

COSC 3380 Spring 2024

Database Systems

M & W 4:00 to 5:30 PM

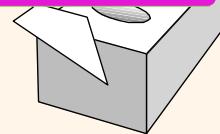
Prof. **Victoria Hilford**

PLEASE TURN your webcam ON (must have)

NO CHATTING during LECTURE

VH, UNHIDE SET 1 Sections 2 & 3

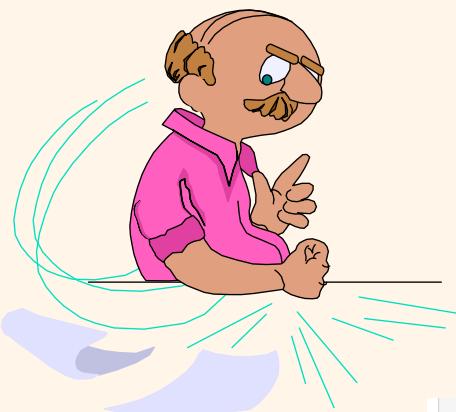
From 4:00 to 4:15 PM – 15 minutes.



COSC 3380

4 to 5:30

**PLEASE
LOG IN
CANVAS**



⋮	📎 COSC 3380 Syllabus.pdf	⬇️	✓	⋮
⋮	📎 COSC 3380 Detailed Syllabus.pdf	⬇️	✓	⋮
⋮	📎 zybook.jpg	⬇️	✓	⋮

Please close all other windows.

On the syllabus

COURSE TITLE/SECTION: COSC3380 - Database Systems Spring 2024 (23578)

TIME: MW 4 – 5:30 PM

FACULTY: Dr. Victoria Hilford **OFFICE HOURS: MW (12–12:30 PM) & by appointment**
Synchronous ONLINE (attendance required, no recordings)

E-mail: CANVAS Email, TEAMS

Phone: N/A **FAX:** N/A

The information contained in this class syllabus is subject to change without notice. Students are expected to be aware of any additional course policies presented by the instructor during the course.

This course uses zyBooks, TEAMS and CANVAS.

LAPTOP REQUIREMENT with A FUNCTIONING CAMERA

All students must have a laptop with a functioning camera for the duration of the class. In purchasing a laptop, you'll want to consider these issues:

Built-in wireless capability to access:

<https://uh.edu/canvas.com> and <http://www.zybooks.com/>

Enter code **UHCOSC3380HilfordSpring2024**;

Click Subscribe (must use your .uh.edu email address).

Cost \$89.

Access to a webcam is required for students participating in this course.

Webcams must be turned on during exams to ensure the academic integrity of exam administration.

On the syllabus

This course will include the following topical (content) areas:

1. OVERVIEW OF DATABASE SYSTEMS
2. INTRODUCTION TO DATABASE DESIGN
3. THE RELATIONAL MODEL
4. RELATIONAL ALGEBRA AND CALCULUS
5. SQL: QUERIES, CONSTRAINTS, TRIGGERS
6. DATABASE APPLICATION DEVELOPMENT
7. INTERNET APPLICATIONS
8. OVERVIEW OF STORAGE AND INDEXING
9. STORING DATA: DISKS AND FILES
10. TREE-STRUCTURED INDEXING
11. HASH-BASED INDEXING
12. OVERVIEW OF QUERY EVALUATION
13. EXTERNAL SORTING
14. EVALUATING RELATIONAL OPERATORS
15. OVERVIEW OF TRANSACTION MANAGEMENT
16. CONCURRENCY CONTROL
17. CRASH RECOVERY
18. SCHEMA REFINEMENT AND NORMAL FORMS
19. SECURITY AND AUTHORIZATION

On the syllabus

Evaluation and Grading:

Must be in TEAMS to take any graded assignment. Any attempt to take a graded assignment not being in TEAMS will result in a grade of ZERO for that graded assignment.

Webcams are required to take EXAMs.

30% [zyBook](#)
20% [Class Participation](#)
50% [EXAMs \(4\)](#)

Grading Scale:

93 or higher A
90 – below 93 A-
87 – below 90 B+
83 – below 87 B
80 – below 83 B-
77 – below 80 C+
73 – below 77 C
70 – below 73 C-
67 – below 70 D+
63 – below 67 D
60 – below 63 D-
0 – below 60 F

On the syllabus

Presence in Class

Your presence in class each session is required, class is synchronous online.

Classes will not be recorded by the instructor.

On the syllabus

Excused Absence Policy

Regular class attendance, participation, and engagement in coursework are important contributors to student success. Absences may be excused as provided in the University of Houston [Undergraduate Excused Absence Policy](#) and [Graduate Excused Absence Policy](#) for reasons including: medical

illness of student or close relative, death of a close family member, legal or government proceeding that a student is obligated to attend, recognized professional and educational activities where the student is presenting, and University-sponsored activity or athletic competition. Under these policies, students with excused absences will be provided with an opportunity to make up any quiz, exam or other work that contributes to the course grade or a satisfactory alternative. Please read the full policy for details regarding reasons for excused absences, the approval process, and extended absences. Additional policies address absences related to [military service](#), [religious holy days](#), [pregnancy and related conditions](#), and [disability](#).

Please remember that this is a synchronous online class. TEAMS and CANVAS can be accessed from any mobile device. You need to be available just the 1 hour and 15 minutes (from 4:00 to 5:15 PM).

Notification of all excused absences must be sent to the instructor before the excused class.

