die so ähnlich wie ERDs aussehen, die aber an eine bestimmte Datenbanktechnologie gebunden sind.
- Wir verwenden den PgModeler², mit dem solche logischen Modelle für PostgreSQL erstellt werden können.

Viele Werkzeuge



- Das sind viele Werkzeuge.



Viele Werkzeuge



- Das sind viele Werkzeuge.

Gute Praxis

Ein(e) Informatiker(in) ist immer in der Lage neue Werkzeuge zu erlernen und tut dies auch gerne.

Viele Werkzeuge



- Das sind viele Werkzeuge.

Gute Praxis

Ein(e) Informatiker(in) ist immer in der Lage neue Werkzeuge zu erlernen und tut dies auch gerne. Ein(e) Informatiker(in) muss in der Lage sein, dutzende oder hunderte verschiedene Softwarewerkzeuge zu nutzen, jeweils für ihre spezifischen Aufgaben.

Viele Werkzeuge



- Das sind viele Werkzeuge.

Gute Praxis

Ein(e) Informatiker(in) ist immer in der Lage neue Werkzeuge zu erlernen und tut dies auch gerne. Ein(e) Informatiker(in) muss in der Lage sein, dutzende oder hunderte verschiedene Softwarewerkzeuge zu nutzen, jeweils für ihre spezifischen Aufgaben. Ein(e) Softwareingenieur(in) ist ein(e) Handwerker(in) und seine/ihre Kenntnis über verschiedene Software ist sein/ihr Werkzeuggürtel.

Spezifische Werkzeuge



- Wir nutzen ein spezifisches DBMS, spezifische Visualisierungstools, eine spezifische Programmiersprache und eine spezifische Bibliothek, um auf das DBMS zuzugreifen.

Spezifische Werkzeuge



- Wir nutzen ein spezifisches DBMS, spezifische Visualisierungstools, eine spezifische Programmiersprache und eine spezifische Bibliothek, um auf das DBMS zuzugreifen.
- Ist das eine gute Idee?

Spezifische Werkzeuge



- Wir nutzen ein spezifisches DBMS, spezifische Visualisierungstools, eine spezifische Programmiersprache und eine spezifische Bibliothek, um auf das DBMS zuzugreifen.
- Ist das eine gute Idee?
- Wenn Sie jemals mit Datenbanken in Ihrem professionellen Leben arbeiten werden, dann wahrscheinlich mit einem anderen DBMS und anderen Werkzeugen.

Spezifische Werkzeuge



- Wir nutzen ein spezifisches DBMS, spezifische Visualisierungstools, eine spezifische Programmiersprache und eine spezifische Bibliothek, um auf das DBMS zuzugreifen.
- Ist das eine gute Idee?
- Wenn Sie jemals mit Datenbanken in Ihrem professionellen Leben arbeiten werden, dann wahrscheinlich mit einem anderen DBMS und anderen Werkzeugen.
- Wird die Kenntnis über die spezifischen Werkzeuge, die wir hier verwenden, Ihnen dann irgendetwas Nutzen?

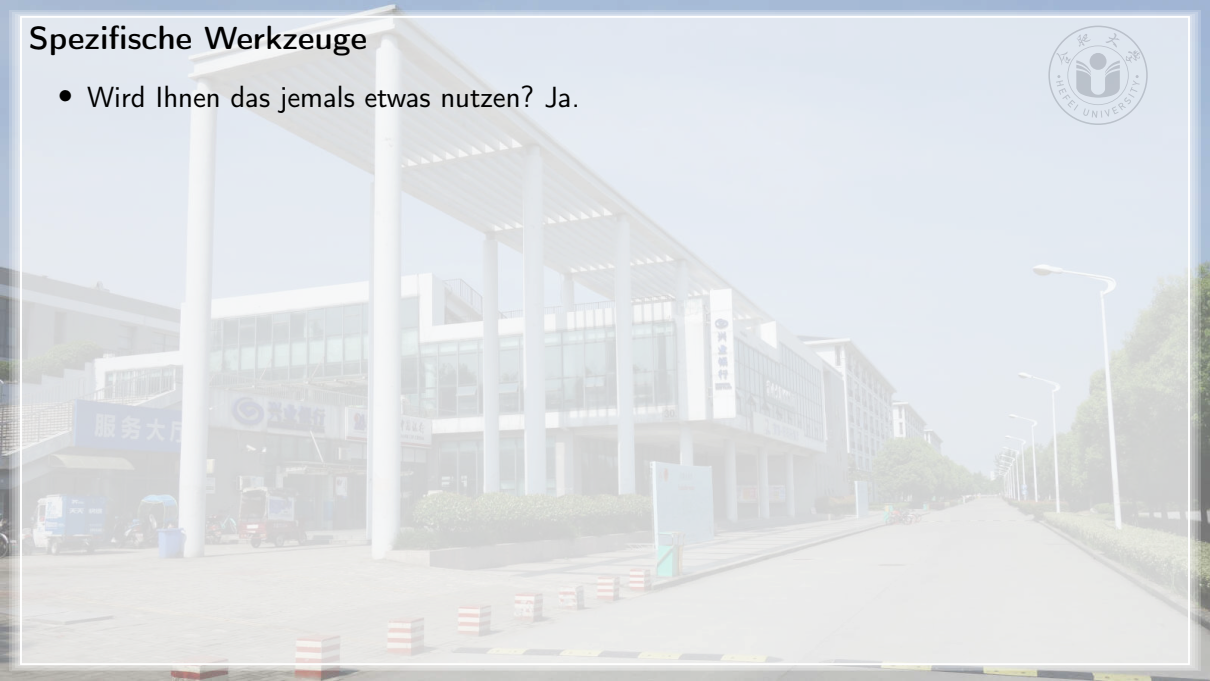
Spezifische Werkzeuge

- Wird Ihnen das jemals etwas nutzen?



Spezifische Werkzeuge

- Wird Ihnen das jemals etwas nutzen? Ja.



