



合肥大學
HEFEI UNIVERSITY



Datenbanken

5. Software und Literatur

Thomas Weise (汤卫思)
twaise@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-Code 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)*³⁹ listed 166 solche Produkte.



Einleitung

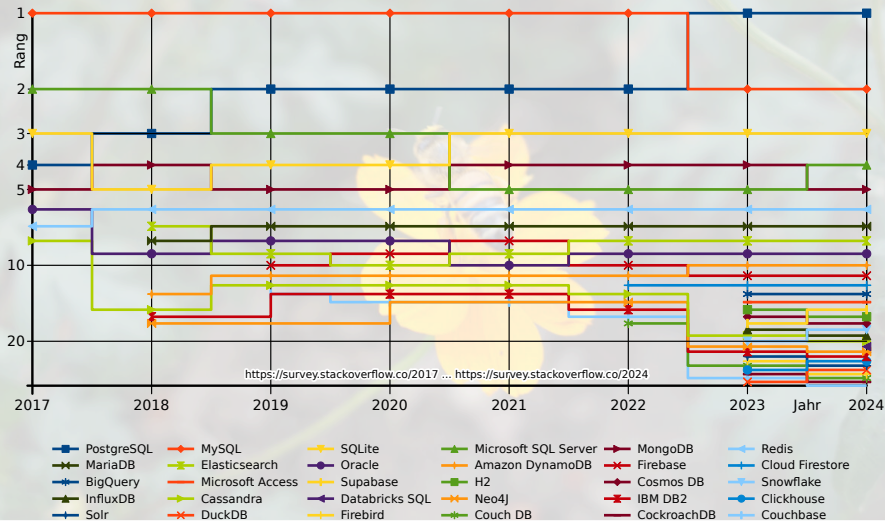


- Heute gibt es eine große Auswahl an relationalen Datenbankmanagementsystemen.
- *DB-Engines Ranking of Relational DBMS (June 2025)*³⁹ listed 166 solche Produkte.
- Im “Stack Overflow 2024 Developer Survey”¹¹⁹ konnten Programmierer Feedback zu 35 Systemen geben.



seit 201

- 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 dollar⁶⁵.

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 dollar⁶⁵.

- 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 dollar⁶⁵.

- 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 dollar⁶⁵.

- 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 Aufstieg⁹⁵.

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 Aufstieg⁹⁵.
- Nun sind sie weitverbreitete und gut getestete Produkte²⁶.

MySQL



- MySQL ist so ein Open Source DBMS für relationale Datenbanken^{18,46,106,124,141}.



MySQL



- MySQL ist so ein Open Source DBMS für relationale Datenbanken^{18,46,106,124,141}.
- Es wurde ursprünglich von Michael Widenius und David Axmark in der schwedischen Forma MySQL AB entwickelt und kam 1996 heraus²⁶.



MySQL



- MySQL ist so ein Open Source DBMS für relationale Datenbanken^{18,46,106,124,141}.
- Es wurde ursprünglich von Michael Widenius und David Axmark in der schwedischen Forma MySQL AB entwickelt und kam 1996 heraus²⁶.
- 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^{24,61}.

MySQL



- MySQL ist so ein Open Source DBMS für relationale Datenbanken^{18,46,106,124,141}.
- Es wurde ursprünglich von Michael Widenius und David Axmark in der schwedischen Forma MySQL AB entwickelt und kam 1996 heraus²⁶.
- 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^{24,61}.
- Es stand bis ins Jahr 2022 an erster Stelle als beliebtestes DBMS in den Stack Overflow Developer Surveys¹¹⁹.

MySQL



- MySQL ist so ein Open Source DBMS für relationale Datenbanken^{18,46,106,124,141}.
- Es wurde ursprünglich von Michael Widenius und David Axmark in der schwedischen Forma MySQL AB entwickelt und kam 1996 heraus²⁶.
- 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^{24,61}.
- Es stand bis ins Jahr 2022 an erster Stelle als beliebtestes DBMS in den Stack Overflow Developer Surveys¹¹⁹.
- In dem Survey⁹³ 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^{18,46,106,124,141}.
- Es wurde ursprünglich von Michael Widenius und David Axmark in der schwedischen Forma MySQL AB entwickelt und kam 1996 heraus²⁶.
- 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^{24,61}.
- Es stand bis ins Jahr 2022 an erster Stelle als beliebtestes DBMS in den Stack Overflow Developer Surveys¹¹⁹.
- In dem Survey⁹³ 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 wurden²⁶.

MySQL



- MySQL ist so ein Open Source DBMS für relationale Datenbanken^{18,46,106,124,141}.
- Es wurde ursprünglich von Michael Widenius und David Axmark in der schwedischen Forma MySQL AB entwickelt und kam 1996 heraus²⁶.
- 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^{24,61}.
- Es stand bis ins Jahr 2022 an erster Stelle als beliebtestes DBMS in den Stack Overflow Developer Surveys¹¹⁹.
- In dem Survey⁹³ 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 wurden²⁶.
- 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^{4,5,10,46,105}.



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^{4,5,10,46,105}.
- Sie versprachen, dass MariaDB für immer OSS bleiben wird.



MariaDB

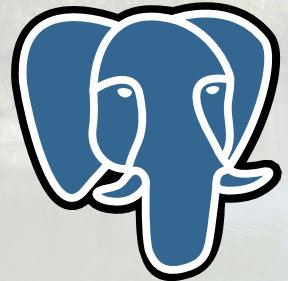
The MariaDB logo is under the copyright of its owners.

- Nach dem Aufkauf von Sun durch Oracle gründeten einige der ursprünglichen MySQL-Entwickler, auch Michael Widenius, den MySQL-Fork MariaDB^{4,5,10,46,105}.
- Sie versprachen, dass MariaDB für immer OSS bleiben wird.
- In der Studie⁹³ über Datenbanknutzung von Java OSS Programmen war MariaDB schon das fünft-populäre relationale DBMS.

PostgreSQL



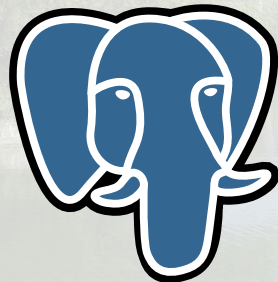
- PostgreSQL^{50,88,100,124} ist ein Objekt-relationales DBMS, das auch Konzepte aus der Objekt-Orientierten Programmierung (OOP) wie Vererbung unter Tabellen unterstützt.