Spezifische Werkzeuge



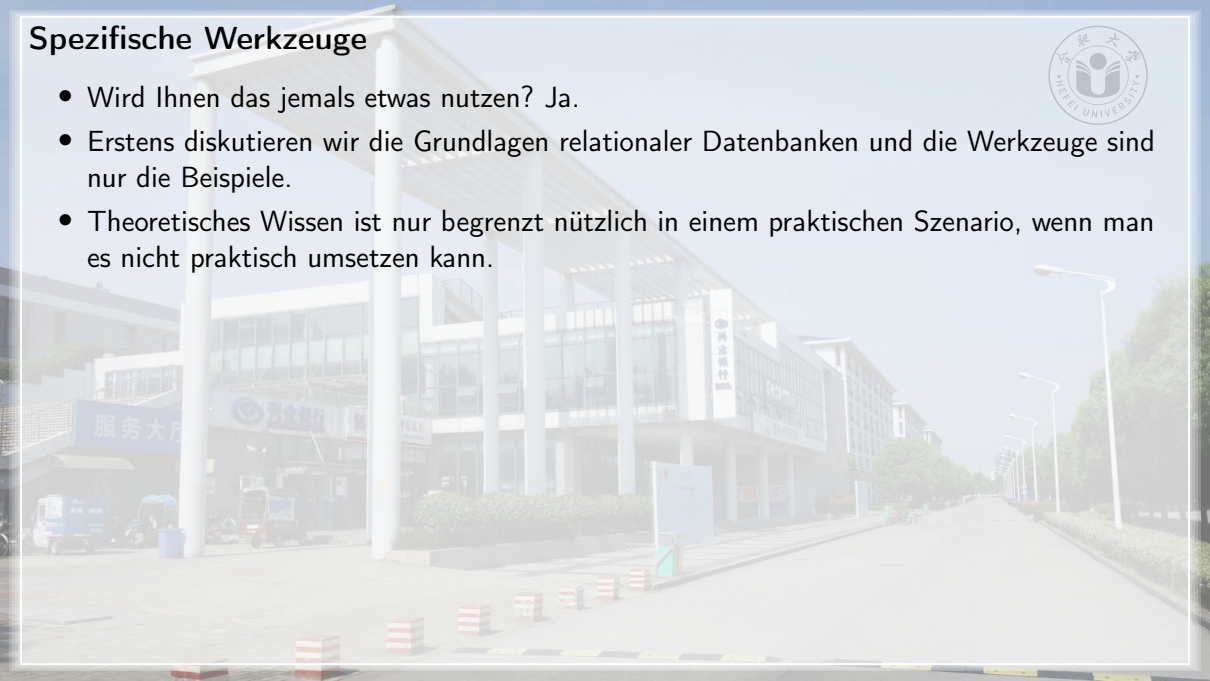
- Wird Ihnen das jemals etwas nutzen? Ja.
- Erstens diskutieren wir die Grundlagen relationaler Datenbanken und die Werkzeuge sind nur die Beispiele.



Spezifische Werkzeuge



- Wird Ihnen das jemals etwas nutzen? Ja.
- Erstens diskutieren wir die Grundlagen relationaler Datenbanken und die Werkzeuge sind nur die Beispiele.
- Theoretisches Wissen ist nur begrenzt nützlich in einem praktischen Szenario, wenn man es nicht praktisch umsetzen kann.



Spezifische Werkzeuge



- Wird Ihnen das jemals etwas nutzen? Ja.
- Erstens diskutieren wir die Grundlagen relationaler Datenbanken und die Werkzeuge sind nur die Beispiele.
- Theoretisches Wissen ist nur begrenzt nützlich in einem praktischen Szenario, wenn man es nicht praktisch umsetzen kann.
- Es hilft gar nichts, die Grundlagen von Datenbanken zu kennen, wenn Sie sich nicht mit einem Terminal zum DBMS verbinden und SQL-Kommandos dorthin senden können, wenn Sie nicht wissen, wie man Fehlermeldungen liest, wenn Sie niemals mit einem echten DBMS gearbeitet haben.

Spezifische Werkzeuge



- Wird Ihnen das jemals etwas nutzen? Ja.
- Erstens diskutieren wir die Grundlagen relationaler Datenbanken und die Werkzeuge sind nur die Beispiele.
- Theoretisches Wissen ist nur begrenzt nützlich in einem praktischen Szenario, wenn man es nicht praktisch umsetzen kann.
- Es hilft gar nichts, die Grundlagen von Datenbanken zu kennen, wenn Sie sich nicht mit einem Terminal zum DBMS verbinden und SQL-Kommandos dorthin senden können, wenn Sie nicht wissen, wie man Fehlermeldungen liest, wenn Sie niemals mit einem echten DBMS gearbeitet haben.
- Sie müssten all das lernen ... *unter Zeitdruck*.

Spezifische Werkzeuge



- Wird Ihnen das jemals etwas nutzen? Ja.
- Erstens diskutieren wir die Grundlagen relationaler Datenbanken und die Werkzeuge sind nur die Beispiele.
- Theoretisches Wissen ist nur begrenzt nützlich in einem praktischen Szenario, wenn man es nicht praktisch umsetzen kann.
- Es hilft gar nichts, die Grundlagen von Datenbanken zu kennen, wenn Sie sich nicht mit einem Terminal zum DBMS verbinden und SQL-Kommandos dorthin senden können, wenn Sie nicht wissen, wie man Fehlermeldungen liest, wenn Sie niemals mit einem echten DBMS gearbeitet haben.
- Sie müssten all das lernen ... *unter Zeitdruck*.
- Wenn Sie aber z.B. PostgreSQL schonmal installiert haben, auf Ihrem Computer, schonmal SQL-Befehle dorthin gesendet haben, schonmal mit PostgreSQL-Fehlermeldungen arbeiten mussten...

Spezifische Werkzeuge



- Wird Ihnen das jemals etwas nutzen? Ja.
- Erstens diskutieren wir die Grundlagen relationaler Datenbanken und die Werkzeuge sind nur die Beispiele.
- Theoretisches Wissen ist nur begrenzt nützlich in einem praktischen Szenario, wenn man es nicht praktisch umsetzen kann.
- Es hilft gar nichts, die Grundlagen von Datenbanken zu kennen, wenn Sie sich nicht mit einem Terminal zum DBMS verbinden und SQL-Kommandos dorthin senden können, wenn Sie nicht wissen, wie man Fehlermeldungen liest, wenn Sie niemals mit einem echten DBMS gearbeitet haben.
- Sie müssten all das lernen ... *unter Zeitdruck*.
- Wenn Sie aber z.B. PostgreSQL schonmal installiert haben, auf Ihrem Computer, schonmal SQL-Befehle dorthin gesendet haben, schonmal mit PostgreSQL-Fehlermeldungen arbeiten mussten...
- ... dann können Sie sicherlich auch sehr schnell lernen, wie man das mit MySQL, MariaDB, oder anderen DBMSen macht.

Gute Praxis

Jede(r) professionelle Informatiker(in), Softwareentwickler(in), Softwareingenieur(in), Datenbankadministrator(in), oder Systemadministrator(in) sollte mit dem Linux Betriebssystem umgehen können.

- Vielleicht können Sie ja mit dem einfach zu nutzenden Ubuntu Linux^{20,37} anfangen.



Gute Praxis

Jede(r) professionelle Informatiker(in), Softwareentwickler(in), Softwareingenieur(in), Datenbankadministrator(in), oder Systemadministrator(in) sollte mit dem Linux Betriebssystem umgehen können.

- Vielleicht können Sie ja mit dem einfach zu nutzenden Ubuntu Linux^{20,37} anfangen.
- Wenn Sie Microsoft Windows¹² nutzen, vielleicht können Sie Ubuntu ja in einer virtuellen Maschine installieren.

Gute Praxis

Jede(r) professionelle Informatiker(in), Softwareentwickler(in), Softwareingenieur(in), Datenbankadministrator(in), oder Systemadministrator(in) sollte mit dem Linux Betriebssystem umgehen können.

- Vielleicht können Sie ja mit dem einfach zu nutzenden Ubuntu Linux^{20,37} anfangen.
- Wenn Sie Microsoft Windows¹² nutzen, vielleicht können Sie Ubuntu ja in einer virtuellen Maschine installieren.
- Ich empfehle jedoch ausdrücklich Linux zu lernen und zu nutzen.



Literatur





- Für diesen Kurs sollten Sie unser Kursbuch lesen: Thomas Weise (汤卫思). *Databases*. Hefei University (合肥大学), School of Artificial Intelligence und Big Data (人工智能与大数据学院), Institute of Applied Optimization (应用优化研究所, IAO), 2025. URL: <https://thomasweise.github.io/databases>.



- Für diesen Kurs sollten Sie unser Kursbuch lesen: Thomas Weise (汤卫思). *Databases*. Hefei University (合肥大学), School of Artificial Intelligence und Big Data (人工智能与大数据学院), Institute of Applied Optimization (应用优化研究所, IAO), 2025. URL: <https://thomasweise.github.io/databases>.
- Um weitere Informationen zu erhalten, empfehle ich das Anschauen der Slides von anderen Vorlesungen über Datenbanken, die online verfügbar sind.



- Für diesen Kurs sollten Sie unser Kursbuch lesen: Thomas Weise (汤卫思). *Databases*. Hefei University (合肥大学), School of Artificial Intelligence und Big Data (人工智能与大数据学院), Institute of Applied Optimization (应用优化研究所, IAO), 2025. URL: <https://thomasweise.github.io/databases>.
- Um weitere Informationen zu erhalten, empfehle ich das Anschauen der Slides von anderen Vorlesungen über Datenbanken, die online verfügbar sind.
- Weiterhin gibt es viele gute Bücher über Datenbanken, von denen Sie sicherlich einige in der Bibliothek finden können, vielleicht sogar in chinesischer Übersetzung.



- Für diesen Kurs sollten Sie unser Kursbuch lesen: Thomas Weise (汤卫思). *Databases*. Hefei University (合肥大学), School of Artificial Intelligence und Big Data (人工智能与大数据学院), Institute of Applied Optimization (应用优化研究所, IAO), 2025. URL: <https://thomasweise.github.io/databases>.
- Um weitere Informationen zu erhalten, empfehle ich das Anschauen der Slides von anderen Vorlesungen über Datenbanken, die online verfügbar sind.
- Weiterhin gibt es viele gute Bücher über Datenbanken, von denen Sie sicherlich einige in der Bibliothek finden können, vielleicht sogar in chinesischer Übersetzung.
- Es gibt auch Webseiten, auf denen man nützliche Informationen finden kann.

Andere Vorlesungen



- Yuriy Shamshin. *Databases*. ISMA University of Applied Sciences, Mai 2024. URL: <https://dbs.academy.lv>,
- Tim Kraska und Michael Cafarella. *6.5830/6.5831: Database Systems*. Massachusetts Institute of Technology (MIT), Herbst 2024. URL: <https://dsg.csail.mit.edu/6.5830>,
- Charles C. Palmer. *COSC 61 Winter 2025: Database Systems*. Dartmouth College, Jan.–März 2025. URL: <https://www.cs.dartmouth.edu/~cs61>,
- Kevin Treu. *CSC-341: Database Management Systems*. Furman University, Frühling 2025. URL: <https://cs.furman.edu/~ktreu/csc341>,
- Junghoo “John” Cho. *CS143: Data Management Systems*. University of California – Los Angeles (UCLA), Sep. 2016–Herbst 2021. URL: <http://oak.cs.ucla.edu/classes/cs143>,
- Saty Raghavachary. *CSCI 585: Database Systems*. University of Southern California (USC), Frühling 2024. URL: <https://bytes.usc.edu/cs585/s24-d-a-t-aaa/lectures>.

Andere Vorlesungen



- Todd J. Green, Hrsg. *ECS 165A Winter 2011 – Introduction to Database Systems*. University of California, Davis, Winter 2011. URL: <https://web.cs.ucdavis.edu/~green/courses/ecs165a-w11>,
- Donnie Pinkston. *CS101b – Introduction to Relational Databases*. California Institute of Technology (Caltech), Winter 2006–Frühling 2007. URL: <http://users.cms.caltech.edu/~donnie/dbcourse/intro0607>,
- Heinz Schweppe und Manuel Scholz. *Einführung in die Datenbanksysteme. Datenbanken für die Bioinformatik*. Freie Universität Berlin, Apr.–Okt. 2005. URL: <https://www.inf.fu-berlin.de/lehre/SS05/19517-V>,
- Scott L. Vandenberg. *CSE 594: Database Management Systems*. University of Washington, Herbst 1999. URL: <https://courses.cs.washington.edu/courses/csep544/99au>.

Generelle Bücher (1)



- Christopher Painter-Wakefield. *A Practical Introduction to Databases*. Runestone Academy, 2022. URL: https://runestone.academy/ns/books/published/practical_db/index.html,
- Abraham “Avi” Silberschatz, Henry F. “Hank” Korth und S. Sudarshan. *Database System Concepts*. 7. Aufl. McGraw-Hill, März 2019. ISBN: 978-0-07-802215-9,
- Jeffrey A. Hoffer, Venkataraman Ramesh und Heikki Topi. *Modern Database Management*. 13. Aufl. Pearson Education, Inc., März 2021. ISBN: 978-0-13-477365-0,
- Ramez Elmasri und Shamkant Navathe. *Fundamentals of Database Systems*. 7. Aufl. Pearson Education, Inc., Juni 2015. ISBN: 978-0-13-397077-7,
- Alfons Kemper und André Eickler. *Datenbanksysteme: Eine Einführung*. Walter de Gruyter GmbH, 2015. ISBN: 978-3-11-044375-2,
- Mana Takahashi, Shoko Azuma und Tokyo, Japan: Trend-Pro Co, Ltd. *The Manga Guide to Databases*. No Starch Press, Jan. 2009. ISBN: 978-1-59327-190-9,
- Michael J. Hernandez. *Database Design for Mere Mortals: 25th Anniversary Edition*. 4. Aufl. Addison-Wesley Professional, Dez. 2020. ISBN: 978-0-13-678813-3,

Generelle Bücher (2)



- Bill Karwin. *SQL Antipatterns: Avoiding the Pitfalls of Database Programming*. Pragmatic Bookshelf by The Pragmatic Programmers, L.L.C., Juni 2017. ISBN: 978-1-934356-55-5,
- Carlos Coronel und Steven Morris. *Database Systems: Design, Implementation, & Management*. 13. Aufl. Cengage Learning, Jan. 2018. ISBN: 978-1-337-62790-0,
- Christopher J. Date. *An Introduction to Database Systems*. 8. Aufl. Pearson Education, Inc., Juli 2003. ISBN: 978-0-321-19784-9,
- Raghu Ramakrishnan und Johannes Gehrke. *Database Management Systems*. 3. Aufl. McGraw-Hill, Aug. 2002. ISBN: 978-0-07-246563-1,
- Jan L. Harrington. *Relational Database Design and Implementation*. 4. Aufl. Morgan Kaufmann Publishers, Apr. 2016. ISBN: 978-0-12-849902-3,
- Hector Garcia-Molina, Jeffrey D. Ullman und Jennifer Widom. *Database Systems: The Complete Book*. 2. Aufl. Pearson Education, Inc., Mai 2008. ISBN: 978-0-13-187325-4,