PostgreSQL



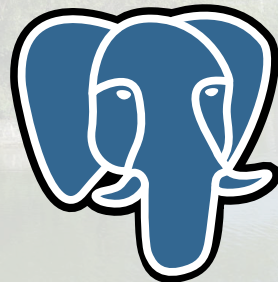
- PostgreSQL^{50,88,100,124} ist ein Objekt-relationales DBMS, das auch Konzepte aus der Objekt-Orientierten Programmierung (OOP) wie Vererbung unter Tabellen unterstützt.
- Es unterstützt auch zusätzliche Datentypen, wie JSON-Objekte und geometrische Daten.



PostgreSQL



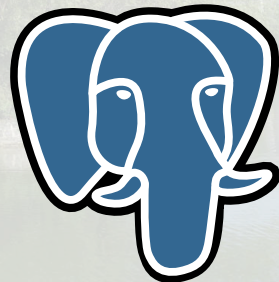
- PostgreSQL^{50,88,100,124} ist ein Objekt-relationales DBMS, das auch Konzepte aus der Objekt-Orientierten Programmierung (OOP) wie Vererbung unter Tabellen unterstützt.
- Es unterstützt auch zusätzliche Datentypen, wie JSON-Objekte und geometrische Daten.
- Es stammt aus dem POSTGRES-Projekt, dem Nachfolger des berühmten INGRES-Projekts an der University of Berkeley in Kalifornien²⁶.



PostgreSQL



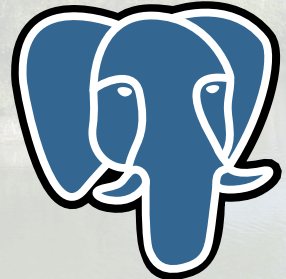
- PostgreSQL^{50,88,100,124} ist ein Objekt-relationales DBMS, das auch Konzepte aus der Objekt-Orientierten Programmierung (OOP) wie Vererbung unter Tabellen unterstützt.
- Es unterstützt auch zusätzliche Datentypen, wie JSON-Objekte und geometrische Daten.
- Es stammt aus dem POSTGRES-Projekt, dem Nachfolger des berühmten INGRES-Projekts an der University of Berkeley in Kalifornien²⁶.
- Es ist vielleicht das OSS DBMS mit den meisten Features.



PostgreSQL



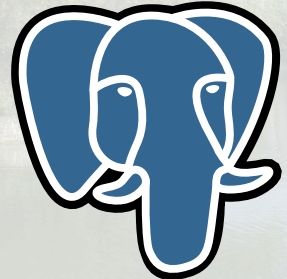
- PostgreSQL^{50,88,100,124} ist ein Objekt-relationales DBMS, das auch Konzepte aus der Objekt-Orientierten Programmierung (OOP) wie Vererbung unter Tabellen unterstützt.
- Es unterstützt auch zusätzliche Datentypen, wie JSON-Objekte und geometrische Daten.
- Es stammt aus dem POSTGRES-Projekt, dem Nachfolger des berühmten INGRES-Projekts an der University of Berkeley in Kalifornien²⁶.
- Es ist vielleicht das OSS DBMS mit den meisten Features.
- Im "Stack Overflow 2024 Developer Survey" war es das populärste DBMS¹¹⁹.



PostgreSQL



- PostgreSQL^{50,88,100,124} ist ein Objekt-relationales DBMS, das auch Konzepte aus der Objekt-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 Kalifornien²⁶.
- Es ist vielleicht das OSS DBMS mit den meisten Features.
- Im "Stack Overflow 2024 Developer Survey" war es das populärste DBMS¹¹⁹.
- Im Open Source Quellcode-Survey⁹³ war es auf Rang 2.



PostgreSQL



- die Structured Query Language (SQL)-Datenbank mit der weitesten Verbreitung ist jedoch SQLite^{26,51,63,142}.



The SQLite logo is under the copyright of its owners.

PostgreSQL



- die SQL-Datenbank mit der weitesten Verbreitung ist jedoch SQLite^{26,51,63,142}.
- 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^{26,51,63,142}.
- 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^{26,51,63,142}.
- 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^{26,51,63,142}.
- 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 bekommen²⁶.



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^{52,79,110} implementiert: Als LibreOffice Base^{49,110}.



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^{52,79,110} implementiert: Als LibreOffice Base^{49,110}.
- 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^{52,79,110} implementiert: Als LibreOffice Base^{49,110}.
- 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^{52,79,110} implementiert: Als LibreOffice Base^{49,110}.
- 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^{52,79,110} implementiert: Als LibreOffice Base^{49,110}.
- 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^{11,32,130}.



LibreOffice®

The LibreOffice logo is under the copyright of its owners.



Proprietäre Datenbankmanagementsysteme



Oracle Database



- Die erste kommerzielle SQL-Datenbank war die Oracle Database^{26,90}.

Oracle Database



- Die erste kommerzielle SQL-Datenbank war die Oracle Database^{26,90}.
- Sie ist noch heute eine der erfolgreichsten und weitverbreiteten kommerziellen Datenbanken, mit vielen hochentwickelten Features^{12,73}.

Oracle Database



- Die erste kommerzielle SQL-Datenbank war die Oracle Database^{26,90}.
- Sie ist noch heute eine der erfolgreichsten und weitverbreiteten kommerziellen Datenbanken, mit vielen hochentwickelten Features^{12,73}.
- Im Juni 2025 war sie das populärste DMBS überhaupt im Survey³⁹.

Oracle Database



- Die erste kommerzielle SQL-Datenbank war die Oracle Database^{26,90}.
- Sie ist noch heute eine der erfolgreichsten und weitverbreiteten kommerziellen Datenbanken, mit vielen hochentwickelten Features^{12,73}.
- Im Juni 2025 war sie das populärste DMBS überhaupt im Survey³⁹.
- In dem Java Quellkodesurvey⁹³ war sie auf Rang 4.

Oracle Database



- Die erste kommerzielle SQL-Datenbank war die Oracle Database^{26,90}.
- Sie ist noch heute eine der erfolgreichsten und weitverbreiteten kommerziellen Datenbanken, mit vielen hochentwickelten Features^{12,73}.
- Im Juni 2025 war sie das populärste DMBS überhaupt im Survey³⁹.
- In dem Java Quellkodesurvey⁹³ war sie auf Rang 4.
- Die Migration von Oracle Database nach PostgreSQL ist in⁷⁴ diskutiert.

Microsoft SQL Server



- Auf Rang 3 im Survey³⁹ stand der Microsoft SQL Server^{1,97,140}.

Microsoft SQL Server



- Auf Rang 3 im Survey³⁹ stand der Microsoft SQL Server^{1,97,140}.
- 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 entwickelten¹²⁵.

Microsoft SQL Server



- Auf Rang 3 im Survey³⁹ stand der Microsoft SQL Server^{1,97,140}.
- 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 entwickelten¹²⁵.
- Die erste Version dieses DBMS kam dann 1989 heraus.