Generelle Bücher (3)



- Hector Garcia-Molina, Jeffrey D. Ullman und Jennifer Widom. *Database Systems: The Complete Book*. 2. Aufl. Pearson Education, Inc., Mai 2008. ISBN: 978-0-13-187325-4,
- Terry Halpin und Tony Morgan. *Information Modeling and Relational Databases*. 3. Aufl. Morgan Kaufmann Publishers, Juli 2024. ISBN: 978-0-443-23791-1,
- Peter Bailis, Joseph M. Hellerstein und Michael Stonebraker, Hrsg. *Readings in Database Systems*. 5. Aufl. 2015. URL: <http://www.redbook.io>,
- Jim Melton und Alan R. Simon. *SQL: 1999 – Understanding Relational Language Components*. Morgan Kaufmann Publishers, Juni 2001. ISBN: 978-1-55860-456-8,
- John Vincent Carlis und Joseph D. Maguire. *Mastering Data Modeling: A User Driven Approach*. Addison-Wesley Professional, Nov. 2000. ISBN: 978-0-201-70045-9,
- Allen Taylor. *Introducing SQL and Relational Databases*. Apress Media, LLC, Sep. 2018. ISBN: 978-1-4842-3841-7,
- Ryan K. Stephens, Ronald R. Plew, Bryan Morgan und Jeff Perkins. *SQL in 21 Tagen. Die Datenbank-Abfragesprache SQL vollständig erklärt (in 14/21 Tagen)*. 6. Aufl. Markt+Technik Verlag GmbH, Feb. 1998. ISBN: 978-3-8272-2020-2.

Bücher zu Spezifischen Werkzeugen (1)



- Luca Ferrari und Enrico Pirozzi. *Learn PostgreSQL*. 2. Aufl. Packt Publishing Ltd, Okt. 2023. ISBN: 978-1-83763-564-1,
- Alkin Tezuysal und Ibrar Ahmed. *Database Design and Modeling with PostgreSQL and MySQL*. Packt Publishing Ltd, Juli 2024. ISBN: 978-1-80323-347-5,
- Dušan Petković. *Microsoft SQL Server 2019: A Beginner's Guide*. 7. Aufl. McGraw-Hill, Jan. 2020. ISBN: 978-1-260-45888-6,
- Dirk Angermann. *T-SQL-Abfragen für Microsoft SQL-Server 2022*. mitp Verlags GmbH & Co. KG, Juni 2024. ISBN: 978-3-7475-0633-2,
- Darl Kuhn und Thomas Kyte. *Expert Oracle Database Architecture: Techniques and Solutions for High Performance and Productivity*. 4. Aufl. Apress Media, LLC, Nov. 2021. ISBN: 978-1-4842-7499-6,
- Ron McFadyen und Cindy Miller. *Relational Databases and Microsoft Access*. 3. Aufl. Harper College, 2014–2019. URL:
<https://harpercollege.pressbooks.pub/relationaldatabases>,

Bücher zu Spezifischen Werkzeugen (2)



- Laurie A. Ulrich und Ken Cook. *Access For Dummies*. For Dummies (Wiley), Dez. 2021. ISBN: 978-1-119-82908-9,
- Christmas, FL, USA: Simon Sez IT. *Microsoft Access 2021 – Beginner to Advanced*. Packt Publishing Ltd, Aug. 2023. ISBN: 978-1-83546-911-8,
- Ben Beitler. *Hands-On Microsoft Access 2019*. Packt Publishing Ltd, März 2020. ISBN: 978-1-83898-747-3,
- Raul F. Chong, Xiaomei Wang, Michael Dang und Dwaine R. Snow. *Understanding DB2®: Learning Visually with Examples*. 2. Aufl. IBM Press, Dez. 2007. ISBN: 978-0-7686-8177-2,
- Jim Bainbridge, Hernando Bedoya, Rob Bestgen, Mike Cain, Dan Cruikshank, Jim Denton, Doug Mack, Tom Mckinley und Simona Pacchiarini. *SQL Procedures, Triggers, and Functions on IBM DB2 for i*. IBM Redbooks, Apr. 2016. ISBN: 978-0-7384-4164-1.



- *PostgreSQL Documentation*. 17.4. The PostgreSQL Global Development Group (PGDG), Feb. 2025. URL: <https://www.postgresql.org/docs/17/index.html>,
- *Database Administrators*. Stack Exchange Inc. URL: <https://dba.stackexchange.com>,
- Ben Brumm. *Database Star*. Elevated Online Services PTY Ltd., Dez. 2024. URL: <https://www.databasestar.com>,
- Adam “djeada” Djellouli. *Database Notes*. Feb. 2022–März 2025. URL: https://adamdjellouli.com/articles/databases_notes,
- Peter Whyte. *Microsoft SQL Server DBA Blog*. 2018–2025. URL: <https://peter-whyte.com/sql-dba-blog>.



Zusammenfassung



Zusammenfassung



- Es gibt eine Vielzahl an relationalen Datenbankmanagementsystemen.

Zusammenfassung



- Es gibt eine Vielzahl an relationalen Datenbankmanagementsystemen.
- Manche sind Open Source, was gut ist.

Zusammenfassung



- Es gibt eine Vielzahl an relationalen Datenbankmanagementsystemen.
- Manche sind Open Source, was gut ist.
- Andere sind proprietär und kosten Geld.

Zusammenfassung



- Es gibt eine Vielzahl an relationalen Datenbankmanagementsystemen.
- Manche sind Open Source, was gut ist.
- Andere sind proprietär und kosten Geld.
- In diesem Kurs nutzen wir nur kostenfreie Produkte.

Zusammenfassung



- Es gibt eine Vielzahl an relationalen Datenbankmanagementsystemen.
- Manche sind Open Source, was gut ist.
- Andere sind proprietär und kosten Geld.
- In diesem Kurs nutzen wir nur kostenfreie Produkte.
- Und zwar ziemlich viele.

Zusammenfassung



- Es gibt eine Vielzahl an relationalen Datenbankmanagementsystemen.
- Manche sind Open Source, was gut ist.
- Andere sind proprietär und kosten Geld.
- In diesem Kurs nutzen wir nur kostenfreie Produkte.
- Und zwar ziemlich viele.
- Es gibt viele Quellen, von denen Sie zusätzliche Informationen erhalten können.



谢谢您门！
Thank you!
Vielen Dank!



References I



- [1] Dirk Angermann. *T-SQL-Abfragen für Microsoft SQL-Server 2022*. Blaufelden, Schwäbisch Hall, Baden-Württemberg, Germany: mitp Verlags GmbH & Co. KG, Juni 2024. ISBN: **978-3-7475-0633-2** (siehe S. **57–61, 131**).
- [2] Raphael “rkhaotix” Araújo e Silva. *pgModeler – PostgreSQL Database Modeler*. Palmas, Tocantins, Brazil, 2006–2025. URL: <https://pgmodeler.io> (besucht am 2025-04-12) (siehe S. **97–101**).
- [3] David Ascher, Hrsg. *Python Cookbook*. 1. Aufl. Sebastopol, CA, USA: O'Reilly Media, Inc., Juli 2002. ISBN: **978-0-596-00167-4** (siehe S. **87–92**).
- [4] Adam Aspin und Karine Aspin. *Query Answers with MariaDB – Volume I: Introduction to SQL Queries*. Tetras Publishing, Okt. 2018. ISBN: **978-1-9996172-4-0**. See also⁵ (siehe S. **32–34, 142**).
- [5] Adam Aspin und Karine Aspin. *Query Answers with MariaDB – Volume II: In-Depth Querying*. Tetras Publishing, Okt. 2018. ISBN: **978-1-9996172-5-7**. See also⁴ (siehe S. **32–34, 142**).
- [6] Peter Bailis, Joseph M. Hellerstein und Michael Stonebraker, Hrsg. *Readings in Database Systems*. 5. Aufl. Stanford, CA, USA, 2015. URL: <http://www.redbook.io> (besucht am 2025-04-04) (siehe S. **130**).
- [7] Jim Bainbridge, Hernando Bedoya, Rob Bestgen, Mike Cain, Dan Cruikshank, Jim Denton, Doug Mack, Tom Mckinley und Simona Pacchiarini. *SQL Procedures, Triggers, and Functions on IBM DB2 for i*. Durham, NC, USA: IBM Redbooks, Apr. 2016. ISBN: **978-0-7384-4164-1** (siehe S. **66–71, 132**).
- [8] Daniel Bartholomew. *Learning the MariaDB Ecosystem: Enterprise-level Features for Scalability and Availability*. New York, NY, USA: Apress Media, LLC, Okt. 2019. ISBN: **978-1-4842-5514-8** (siehe S. **32–34**).
- [9] Ben Beitler. *Hands-On Microsoft Access 2019*. Birmingham, England, UK: Packt Publishing Ltd, März 2020. ISBN: **978-1-83898-747-3** (siehe S. **46–50, 62–65, 132**).
- [10] Peter Belknap, John Beresniewicz, Benoît Dageville, Karl Dias, Yakov Shafranovich und Khaled Yagoub. “A Decade of Oracle Database Manageability”. 34(4):20–27, Dez. 2011. URL: http://sites.computer.org/debull/A11dec/DODM%5C_V2.pdf (siehe S. **52–56**).
- [11] Silvia Botros und Jeremy Tinley. *High Performance MySQL*. 4. Aufl. Sebastopol, CA, USA: O'Reilly Media, Inc., Nov. 2021. ISBN: **978-1-4920-8051-0** (siehe S. **25–31**).

References II



- [12] Ed Bott. *Windows 11 Inside Out*. Hoboken, NJ, USA: Microsoft Press, Pearson Education, Inc., Feb. 2023. ISBN: 978-0-13-769132-6 (siehe S. 118–120).
- [13] Ben Brumm. *Database Star*. Armadale, VIC, Australia: Elevated Online Services PTY Ltd., Dez. 2024. URL: <https://www.databasestar.com> (besucht am 2025-03-29) (siehe S. 133).
- [14] Jason Cannon. *High Availability for the LAMP Stack*. Shelter Island, NY, USA: Manning Publications, Juni 2022 (siehe S. 25–31).
- [15] John Vincent Carlis und Joseph D. Maguire. *Mastering Data Modeling: A User Driven Approach*. Reading, MA, USA: Addison-Wesley Professional, Nov. 2000. ISBN: 978-0-201-70045-9 (siehe S. 130).
- [16] Donald D. Chamberlin. “50 Years of Queries”. *Communications of the ACM (CACM)* 67(8):110–121, Aug. 2024. New York, NY, USA: Association for Computing Machinery (ACM). ISSN: 0001-0782. doi:10.1145/3649887. URL: <https://cacm.acm.org/research/50-years-of-queries> (besucht am 2025-01-09) (siehe S. 19–31, 35–45, 52–56).
- [17] Junghoo “John” Cho. *CS143: Data Management Systems*. Davis, CA, USA: University of California – Los Angeles (UCLA), Sep. 2016–Herbst 2021. URL: <http://oak.cs.ucla.edu/classes/cs143> (besucht am 2025-04-04) (siehe S. 126).
- [18] Raul F. Chong, Xiaomei Wang, Michael Dang und Dwaine R. Snow. *Understanding DB2®: Learning Visually with Examples*. 2. Aufl. Indianapolis, IN, USA: IBM Press, Dez. 2007. ISBN: 978-0-7686-8177-2 (siehe S. 66–71, 132).
- [19] Christmas, FL, USA: Simon Sez IT. *Microsoft Access 2021 – Beginner to Advanced*. Birmingham, England, UK: Packt Publishing Ltd, Aug. 2023. ISBN: 978-1-83546-911-8 (siehe S. 46–50, 62–65, 132).
- [20] David Clinton und Christopher Negus. *Ubuntu Linux Bible*. 10. Aufl. Bible Series. Chichester, West Sussex, England, UK: John Wiley and Sons Ltd., 10. Nov. 2020. ISBN: 978-1-119-72233-5 (siehe S. 118–120).
- [21] Carlos Coronel und Steven Morris. *Database Systems: Design, Implementation, & Management*. 13. Aufl. Boston, MA, USA: Cengage Learning, Jan. 2018. ISBN: 978-1-337-62790-0 (siehe S. 129).
- [22] Christopher J. Date. *An Introduction to Database Systems*. 8. Aufl. Hoboken, NJ, USA: Pearson Education, Inc., Juli 2003. ISBN: 978-0-321-19784-9 (siehe S. 129).