Microsoft SQL Server



- Auf Rang 3 im Survey³⁹ stand der Microsoft SQL Server^{1,97,140}.
- 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 entwickelten¹²⁵.
- Die erste Version dieses DBMS kam dann 1989 heraus.
- Spätere Versionen unterstützten Microsoft Windows NT.

Microsoft SQL Server



- Auf Rang 3 im Survey³⁹ stand der Microsoft SQL Server^{1,97,140}.
- 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 entwickelten¹²⁵.
- 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).

- Mit Microsoft Access⁷⁶ bietet Microsoft auch ein DBMS an, das hauptsächlich für einzelne Benutzer auf einem einzelnen Computer gedacht ist.



- Mit Microsoft Access⁷⁶ 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^{11,32,84,130}.



- Mit Microsoft Access⁷⁶ 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^{11,32,84,130}.
- Es war sicherlich die Vorlage für LibreOffice Base.



- Mit Microsoft Access⁷⁶ 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^{11,32,84,130}.
- 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^{7,31} ist ein weiteres frühes relationales DBMS⁵⁷.

IBM DB2



- IBM's DB2^{7,31} ist ein weiteres frühes relationales DBMS⁵⁷.
- Es entstand als Software für mächtige Mainframe-Computer, die IBM selbst hergestellt hat.

IBM DB2



- IBM's DB2^{7,31} ist ein weiteres frühes relationales DBMS⁵⁷.
- 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^{7,31} ist ein weiteres frühes relationales DBMS⁵⁷.
- 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's DB2^{7,31} ist ein weiteres frühes relationales DBMS⁵⁷.
- 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 installiert⁵⁷.



- IBM's DB2^{7,31} ist ein weiteres frühes relationales DBMS⁵⁷.
- 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 installiert⁵⁷.
- Die DB2 ist auf Rang 6 im *DB-Engines Ranking of Relational DBMS (June 2025)*³⁹ und auf Rang 23 im "Stack Overflow 2024 Developer Survey"¹¹⁹.



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.

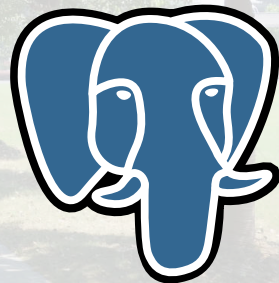
Dieser Kurs



- 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.

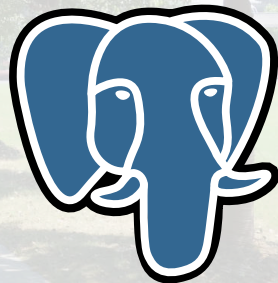


The PostgreSQL logo is under the copyright of its owners.

Datenbankmanagementsystem: PostgreSQL



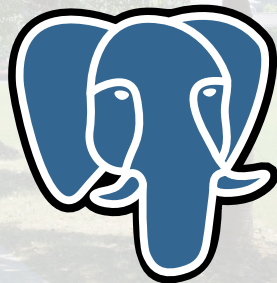
- Wir werden mit einem echten DBMS arbeiten.
- Wir haben uns für PostgreSQL^{50,88,100,124} entschieden.



Datenbankmanagementsystem: PostgreSQL



- Wir werden mit einem echten DBMS arbeiten.
- Wir haben uns für PostgreSQL^{50,88,100,124} entschieden.
- DBMSs sind normalerweise Server an die man SQL-Befehle mit Hilfe eines Clientprogramms vom Terminal aus schicken kann.

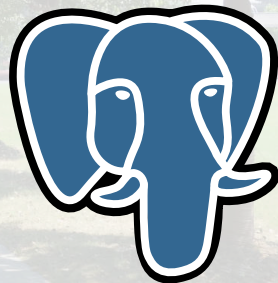


Datenbankmanagementsystem: PostgreSQL



- Wir werden mit einem echten DBMS arbeiten.
- Wir haben uns für PostgreSQL^{50,88,100,124} 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^{49,110}.



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 Psychopg team

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



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



Copyright © Gabriella Albano and the Psychopg team

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



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



Copyright © Gabriella Albano and the Psycopg 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 Psycopg 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.



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



Copyright © Gabriella Albano and the Psycopg 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^{3,66,71,77,137} verwenden.



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



Copyright © Gabriella Albano and the Psycopg 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^{3,66,71,77,137} verwenden.
- Dazu gibt es eine Bibliothek für den Zugriff auf PostgreSQL, nämlich psycopg¹³⁵.

Konzeptuelle Modelle



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



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 Diagramme (ERDs) gezeichnet, die die Objekte und deren Beziehungen aus der realen Welt darstellen, die wir in der Datenbank modellieren wollen.

Konzeptuelle Modelle



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 Diagramme (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.

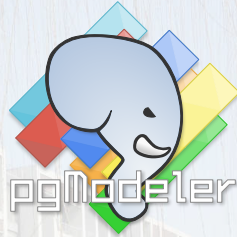
Konzeptuelle Modelle



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 Diagramme (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^{109,144}.

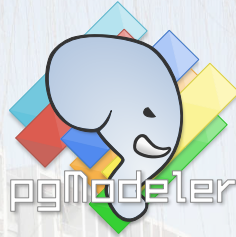
Logische Modelle



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.

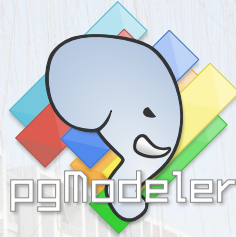
Logische Modelle



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.

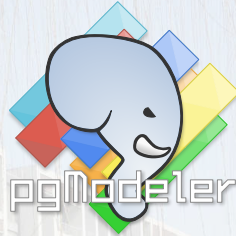
Logische Modelle



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.

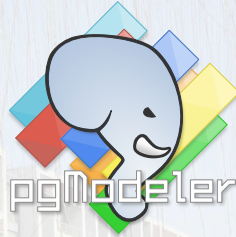
Logische Modelle



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.
- Sie können auch visuell mit Werkzeugen modelliert werden, als Diagramme die so ähnlich wie ERDs aussehen, die aber an eine bestimmte Datenbanktechnologie gebunden sind.

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.
- 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^{33,61} 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^{33,61} anfangen.
- Wenn Sie Microsoft Windows¹⁹ 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^{33,61} anfangen.
- Wenn Sie Microsoft Windows¹⁹ nutzen, vielleicht können Sie Ubuntu ja in einer virtuellen Maschine installieren.
- Ich empfehle jedoch ausdrücklich Linux zu lernen und zu nutzen.



Literatur



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 and 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 and 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 and 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 and 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.