References III



- [23] *DB-Engines Ranking of Relational DBMS (June 2025)*. Cambridge, England, UK: Red Gate Software Ltd., Juni 2025. URL: <https://db-engines.com/en/ranking/relational+dbms> (besucht am 2025-06-01) (siehe S. 5–7, 52–61, 66–71).
- [24] *Database Administrators*. New York, NY, USA: Stack Exchange Inc. URL: <https://dba.stackexchange.com> (besucht am 2025-02-27) (siehe S. 133).
- [25] Adam "djeada" Djellouli. *Database Notes*. Berlin, Germany, Feb. 2022–März 2025. URL: https://adamdjellouli.com/articles/databases_notes (besucht am 2025-03-27) (siehe S. 133).
- [26] Russell J.T. Dyer. *Learning MySQL and MariaDB*. Sebastopol, CA, USA: O'Reilly Media, Inc., März 2015. ISBN: 978-1-4493-6290-4 (siehe S. 25–34).
- [27] Ramez Elmasri und Shamkant Navathe. *Fundamentals of Database Systems*. 7. Aufl. Hoboken, NJ, USA: Pearson Education, Inc., Juni 2015. ISBN: 978-0-13-397077-7 (siehe S. 128).
- [28] Steve Fanning, Vasudev Narayanan, "flywire", Olivier Hallot, Jean Hollis Weber, Jenna Sargent, Pulkit Krishna, Dan Lewis, Peter Schofield, Jochen Schiffers, Robert Großkopf, Jost Lange, Martin Fox, Hazel Russman, Steve Schwettman, Alain Romedenne, Andrew Pitonyak, Jean-Pierre Ledure, Drew Jensen und Randolph Gam. *Base Guide 7.3. Revision 1. Based on LibreOffice 7.3 Community*. Berlin, Germany: The Document Foundation, Aug. 2022. URL: <https://books.libreoffice.org/en/BG73/BG73-BaseGuide.pdf> (besucht am 2025-01-13) (siehe S. 46–50, 82–86).
- [29] Luca Ferrari und Enrico Pirozzi. *Learn PostgreSQL*. 2. Aufl. Birmingham, England, UK: Packt Publishing Ltd, Okt. 2023. ISBN: 978-1-83763-564-1 (siehe S. 35–40, 78–81, 131).
- [30] Kevin P. Gaffney, Martin Prammer, Laurence C. Brasfield, D. Richard Hipp, Dan R. Kennedy und Jignesh M. Patel. "SQLite: Past, Present, and Future". *Proceedings of the VLDB Endowment (PVLDB)* 15(12):3535–3547, Aug. 2022. Irvine, CA, USA: Very Large Data Bases Endowment Inc. ISSN: 2150-8097. doi:10.14778/3554821.3554842. URL: <https://www.vldb.org/pvldb/vol15/p3535-gaffney.pdf> (besucht am 2025-01-12). All papers in this issue were presented at the 48th International Conference on Very Large Data Bases (VLDB 2022), Sep. 5-9, 2022, hybrid/Sydney, NSW, Australia (siehe S. 41–45).

References IV



- [31] Jonas Gamalielsson und Björn Lundell. "Long-Term Sustainability of Open Source Software Communities beyond a Fork: A Case Study of LibreOffice". In: *8th IFIP WG 2.13 International Conference on Open Source Systems: Long-Term Sustainability OSS'2012*. 10.–13. Sep. 2012, Hammamet, Tunisia. Hrsg. von Imed Hammouda, Björn Lundell, Tommi Mikkonen und Walt Scacchi. Bd. 378. IFIP Advances in Information and Communication Technology (IFIPACT). Heidelberg, Baden-Württemberg, Germany: Springer-Verlag GmbH Germany, 2012, S. 29–47. ISSN: 1868-4238. ISBN: 978-3-642-33441-2. doi:10.1007/978-3-642-33442-9_3 (siehe S. 46–50).
- [32] Hector Garcia-Molina, Jeffrey D. Ullman und Jennifer Widom. *Database Systems: The Complete Book*. 2. Aufl. Hoboken, NJ, USA: Pearson Education, Inc., Mai 2008. ISBN: 978-0-13-187325-4 (siehe S. 129, 130).
- [33] Todd J. Green, Hrsg. *ECS 165A Winter 2011 – Introduction to Database Systems*. Davis, CA, USA: University of California, Davis, Winter 2011. URL: <https://web.cs.ucdavis.edu/~green/courses/ecs165a-w11> (besucht am 2025-03-25) (siehe S. 127).
- [34] Donald J. Haderle und Cynthia M. Saracco. "The History and Growth of IBM's DB2". *IEEE Annals of the History of Computing* 35(2):54–66, Apr. 2013–Juni 2014. Piscataway, NJ, USA: Institute of Electrical and Electronics Engineers (IEEE). ISSN: 1058-6180. doi:10.1109/MAHC.2012.55 (siehe S. 66–71).
- [35] Terry Halpin und Tony Morgan. *Information Modeling and Relational Databases*. 3. Aufl. Burlington, MA, USA/San Mateo, CA, USA: Morgan Kaufmann Publishers, Juli 2024. ISBN: 978-0-443-23791-1 (siehe S. 130).
- [36] Jan L. Harrington. *Relational Database Design and Implementation*. 4. Aufl. Burlington, MA, USA/San Mateo, CA, USA: Morgan Kaufmann Publishers, Apr. 2016. ISBN: 978-0-12-849902-3 (siehe S. 129).
- [37] Matthew Helmke. *Ubuntu Linux Unleashed 2021 Edition*. 14. Aufl. Reading, MA, USA: Addison-Wesley Professional, Aug. 2020. ISBN: 978-0-13-668539-5 (siehe S. 25–31, 118–120).
- [38] Michael J. Hernandez. *Database Design for Mere Mortals: 25th Anniversary Edition*. 4. Aufl. Reading, MA, USA: Addison-Wesley Professional, Dez. 2020. ISBN: 978-0-13-678813-3 (siehe S. 128).
- [39] D. Richard Hipp u. a. "Well-Known Users of SQLite". In: *SQLite*. Charlotte, NC, USA: Hipp, Wyrick & Company, Inc. (Hwaci), 2. Jan. 2023. URL: <https://www.sqlite.org/famous.html> (besucht am 2025-01-12) (siehe S. 41–45).

References V



- [40] Jeffrey A. Hoffer, Venkataraman Ramesh und Heikki Topi. *Modern Database Management*. 13. Aufl. Hoboken, NJ, USA: Pearson Education, Inc., März 2021. ISBN: **978-0-13-477365-0** (siehe S. 128).
- [41] Manuel Hoffmann, Frank Nagle und Yanuo Zhou. *The Value of Open Source Software*. Working Paper 24-038. Boston, MA, USA: Harvard Business School, 1. Jan. 2024. URL: https://www.hbs.edu/ris/Publication%20Files/24-038_51f8444f-502c-4139-8bf2-56eb4b65c58a.pdf (besucht am 2025-06-04) (siehe S. 9–17).
- [42] John Hunt. *A Beginners Guide to Python 3 Programming*. 2. Aufl. Undergraduate Topics in Computer Science (UTICS). Cham, Switzerland: Springer, 2023. ISBN: **978-3-031-35121-1**. doi:10.1007/978-3-031-35122-8 (siehe S. 87–92).
- [43] Bill Karwin. *SQL Antipatterns: Avoiding the Pitfalls of Database Programming*. Flower Mound, TX, USA: Pragmatic Bookshelf by The Pragmatic Programmers, L.L.C., Juni 2017. ISBN: **978-1-934356-55-5** (siehe S. 129).
- [44] Alfons Kemper und André Eickler. *Datenbanksysteme: Eine Einführung*. De Gruyter Studium. Berlin, Germany: Walter de Gruyter GmbH, 2015. ISBN: **978-3-11-044375-2** (siehe S. 128).
- [45] Bernd Klein. *Einführung in Python 3 – Für Ein- und Umsteiger*. 3., überarbeitete. München, Bayern, Germany: Carl Hanser Verlag GmbH & Co. KG, 2018. ISBN: **978-3-446-45208-4**. doi:10.3139/9783446453876 (siehe S. 87–92).
- [46] Tim Kraska und Michael Cafarella. *6.5830/6.5831: Database Systems*. Cambridge, MA, USA: Massachusetts Institute of Technology (MIT), Herbst 2024. URL: <https://dsg.csail.mit.edu/6.5830> (besucht am 2025-01-08) (siehe S. 126).
- [47] Darl Kuhn und Thomas Kyte. *Expert Oracle Database Architecture: Techniques and Solutions for High Performance and Productivity*. 4. Aufl. New York, NY, USA: Apress Media, LLC, Nov. 2021. ISBN: **978-1-4842-7499-6** (siehe S. 52–56, 131).
- [48] E. A. Kurako und V. L. Orlov. "Database Migration from ORACLE to PostgreSQL". 49(5):455–463, Okt. 2023. doi:10.1134/S0361768823050055 (siehe S. 52–56).
- [49] Joan Lambert und Curtis Frye. *Microsoft Office Step by Step (Office 2021 and Microsoft 365)*. Hoboken, NJ, USA: Microsoft Press, Pearson Education, Inc., Juni 2022. ISBN: **978-0-13-754493-6** (siehe S. 62–65).
- [50] Kent D. Lee und Steve Hubbard. *Data Structures and Algorithms with Python*. Undergraduate Topics in Computer Science (UTICS). Cham, Switzerland: Springer, 2015. ISBN: **978-3-319-13071-2**. doi:10.1007/978-3-319-13072-9 (siehe S. 87–92).

References VI



- [51] *LibreOffice – The Document Foundation*. Berlin, Germany: The Document Foundation, 2024. URL: <https://www.libreoffice.org> (besucht am 2024-12-12) (siehe S. 46–50).
- [52] Ron McFadyen und Cindy Miller. *Relational Databases and Microsoft Access*. 3. Aufl. Palatine, IL, USA: Harper College, 2014–2019. URL: <https://harpercollege.pressbooks.pub/relationaldatabases> (besucht am 2025-04-11) (siehe S. 62–65, 131).
- [53] Jim Melton und Alan R. Simon. *SQL: 1999 – Understanding Relational Language Components*. The Morgan Kaufmann Series in Data Management Systems. Burlington, MA, USA/San Mateo, CA, USA: Morgan Kaufmann Publishers, Juni 2001. ISBN: 978-1-55860-456-8 (siehe S. 130).
- [54] Regina O. Obe und Leo S. Hsu. *PostgreSQL: Up and Running*. 3. Aufl. Sebastopol, CA, USA: O'Reilly Media, Inc., Okt. 2017. ISBN: 978-1-4919-6336-4 (siehe S. 35–40, 78–81).
- [55] "Oracle Timeline: Highlighting the most important moments in Oracle's history, with commentary from the people who made it happen". *Profit – The Executive Guide to Oracle Applications* 12(2):26–33, Mai 2007. Redwood Shores, CA, USA: Oracle Corporation. ISSN: 1531-7455. URL: <https://www.oracle.com/us/corporate/profit/profit-may-07-151925.pdf> (besucht am 2025-06-03) (siehe S. 52–56).
- [56] Christopher Painter-Wakefield. *A Practical Introduction to Databases*. Luck, WI, USA: Runestone Academy, 2022. URL: https://runestone.academy/ns/books/published/practical_db/index.html (besucht am 2025-04-06) (siehe S. 128).
- [57] Camila A. Paiva, Raquel Maximino, Frederico Paiva, Rafael Accetta Vieira, Nicole Espanha, João Felipe Pimentel, Igor Wiese, Marco Aurélio Gerosa, Igor Steinmacher, Leonardo Murta und Vanessa Braganholo. "Analyzing the Adoption of Database Management Systems throughout the History of Open Source Projects". *Empirical Software Engineering: An International Journal* 30(3):71, Feb. 2025. London, England, UK: Springer Nature Limited. ISSN: 1382-3256. doi:10.1007/S10664-025-10627-Z. URL: <https://www.authorea.com/users/677798/articles/674742> (besucht am 2025-06-04) (siehe S. 25–40, 52–56).
- [58] Charles C. Palmer. *COSC 61 Winter 2025: Database Systems*. Hanover, MD, USA: Dartmouth College, Jan.–März 2025. URL: <https://www.cs.dartmouth.edu/~cs61> (besucht am 2025-04-06) (siehe S. 126).
- [59] Linda Dailey Paulson. "Open Source Databases Move into the Marketplace". *Computer* 37(7):13–15, Juli 2004. Piscataway, NJ, USA: Institute of Electrical and Electronics Engineers (IEEE). ISSN: 0018-9162. doi:10.1109/MC.2004.62 (siehe S. 19–24).

References VII



- [60] Dušan Petković. *Microsoft SQL Server 2019: A Beginner's Guide*. 7. Aufl. New York, NY, USA: McGraw-Hill, Jan. 2020. ISBN: **978-1-260-45888-6** (siehe S. **57–61, 131**).
- [61] Donnie Pinkston. *CS101b – Introduction to Relational Databases*. Pasadena, CA, USA: California Institute of Technology (Caltech), Winter 2006–Frühling 2007. URL: <http://users.cms.caltech.edu/~donnie/dbcourse/intro0607> (besucht am 2025-04-04) (siehe S. **127**).
- [62] *PostgreSQL Documentation*. 17.4. The PostgreSQL Global Development Group (PGDG), Feb. 2025. URL: <https://www.postgresql.org/docs/17/index.html> (besucht am 2025-02-25) (siehe S. **133**).
- [63] *PostgreSQL Essentials: Leveling Up Your Data Work*. Sebastopol, CA, USA: O'Reilly Media, Inc., März 2024 (siehe S. **35–40, 78–81**).
- [64] Saty Raghavachary. *CSCI 585: Database Systems*. Los Angeles, CA, USA: University of Southern California (UCS), Frühling 2024. URL: <https://bytes.usc.edu/cs585/s24-d-a-t-aaa/lectures> (besucht am 2025-04-06) (siehe S. **126**).
- [65] Raghu Ramakrishnan und Johannes Gehrke. *Database Management Systems*. 3. Aufl. New York, NY, USA: McGraw-Hill, Aug. 2002. ISBN: **978-0-07-246563-1** (siehe S. **129**).
- [66] Federico Razzoli. *Mastering MariaDB*. Birmingham, England, UK: Packt Publishing Ltd, Sep. 2014. ISBN: **978-1-78398-154-0** (siehe S. **32–34**).
- [67] Mike Reichardt, Michael Gundall und Hans D. Schotten. "Benchmarking the Operation Times of NoSQL and MySQL Databases for Python Clients". In: *47th Annual Conference of the IEEE Industrial Electronics Society (IECON'2021)*. 13.–15. Okt. 2021, Toronto, ON, Canada. Piscataway, NJ, USA: Institute of Electrical and Electronics Engineers (IEEE), Okt. 2021, S. 1–8. ISSN: **2577-1647**. ISBN: **978-1-6654-3554-3**. doi:10.1109/IECON48115.2021.9589382 (siehe S. **25–31**).
- [68] Heinz Schweppe und Manuel Scholz. *Einführung in die Datenbanksysteme. Datenbanken für die Bioinformatik*. Berlin, Germany: Freie Universität Berlin, Apr.–Okt. 2005. URL: <https://www.inf.fu-berlin.de/lehre/SS05/19517-V> (besucht am 2025-01-08) (siehe S. **127**).