谢谢你们！
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, 160**).
- [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, 160**).
- [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, 160**).
- [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, 160**).
- [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, 158**).
- [8] Richard Barker. *Case*Method: Entity Relationship Modelling (Oracle)*. 1. Aufl. Redwood City, CA, USA: Addison Wesley Longman Publishing Co., Inc., Jan. 1990. ISBN: **978-0-201-41696-1** (siehe S. **158**).
- [9] Daniel J. Barrett. *Efficient Linux at the Command Line*. Sebastopol, CA, USA: O'Reilly Media, Inc., Feb. 2022. ISBN: **978-1-0981-1340-7** (siehe S. **159, 161**).
- [10] 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, 160**).
- [11] 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, 159, 160**).

References II



- [12] Peter Belknap, John Beresiewicz, 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, 160).
- [13] Tim Berners-Lee. *Re: Qualifiers on Hypertext links*. . . Geneva, Switzerland: World Wide Web project, European Organization for Nuclear Research (CERN) und Newsgroups: alt.hypertext, 6. Aug. 1991. URL: <https://www.w3.org/People/Berners-Lee/1991/08/art-6484.txt> (besucht am 2025-02-05) (siehe S. 162).
- [14] Alex Berson. *Client/Server Architecture*. 2. Aufl. Computer Communications Series. New York, NY, USA: McGraw-Hill, 29. März 1996. ISBN: 978-0-07-005664-0 (siehe S. 158).
- [15] Joshua Bloch. *Effective Java*. Reading, MA, USA: Addison-Wesley Professional, Mai 2008. ISBN: 978-0-321-35668-0 (siehe S. 159).
- [16] Bernard Obeng Boateng. *Data Modeling with Microsoft Excel*. Birmingham, England, UK: Packt Publishing Ltd, Nov. 2023. ISBN: 978-1-80324-028-2 (siehe S. 160).
- [17] Grady Booch, James Rumbaugh und Ivar Jacobson. *The Unified Modeling Language Reference Manual*. 1. Aufl. Reading, MA, USA: Addison-Wesley Professional, Jan. 1999. ISBN: 978-0-201-57168-4 (siehe S. 162).
- [18] 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, 160).
- [19] 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, 160).
- [20] Ron Brash und Ganesh Naik. *Bash Cookbook*. Birmingham, England, UK: Packt Publishing Ltd, Juli 2018. ISBN: 978-1-78862-936-2 (siehe S. 158).
- [21] Tim Bray. *The JavaScript Object Notation (JSON) Data Interchange Format*. Request for Comments (RFC) 8259. Wilmington, DE, USA: Internet Engineering Task Force (IETF), Dez. 2017. URL: <https://www.ietf.org/rfc/rfc8259.txt> (besucht am 2025-02-05) (siehe S. 159).

References III



- [22] Ben Brumm. "A Guide to the Entity Relationship Diagram (ERD)". In: *Database Star*. Armadale, VIC, Australia: Elevated Online Services PTY Ltd., 30. Juli 2019–23. Dez. 2023. URL: <https://www.databasestar.com/entity-relationship-diagram> (besucht am 2025-03-29) (siehe S. 158).
- [23] 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).
- [24] Jason Cannon. *High Availability for the LAMP Stack*. Shelter Island, NY, USA: Manning Publications, Juni 2022 (siehe S. 25–31, 159, 161).
- [25] 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).
- [26] 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, 160, 161).
- [27] Peter Pin-Shan Chen. "Entity-Relationship Modeling: Historical Events, Future Trends, and Lessons Learned". In: *Software Pioneers: Contributions to Software Engineering*. Hrsg. von Manfred Broy und Ernst Denert. Berlin/Heidelberg, Germany: Springer-Verlag GmbH Germany, Feb. 2002, S. 296–310. doi:10.1007/978-3-642-59412-0_17. URL: http://bit.csc.lsu.edu/%7Echen/pdf/Chen_Pioneers.pdf (besucht am 2025-03-06) (siehe S. 158).
- [28] Peter Pin-Shan Chen. "The Entity-Relationship Model – Toward a Unified View of Data". *ACM Transactions on Database Systems (TODS)* 1(1):9–36, März 1976. New York, NY, USA: Association for Computing Machinery (ACM). ISSN: 0362-5915. doi:10.1145/320434.320440 (siehe S. 144, 158).
- [29] Peter Pin-Shan Chen. "The Entity-Relationship Model: Toward a Unified View of Data". In: *1st International Conference on Very Large Data Bases (VLDB'1975)*. 22.–24. Sep. 1975, Framingham, MA, USA. Hrsg. von Douglas S. Kerr. New York, NY, USA: Association for Computing Machinery (ACM), 1975, S. 173. ISBN: 978-1-4503-3920-9. doi:10.1145/1282480.1282492. See²⁸ for a more comprehensive introduction. (Siehe S. 158).

References IV



- [30] 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).
- [31] 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, 158).
- [32] 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, 159, 160).
- [33] 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, 161, 162).
- [34] Edgar Frank „Ted“ Codd. “A Relational Model of Data for Large Shared Data Banks”. *Communications of the ACM (CACM)* 13(6):377–387, Juni 1970. New York, NY, USA: Association for Computing Machinery (ACM). ISSN: 0001-0782. doi:10.1145/362384.362685. URL: <https://www.seas.upenn.edu/~zives/03f/cis550/codd.pdf> (besucht am 2025-01-05) (siehe S. 161).
- [35] 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).
- [36] *Database Language SQL*. Techn. Ber. ANSI X3.135-1986. Washington, D.C., USA: American National Standards Institute (ANSI), 1986 (siehe S. 161).
- [37] 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).
- [38] Matt David und Blake Barnhill. *How to Teach People SQL*. San Francisco, CA, USA: The Data School, Chart.io, Inc., 10. Dez. 2019–10. Apr. 2023. URL: <https://dataschool.com/how-to-teach-people-sql> (besucht am 2025-02-27) (siehe S. 161).
- [39] *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).
- [40] *Database Administrators*. New York, NY, USA: Stack Exchange Inc. URL: <https://dba.stackexchange.com> (besucht am 2025-02-27) (siehe S. 133).