References VIII



- [69] Matthias Sedlmeier und Martin Gogolla. "Model Driven ActiveRecord with yEd". In: *25th International Conference on Information Modelling and Knowledge Bases XXVII (EJC'2015)*. 8.–12. Juni 2015, Maribor, Štajerska, Podravska, Slovenia. Hrsg. von Tatjana Welzer, Hannu Jaakkola, Bernhard Thalheim, Yasushi Kiyoki und Naofumi Yoshida. Bd. 280. Frontiers in Artificial Intelligence and Applications. Amsterdam, The Netherlands: IOS Press BV, S. 65–76. ISSN: 0922-6389. ISBN: 978-1-61499-610-1. doi:10.3233/978-1-61499-611-8-65 (siehe S. 93–96).
- [70] Winfried Seimert. *LibreOffice 7.3 – Praxiswissen für Ein- und Umsteiger*. Blaufelden, Schwäbisch Hall, Baden-Württemberg, Germany: mitp Verlags GmbH & Co. KG, Apr. 2022. ISBN: 978-3-7475-0504-5 (siehe S. 46–50, 82–86).
- [71] Yuriy Shamshin. *Databases*. Riga, Latvia: ISMA University of Applied Sciences, Mai 2024. URL: <https://dbs.academy.lv> (besucht am 2025-01-11) (siehe S. 126).
- [72] Abraham "Avi" Silberschatz, Henry F. "Hank" Korth und S. Sudarshan. *Database System Concepts*. 7. Aufl. New York, NY, USA: McGraw-Hill, März 2019. ISBN: 978-0-07-802215-9 (siehe S. 128).
- [73] "Stack Overflow 2024 Developer Survey". In: *Stack Overflow*. New York, NY, USA: Stack Exchange Inc., Mai–Juni 2024. URL: <https://survey.stackoverflow.co/2024> (besucht am 2025-06-01) (siehe S. 5–8, 25–31, 35–40, 66–71).
- [74] Ryan K. Stephens und Ronald R. Plew. *Sams Teach Yourself SQL in 21 Days*. 4. Aufl. Sams Tech Yourself. Indianapolis, IN, USA: SAMS Technical Publishing und Hoboken, NJ, USA: Pearson Education, Inc., Okt. 2002. ISBN: 978-0-672-32451-2 (siehe S. 149).
- [75] Ryan K. Stephens, Ronald R. Plew, Bryan Morgan und Jeff Perkins. *SQL in 21 Tagen. Die Datenbank-Abfragesprache SQL vollständig erklärt (in 14/21 Tagen)*. 6. Aufl. Burghann, Bayern, Germany: Markt+Technik Verlag GmbH, Feb. 1998. ISBN: 978-3-8272-2020-2. Translation of ⁷⁴ (siehe S. 130).
- [76] Mana Takahashi, Shoko Azuma und Tokyo, Japan: Trend-Pro Co, Ltd. *The Manga Guide to Databases*. San Francisco, CA, USA: No Starch Press, Jan. 2009. ISBN: 978-1-59327-190-9 (siehe S. 128).
- [77] Allen Taylor. *Introducing SQL and Relational Databases*. New York, NY, USA: Apress Media, LLC, Sep. 2018. ISBN: 978-1-4842-3841-7 (siehe S. 130).

References IX



- [78] Alkin Tezuysal und Ibrar Ahmed. *Database Design and Modeling with PostgreSQL and MySQL*. Birmingham, England, UK: Packt Publishing Ltd, Juli 2024. ISBN: 978-1-80323-347-5 (siehe S. 25–31, 35–40, 78–81, 131).
- [79] “The Evolution of Microsoft SQL Server”. In: 15. Feb. 2025. URL: <https://peter-whyte.com/2025/02/the-evolution-of-microsoft-sql-server> (besucht am 2025-06-03) (siehe S. 57–61).
- [80] Kevin Treu. *CSC-341: Database Management Systems*. Greenville, SC, USA: Furman University, Frühling 2025. URL: <https://cs.furman.edu/~ktreu/csc341> (besucht am 2025-04-05) (siehe S. 126).
- [81] Laurie A. Ulrich und Ken Cook. *Access For Dummies*. Hoboken, NJ, USA: For Dummies (Wiley), Dez. 2021. ISBN: 978-1-119-82908-9 (siehe S. 46–50, 62–65, 132).
- [82] Scott L. Vandenberg. *CSE 594: Database Management Systems*. Seattle, WA, USA: University of Washington, Herbst 1999. URL: <https://courses.cs.washington.edu/courses/csep544/99au> (besucht am 2025-03-25) (siehe S. 127).
- [83] Daniele “dvarrazzo” Varrazzo, Federico “fogzot” Di Gregorio und Jason “jerickso” Erickson. *Psycopg*. London, England, UK: The Psycopg Team, 2010–2023. URL: <https://www.psycopg.org> (besucht am 2025-02-02) (siehe S. 87–92).
- [84] Thomas Weise (汤卫恩). *Databases*. Hefei, Anhui, China (中国安徽省合肥市): Hefei University (合肥大学), School of Artificial Intelligence und Big Data (人工智能与大数据学院), Institute of Applied Optimization (应用优化研究所, IAO), 2025. URL: <https://thomasweise.github.io/databases> (besucht am 2025-01-05) (siehe S. 122–125).
- [85] Thomas Weise (汤卫恩). *Programming with Python*. Hefei, Anhui, China (中国安徽省合肥市): Hefei University (合肥大学), School of Artificial Intelligence und Big Data (人工智能与大数据学院), Institute of Applied Optimization (应用优化研究所, IAO), 2024–2025. URL: <https://thomasweise.github.io/programmingWithPython> (besucht am 2025-01-05) (siehe S. 87–92).
- [86] Peter Whyte. *Microsoft SQL Server DBA Blog*. Edinburgh, Scotland, UK, 2018–2025. URL: <https://peter-whyte.com/sql-dba-blog> (besucht am 2025-06-03) (siehe S. 57–61, 133).
- [87] Ulf Michael “Monty” Widenius, David Axmark und Uppsala, Sweden: MySQL AB. *MySQL Reference Manual – Documentation from the Source*. Sebastopol, CA, USA: O'Reilly Media, Inc., 9. Juli 2002. ISBN: 978-0-596-00265-7 (siehe S. 25–31).

References X



- [88] Marianne Winslett und Vanessa Braganholo. "Richard Hipp Speaks Out on SQLite". *ACM SIGMOD Record* 48(2):39–46, Juni 2019. New York, NY, USA: Association for Computing Machinery (ACM). ISSN: **0163-5808**. doi:[10.1145/3377330.3377338](https://doi.org/10.1145/3377330.3377338) (siehe S. **41–45**).
- [89] *yEd Graph Editor Manual*. Tübingen, Baden-Württemberg, Germany: yWorks GmbH, 2011–2025. URL: <https://yed.yworks.com/support/manual/index.html> (besucht am 2025-03-31) (siehe S. **93–96**).