References V



- [41] *Database Language SQL*. International Standard ISO 9075-1987. Geneva, Switzerland: International Organization for Standardization (ISO), 1987 (siehe S. 161).
- [42] Paul Deitel, Harvey Deitel und Abbey Deitel. *Internet & World Wide Web: How to Program*. 5. Aufl. Hoboken, NJ, USA: Pearson Education, Inc., Nov. 2011. ISBN: 978-0-13-299045-5 (siehe S. 162).
- [43] 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).
- [44] Slobodan Dmitrović. *Modern C for Absolute Beginners: A Friendly Introduction to the C Programming Language*. New York, NY, USA: Apress Media, LLC, März 2024. ISBN: 979-8-8688-0224-9 (siehe S. 158).
- [45] Pooyan Doozandeh und Frank E. Ritter. “Some Tips for Academic Writing and Using Microsoft Word”. *XRDS: Crossroads, The ACM Magazine for Students* 26(1):10–11, Herbst 2019. New York, NY, USA: Association for Computing Machinery (ACM). ISSN: 1528-4972. doi:10.1145/3351470 (siehe S. 160).
- [46] 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, 160).
- [47] *ECMAScript Language Specification*. Standard ECMA-262, 3rd Edition. Geneva, Switzerland: Ecma International, Dez. 1999. URL: https://ecma-international.org/wp-content/uploads/ECMA-262_3rd_edition_december_1999.pdf (besucht am 2024-12-15) (siehe S. 159).
- [48] 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).
- [49] Steve Fanning, Vasudev Narayanan, „flywire“, Olivier Hallot, Jean Hollis Weber, Jenna Sargent, Pulkit Krishna, Dan Lewis, Peter Schofield, Jochen Schiffrers, 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, 159).

References VI



- [50] 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, 160**).
- [51] 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)*, 9 5-9, 2022, hybrid/Sydney, NSW, Australia (siehe S. **41–45, 161**).
- [52] 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). Berlin/Heidelberg, 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, 159**).
- [53] 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**).
- [54] Michael Goodwin. *What is an API?* Armonk, NY, USA: International Business Machines Corporation (IBM), 9. Apr. 2024. URL: <https://www.ibm.com/topics/api> (besucht am 2024-12-12) (siehe S. **158**).
- [55] 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**).
- [56] Dawn Griffiths. *Excel Cookbook – Receipts for Mastering Microsoft Excel*. Sebastopol, CA, USA: O'Reilly Media, Inc., Mai 2024. ISBN: **978-1-0981-4332-9** (siehe S. **160**).
- [57] 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, 158**).

References VII



- [58] 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**, **161**).
- [59] 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**, **161**).
- [60] Michael Hausenblas. *Learning Modern Linux*. Sebastopol, CA, USA: O'Reilly Media, Inc., Apr. 2022. ISBN: **978-1-0981-0894-6** (siehe S. **159**).
- [61] 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**, **159**, **162**).
- [62] 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**).
- [63] 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**, **161**).
- [64] 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**).
- [65] 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**).
- [66] 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**, **161**).

References VIII



- [67] *Information Technology – Database Languages – SQL – Part 1: Framework (SQL/Framework), Part 1*. International Standard ISO/IEC 9075-1:2023(E), Sixth Edition, (ANSI X3.135). Geneva, Switzerland: International Organization for Standardization (ISO) und International Electrotechnical Commission (IEC), Juni 2023. URL: [https://standards.iso.org/ittf/PubliclyAvailableStandards/ISO_IEC_9075-1_2023_ed_6_-_id_76583_Publication_PDF_\(en\).zip](https://standards.iso.org/ittf/PubliclyAvailableStandards/ISO_IEC_9075-1_2023_ed_6_-_id_76583_Publication_PDF_(en).zip) (besucht am 2025-01-08). Consists of several parts, see <https://modern-sql.com/standard> for information where to obtain them. (Siehe S. 161).
- [68] 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).
- [69] Shannon Kempe und Paul Williams. *A Short History of the ER Diagram and Information Modeling*. Studio City, CA, USA: Dataversity Digital LLC, 25. Sep. 2012. URL: <https://www.dataversity.net/a-short-history-of-the-er-diagram-and-information-modeling> (besucht am 2025-03-06) (siehe S. 158).
- [70] 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).
- [71] 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).
- [72] 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).
- [73] 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, 160).
- [74] 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).
- [75] Jay LaCroix. *Mastering Ubuntu Server*. 4. Aufl. Birmingham, England, UK: Packt Publishing Ltd, Sep. 2022. ISBN: 978-1-80323-424-3 (siehe S. 161).

References IX



- [76] 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, 160**).
- [77] 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, 161**).
- [78] Marc-André Lemburg. *Python Database API Specification v2.0*. Python Enhancement Proposal (PEP) 249. Beaverton, OR, USA: Python Software Foundation (PSF), 12. Apr. 1999. URL: **<https://peps.python.org/pep-0249>** (besucht am 2025-02-02) (siehe S. **161**).
- [79] *LibreOffice – The Document Foundation*. Berlin, Germany: The Document Foundation, 2024. URL: **<https://www.libreoffice.org>** (besucht am 2024-12-12) (siehe S. **46–50, 159, 160**).
- [80] Gloria Lotha, Aakanksha Gaur, Erik Gregersen, Swati Chopra und William L. Hosch. "Client-Server Architecture". In: *Encyclopaedia Britannica*. Hrsg. von The Editors of Encyclopaedia Britannica. Chicago, IL, USA: Encyclopædia Britannica, Inc., 3. Jan. 2025. URL: **<https://www.britannica.com/technology/client-server-architecture>** (besucht am 2025-01-20) (siehe S. **158**).
- [81] Marc Loy, Patrick Niemeyer und Daniel Leuck. *Learning Java*. 5. Aufl. Sebastopol, CA, USA: O'Reilly Media, Inc., März 2020. ISBN: **978-1-4920-5627-0** (siehe S. **159**).
- [82] Mark Lutz. *Learning Python*. 6. Aufl. Sebastopol, CA, USA: O'Reilly Media, Inc., März 2025. ISBN: **978-1-0981-7130-8** (siehe S. **161**).
- [83] *MariaDB Server Documentation*. Milpitas, CA, USA: MariaDB, 2025. URL: **<https://mariadb.com/kb/en/documentation>** (besucht am 2025-04-24) (siehe S. **160**).
- [84] 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, 160**).
- [85] 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, 161**).
- [86] *Microsoft Word*. Redmond, WA, USA: Microsoft Corporation, 2024. URL: **<https://www.microsoft.com/en-us/microsoft-365/word>** (besucht am 2024-12-12) (siehe S. **160**).

References X



- [87] Cameron Newham und Bill Rosenblatt. *Learning the Bash Shell – Unix Shell Programming: Covers Bash 3.0*. 3. Aufl. Sebastopol, CA, USA: O'Reilly Media, Inc., 2005. ISBN: **978-0-596-00965-6** (siehe S. **158**).
- [88] 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, 160**).
- [89] **OMG® Unified Modeling Language® (OMG UML®)**. Version 2.5.1. OMG Document formal/2017-12-05. Milford, MA, USA: Object Management Group, Inc. (OMG), Dez. 2017. URL: <https://www.omg.org/spec/UML/2.5.1/PDF> (besucht am 2025-03-30) (siehe S. **162**).
- [90] "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**).
- [91] Robert Orfali, Dan Harkey und Jeri Edwards. *Client/Server Survival Guide*. 3. Aufl. Chichester, West Sussex, England, UK: John Wiley and Sons Ltd., 25. Jan. 1999. ISBN: **978-0-471-31615-2** (siehe S. **158**).
- [92] 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**).
- [93] 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**).
- [94] 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**).
- [95] 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 XI



- [96] Yasset Pérez-Riverol, Laurent Gatto, Rui Wang, Timo Sachsenberg, Julian Uszkoreit, Felipe da Veiga Leprevost, Christian Fufezan, Tobias Ternent, Stephen J. Eglén, Daniel S. Katz, Tom J. Pollard, Alexander Kononov, Robert M. Flight, Kai Blin und Juan Antonio Vizcaíno. "Ten Simple Rules for Taking Advantage of Git and GitHub". *PLOS Computational Biology* 12(7), 14. Juli 2016. San Francisco, CA, USA: Public Library of Science (PLOS). ISSN: **1553-7358**. doi:[10.1371/JOURNAL.PCBI.1004947](https://doi.org/10.1371/JOURNAL.PCBI.1004947) (siehe S. **159**).
- [97] 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, 160**).
- [98] 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**).
- [99] *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**).
- [100] *PostgreSQL Essentials: Leveling Up Your Data Work*. Sebastopol, CA, USA: O'Reilly Media, Inc., März 2024 (siehe S. **35–40, 78–81, 160**).
- [101] *Programming Languages – C, Working Document of SC22/WG14*. International Standard ISO/31EC9899:2017 C17 Ballot N2176. Geneva, Switzerland: International Organization for Standardization (ISO) und International Electrotechnical Commission (IEC), Nov. 2017. URL: <https://files.lhmouse.com/standards/ISO%20C%20N2176.pdf> (besucht am 2024-06-29) (siehe S. **158**).
- [102] 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**).
- [103] 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**).
- [104] Abhishek Ratan, Eric Chou, Pradeeban Kathiravelu und Dr. M.O. Faruque Sarker. *Python Network Programming*. Birmingham, England, UK: Packt Publishing Ltd, Jan. 2019. ISBN: **978-1-78883-546-6** (siehe S. **158**).

References XII



- [105] Federico Razzoli. *Mastering MariaDB*. Birmingham, England, UK: Packt Publishing Ltd, Sep. 2014. ISBN: **978-1-78398-154-0** (siehe S. **32–34, 160**).
- [106] 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), 2021, S. 1–8. ISSN: **2577-1647**. ISBN: **978-1-6654-3554-3**. doi:[10.1109/IECON48115.2021.9589382](https://doi.org/10.1109/IECON48115.2021.9589382) (siehe S. **25–31, 160**).
- [107] Mark Richards und Neal Ford. *Fundamentals of Software Architecture: An Engineering Approach*. Sebastopol, CA, USA: O'Reilly Media, Inc., Jan. 2020. ISBN: **978-1-4920-4345-4** (siehe S. **158**).
- [108] 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**).
- [109] 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 der Reihe Frontiers in Artificial Intelligence and Applications. Amsterdam, The Netherlands: IOS Press BV, 2015, S. 65–76. ISSN: **0922-6389**. ISBN: **978-1-61499-610-1**. doi:[10.3233/978-1-61499-611-8-65](https://doi.org/10.3233/978-1-61499-611-8-65) (siehe S. **93–96, 162**).
- [110] 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, 159, 160**).
- [111] Yuriy Shamshin. "Conceptual Database Model. Entity Relationship Diagram (ERD)". In: *Databases*. Riga, Latvia: ISMA University of Applied Sciences, Mai 2024. Kap. 04. URL: https://dbs.academy.lv/lection/dbs_LS04EN_erd.pdf (besucht am 2025-03-29) (siehe S. **158**).
- [112] 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**).
- [113] Ellen Siever, Stephen Figgins, Robert Love und Arnold Robbins. *Linux in a Nutshell*. 6. Aufl. Sebastopol, CA, USA: O'Reilly Media, Inc., Sep. 2009. ISBN: **978-0-596-15448-6** (siehe S. **159**).

References XIII



- [114] 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**).
- [115] Anna Skoulikari. *Learning Git*. Sebastopol, CA, USA: O'Reilly Media, Inc., Mai 2023. ISBN: **978-1-0981-3391-7** (siehe S. **159**).
- [116] John Miles Smith und Philip Yen-Tang Chang. "Optimizing the Performance of a Relational Algebra Database Interface". *Communications of the ACM (CACM)* 18(10):568–579, Okt. 1975. New York, NY, USA: Association for Computing Machinery (ACM). ISSN: **0001-0782**. doi:**10.1145/361020.361025** (siehe S. **161**).
- [117] *SQLite*. Charlotte, NC, USA: Hipp, Wyrick & Company, Inc. (Hwaci), 2025. URL: <https://sqlite.org> (besucht am 2025-04-24) (siehe S. **161**).
- [118] "SQL Commands". In: *PostgreSQL Documentation*. 17.4. The PostgreSQL Global Development Group (PGDG), 20. Feb. 2025. Kap. Part VI. Reference. URL: <https://www.postgresql.org/docs/17/sql-commands.html> (besucht am 2025-02-25) (siehe S. **161**).
- [119] "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**).
- [120] 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. **154, 161**).
- [121] 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 ¹²⁰ (siehe S. **130, 161**).
- [122] 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**).
- [123] 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, 161**).

References XIV



- [124] 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, 160**).
- [125] "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**).
- [126] *The JSON Data Interchange Syntax*. Standard ECMA-404, 2nd Edition. Geneva, Switzerland: Ecma International, Dez. 2017. URL: <https://ecma-international.org/publications-and-standards/standards/ecma-404> (besucht am 2024-12-15) (siehe S. **159**).
- [127] Linus Torvalds. "The Linux Edge". *Communications of the ACM (CACM)* 42(4):38–39, Apr. 1999. New York, NY, USA: Association for Computing Machinery (ACM). ISSN: **0001-0782**. doi:[10.1145/299157.299165](https://doi.org/10.1145/299157.299165) (siehe S. **159**).
- [128] 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**).
- [129] Mariot Tsitoara. *Beginning Git and GitHub: Version Control, Project Management and Teamwork for the New Developer*. New York, NY, USA: Apress Media, LLC, März 2024. ISBN: **979-8-8688-0215-7** (siehe S. **159, 162**).
- [130] 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, 159, 160**).
- [131] *UML Distilled: A Brief Guide to the Standard Object Modeling Language*. 3. Aufl. The Addison-Wesley Object Technology Series. Reading, MA, USA: Addison-Wesley Professional, Sep. 2003. ISBN: **978-0-321-19368-1** (siehe S. **162**).
- [132] *UML Notation Guide*. Version 1.1. Santa Clara, CA, USA: Rational Software Corporation, Redmond, WA, USA: Microsoft Corporation, Palo Alto, CA, USA: Hewlett-Packard Company, Redwood Shores, CA, USA: Oracle Corporation, Dallas, TX, USA: Sterling Software, Ottawa, ON, Canada: MCI Systemhouse Corporation, Blue Bell, PA, USA: Unisys Corporation, Blue Bell, PA, USA: ICON Computing, Santa Clara, CA, USA: IntelliCorp, Burlington, MA, USA: i-Logix, Armonk, NY, USA: International Business Machines Corporation (IBM), Kanata, ON, Canada: ObjectTime Limited, Chicago, IL, USA: Platinum Technology Inc., Boston, MA, USA: Ptech Inc., Orlando, FL, USA: Taskon A/S, Paoli, PA, USA: Reich Technologies und Paris, Île-de-France, France: Softeam, 1. Sep. 1997. URL: <https://web.cse.msu.edu/~cse870/Materials/uml-notation-guide-9-97.pdf> (besucht am 2025-03-30) (siehe S. **162**).

References XV



- [133] Sander van Vugt. *Linux Fundamentals*. 2. Aufl. Hoboken, NJ, USA: Pearson IT Certification, Juni 2022. ISBN: 978-0-13-792931-3 (siehe S. 159).
- [134] 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).
- [135] 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, 161).
- [136] Thomas Weise (汤卫思). *Databases*. Hefei, Anhui, China (中国安徽省合肥市): Hefei University (合肥大学), School of Artificial Intelligence and 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, 158, 159, 161).
- [137] Thomas Weise (汤卫思). *Programming with Python*. Hefei, Anhui, China (中国安徽省合肥市): Hefei University (合肥大学), School of Artificial Intelligence and 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, 161).
- [138] Matthew West. *Developing High Quality Data Models*. Version: 2.0, Issue: 2.1. London, England, UK: Shell International Limited und European Process Industries STEP Technical Liaison Executive (EPISTLE); Burlington, MA, USA/San Mateo, CA, USA: Morgan Kaufmann Publishers, 8. Dez. 1995–Dez. 2010. ISBN: 978-0-12-375107-2. URL: <https://www.researchgate.net/publication/286610894> (besucht am 2025-03-24). Edited by Julian Fowler (siehe S. 158).
- [139] *What is a Relational Database?* Armonk, NY, USA: International Business Machines Corporation (IBM), 20. Okt. 2021–12. Dez. 2024. URL: <https://www.ibm.com/think/topics/relational-databases> (besucht am 2025-01-05) (siehe S. 161).
- [140] 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, 160).
- [141] 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, 160).

References XVI



- [142] 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 (siehe S. 41–45, 161).
- [143] Kinza Yasar und Craig S. Mullins. *Definition: Database Management System (DBMS)*. Newton, MA, USA: TechTarget, Inc., Juni 2024. URL: <https://www.techtarget.com/searchdatamanagement/definition/database-management-system> (besucht am 2025-01-11) (siehe S. 158).
- [144] *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, 162).
- [145] Pavlo V. Zahorodko und Pavlo V. Merzlykin. "An Approach for Processing and Document Flow Automation for Microsoft Word and LibreOffice Writer File Formats". In: *4th Workshop for Young Scientists in Computer Science & Software Engineering (CS&SE@SW'2021)*. 18. Dez. 2021, Virtual Event and Kryvyi Rih, Ukraine. Hrsg. von Arnold E. Kiv, Serhiy O. Semerikov, Vladimir N. Soloviev und Andrii M. Striuk. Bd. 3077 der Reihe CEUR Workshop Proceedings ([CEUR-WS.org](http://ceur-ws.org)). Aachen, Nordrhein-Westfalen, Germany: CEUR-WS Team, Rheinisch-Westfälische Technische Hochschule (RWTH) Aachen, 2022, S. 66–82. ISSN: 1613-0073. URL: <https://ceur-ws.org/Vol-3077/paper12.pdf> (besucht am 2025-10-04) (siehe S. 159, 160).
- [146] Giorgio Zarrelli. *Mastering Bash*. Birmingham, England, UK: Packt Publishing Ltd, Juni 2017. ISBN: 978-1-78439-687-9 (siehe S. 158).

Glossary (in English) I



API An *Application Programming Interface* is a set of rules or protocols that enables one software application or component to use or communicate with another⁵⁴.

Bash is a the shell used under Ubuntu Linux, i.e., the program that „runs“ in the terminal and interprets your commands, allowing you to start and interact with other programs^{20,87,146}. Learn more at <https://www.gnu.org/software/bash>.

C is a programming language, which is very successful in system programming situations^{44,101}.

client In a client-server architecture, the client is a device or process that requests a service from the server. It initiates the communication with the server, sends a request, and receives the response with the result of the request. Typical examples for clients are web browsers in the internet as well as clients for DBMSes, such as psql.

client-server architecture is a system design where a central server receives requests from one or multiple clients^{14,80,91,104,107}. These requests and responses are usually sent over network connections. A typical example for such a system is the World Wide Web (WWW), where web servers host websites and make them available to web browsers, the clients. Another typical example is the structure of database (DB) software, where a central server, the DBMS, offers access to the DB to the different clients. Here, the client can be some terminal software shipping with the DBMS, such as psql, or the different applications that access the DBs.

DB A *database* is an organized collection of structured information or data, typically stored electronically in a computer system. Databases are discussed in our book *Databases*¹³⁶.

DB2 developed by IBM is one of the older and popular relational DBMSes^{7,31,57}.

DBMS A *database management system* is the software layer located between the user or application and the DB. The DBMS allows the user/application to create, read, write, update, delete, and otherwise manipulate the data in the DB¹⁴³.

ERD Entity relationship diagrams show the relationships between objects, e.g., between the tables in a DB and how they reference each other^{8,22,27–29,69,111,138}.

Glossary (in English) II



- Git** is a distributed Version Control Systems (VCS) which allows multiple users to work on the same code while preserving the history of the code changes^{115,129}. Learn more at <https://git-scm.com>.
- GitHub** is a website where software projects can be hosted and managed via the Git VCS^{96,129}. Learn more at <https://github.com>.
- GUI** graphical user interface
- Java** is another very successful programming language, with roots in the C family of languages^{15,81}.
- JavaScript** JavaScript is the predominant programming language used in websites to develop interactive contents for display in browsers⁴⁷.
- JSON** *JavaScript Object Notation* is a data interchange format^{21,126} based on JavaScript⁴⁷ syntax.
- LAMP Stack** A system setup for web applications: Linux, Apache (a web server), MySQL, and the server-side scripting language PHP^{24,61}.
- LibreOffice** is on open source office suite^{52,79,110} which is a good and free alternative to Microsoft Office. It offers software such as LibreOffice Writer, LibreOffice Calc, and LibreOffice Base. See¹³⁶ for more information and installation instructions.
- LibreOffice Base** is a DBMS that can work on stand-alone files but also connect to other popular relational databases^{49,110}. It is part of LibreOffice^{52,79,110} and has functionality that is comparable to Microsoft Access^{11,32,130}.
- LibreOffice Calc** is a spreadsheet software that allows you to arrange and perform calculations with data in a tabular grid. It is a free and open source spread sheet software^{79,110}, i.e., an alternative to Microsoft Excel. It is part of LibreOffice^{52,79,110}.
- LibreOffice Writer** is a free and open source text writing program¹⁴⁵ and part of LibreOffice^{52,79,110}. It is a good alternative to Microsoft Word.
- Linux** is the leading open source operating system, i.e., a free alternative for Microsoft Windows^{9,60,113,127,133}. We recommend using it for this course, for software development, and for research. Learn more at <https://www.linux.org>. Its variant Ubuntu is particularly easy to use and install.

Glossary (in English) III



- MariaDB** An open source relational database management system that has forked off from MySQL^{4,5,10,46,83,105}. See <https://mariadb.org> for more information.
- Microsoft Access** is a DBMS that can work on DBs stored in single, stand-alone files but also connect to other popular relational databases^{11,32,84,130}. It is part of Microsoft Office. A free and open source alternative to this commercial software is LibreOffice Base.
- Microsoft Excel** is a spreadsheet program that allows users to store, organize, manipulate, and calculate data in tabular structures^{16,56,76}. It is part of Microsoft Office. A free alternative to this commercial software is LibreOffice Calc^{79,110}.
- Microsoft Office** is a commercial suite of office software, including Microsoft Excel, Microsoft Word, and Microsoft Access⁷⁶. LibreOffice is a free and open source alternative.
- Microsoft SQL Server** The Microsoft SQL Server is a successful commercial relational/SQL-based DBMS^{1,97,140}. Learn more at <https://www.microsoft.com/sql-server> and <https://learn.microsoft.com/en-us/sql>.
- Microsoft Windows** is a commercial proprietary operating system¹⁹. It is widely spread, but we recommend using a Linux variant such as Ubuntu for software development and for our course. Learn more at <https://www.microsoft.com/windows>.
- Microsoft Word** is one of the leading text writing programs^{45,86,145} and part of Microsoft Office. A free alternative to this commercial software is the LibreOffice Writer.
- MySQL** An open source relational database management system^{18,46,106,124,141}. MySQL is famous for its use in the LAMP Stack. See <https://www.mysql.com> for more information.
- Oracle Database** The Oracle Database was the first commercial SQL-based relational database²⁶. It is still a highly successful proprietary product with many features^{12,73}. Learn more at <https://www.oracle.com/database>.
- PgModeler** the PostgreSQL DB modeler is a tool that allows for graphical modeling of logical schemas for DBs using an entity relationship diagram (ERD)-like notation². Learn more at <https://pgmodeler.io>.
- PostgreSQL** An open source object-relational DBMS^{50,88,100,124}. See <https://postgresql.org> for more information.

Glossary (in English) IV



psql is the client program used to access the PostgreSQL DBMS server.

psycopg or, more exactly, **psycopg 3**, is the most popular PostgreSQL adapter for Python, implementing the Python DB API 2.0 specification⁷⁸. Learn more at <https://www.psycopg.org>¹³⁵.

Python The Python programming language^{66,77,82,137}, i.e., what you will learn about in our book¹³⁷. Learn more at <https://python.org>.

relational database A relational DB is a database that organizes data into rows (tuples, records) and columns (attributes), which collectively form tables (relations) where the data points are related to each other^{34,58,59,116,123,136,139}.

server In a client-server architecture, the server is a process that fulfills the requests of the clients. It usually waits for incoming communication carrying the requests from the clients. For each request, it takes the necessary actions, performs the required computations, and then sends a response with the result of the request. Typical examples for servers are web servers²⁴ in the internet as well as DBMSes. It is also common to refer to the computer running the server processes as server as well, i.e., to call it the „server computer“⁷⁵.

SQL The *Structured Query Language* is basically a programming language for querying and manipulating relational databases^{26,36,38,41,67,85,118,120,121,123}. It is understood by many DBMSes. You find the SQL commands supported by PostgreSQL in the reference¹¹⁸.

SQLite is an relational DBMS which runs as in-process library that works directly on files as opposed to the client-server architecture used by other common DBMSes. It is the most wide-spread SQL-based DB in use today, installed in nearly every smartphone, computer, web browser, television, and automobile^{26,51,63,142}. Learn more at <https://sqlite.org>¹¹⁷.

terminal A terminal is a text-based window where you can enter commands and execute them^{9,33}. Knowing what a terminal is and how to use it is very essential in any programming- or system administration-related task. If you want to open a terminal under Microsoft Windows, you can Druck auf  + **R**, dann Schreiben von `cmd`, dann Druck auf . Under Ubuntu Linux, **Ctrl** + **Alt** + **T** opens a terminal, which then runs a Bash shell inside.

Glossary (in English) V



Ubuntu is a variant of the open source operating system Linux^{33,61}. We recommend that you use this operating system to follow this class, for software development, and for research. Learn more at <https://ubuntu.com>. If you are in China, you can download it from <https://mirrors.ustc.edu.cn/ubuntu-releases>.

UML The Unified Modeling Language (UML) is a graphical language for visualizing, specifying, constructing, and documenting the artifacts of distributed object systems^{17,89,131,132}

VCS A *Version Control System* is a software which allows you to manage and preserve the historical development of your program code¹²⁹. A distributed VCS allows multiple users to work on the same code and upload their changes to the server, which then preserves the change history. The most popular distributed VCS is Git.

WWW World Wide Web^{13,42}

yEd is a graph editor for high-quality graph-based diagrams^{109,144}, suitable to draw, e.g., technology-independent ERDs, control flow charts, or Unified Modeling Language (UML) class diagrams. An online version of the editor is available at <https://www.yworks.com/yed-live>. Learn more at <https://www.yworks.com/products/yed>.