EXCUSSED ABSENCES MUST BE AQUIRED BEFORE THE EXCUSED

On the syllabus

Recording of Class

Students may not record all or part of class, livestream all or part of class, or make/distribute screen captures, without advanced written consent of the instructor. If you have or think you may have a disability such that you need to record class-related activities, please contact the Justin Dart, Jr. Student Accessibility Center. If you have accommodation to record class-related activities, those recordings may not be shared with any other student, whether in this course or not, or with any other person or on any other platform. Failure to comply with requirements regarding recordings will result in a disciplinary referral to the Dean of Students Office and may result in disciplinary action.

On the syllabus

Academic Honesty Policy

High ethical standards are critical to the integrity of any institution, and bear directly on the ultimate value of conferred degrees. All UH community members are expected to contribute to an atmosphere of the highest possible ethical standards. Maintaining such an atmosphere requires that any instances of academic dishonesty be recognized and addressed. The [UH Academic Honesty Policy](#) is designed to handle those instances with fairness to all parties involved: the students, the instructors, and the University itself. All students and faculty of the University of Houston are responsible for being familiar with this policy.

VH, Victoria Hilford

Office Hours:

M & W 12:00 to 12:30 PM

Online TEAMS

TA, Fernando Ramirez (A - H)

Office Hours:

M & W 2:45 to 3:45 PM

Online TEAMS

TA, Alvaro Urtaza (I - P)

Office Hours:

M & W 2:45 to 3:45 PM

Online TEAMS

TA, Jordan Yu (Q - Z)

Office Hours:

T & Th 2:30 to 3:30 PM

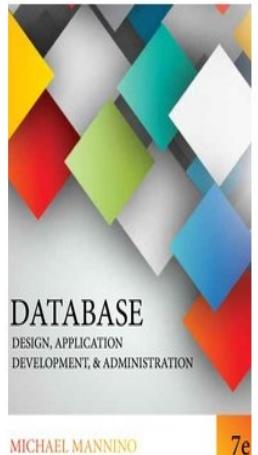
Online TEAMS

TEAMS CHAT

The screenshot shows the Microsoft Teams calendar interface for January 2024. The left sidebar includes links for Activity, Chat, Teams, Assignments, Calls, OneDrive, Shifts, People, and more. The main view displays a weekly grid from Monday to Friday. A specific meeting is scheduled for Tuesday, January 16, from 3:50 PM to 5:15 PM. The meeting details are as follows:

COSC 3380 23578 M & W 4 to 5:30 PM
LECTURE (in TEAMS from 3:50 to 5:15 PM)
Microsoft Teams Meeting
Hilford, Victoria

You may not use TEAMS or CANVAS Class Roll to create outside groups



Database Design, Application Development & Administration

by Mannino

Publisher: Chicago Business Press

Print ISBN: 9781948426008, 1948426005

eText ISBN: 9781948426060, 1948426064

Edition: 7th

Copyright year: 2019

DATABASE
DESIGN, APPLICATION
DEVELOPMENT, & ADMINISTRATION

MICHAEL MANNINO

7e

2019



Database Systems: Design, Implementation, & Management , 13t Edition

Carlos Coronel; Steven Morris
ISBN-10: 1-337-62790-9
ISBN-13: 978-1-337-62790-0

[View Sample Now](#)

Databases Illuminated - Access - 3rd edition

ISBN13: 9781284077094
ISBN10: 1284077098

USED
\$79.46

List price: \$105.95

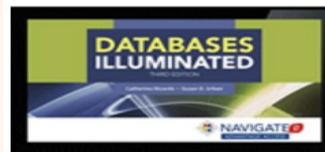
All of our used books are 100% hand-inspected and guaranteed to be in great condition.

FREE Shipping on \$25+

Ships tomorrow

Easy 30-Day Returns

Ships directly from us



by Catherine M. Ricardo

Cover type: Access Code
Edition: 3RD 17
Copyright: 2017
Publisher: Jones & Bartlett Publishers
Published: 2017
International: No

SHOP US WITH CONFIDENCE

4.6 ★★★★
Google
Customer Reviews

2017

Database Management Systems



Raghu Ramakrishnan, Johannes Gehrke

McGraw-Hill, 2003 - Database management - 1065 pages

★★★★★

0 Reviews

G+

TABLE OF CONTENTS Preface Pt. I Foundations 1 1 Overview of Database Systems 3 2 Introduction to Database Design 25 3 The Relational Model 57 4 Relational Algebra and Calculus 100 5 SQL: Queries, Constraints, Triggers 130 Pt. II Application Development 183 6 Database Application Development 185 7 Internet Applications 220 Pt. III Storage and Indexing 273 8 Overview of Storage and Indexing 273 9 Storing Data: Disks and Files 304 10 Tree-Structured Indexing 338 11 Hash-Based Indexing 370 Pt. IV Query Evaluation 391 12 Overview of Query Evaluation 393 13 External Sorting 421 14 Evaluating Relational Operators 439 15 A Typical Relational Query Optimizer 478 Pt. V Transaction Management 517 16 Overview of Transaction Management 519 17 Concurrency Control 549 18 Crash Recovery 579 Pt. VI Database Design and Tuning 603 19 Schema Refinement and Normal Forms 605 20 Physical Database Design

More »

2021

????

Same fundamentals

2003





Welcome to your class zyBook

Instructions for your students

- Students will access zyBooks directly.
- Students will access zyBooks through links in an LMS (Blackboard, Canvas, etc.)

Please provide the following instructions to your students. Copy into your syllabus, discussion board, etc.

[Copy instructions to clipboard](#)

1. Sign in or create an account at learn.zybooks.com
2. Enter zyBook code

UHCOSC3380HilfordSpring2024

3. Subscribe

A subscription is **\$89**. Students may begin subscribing on Jan 01, 2024 and the cutoff to subscribe is Apr 29, 2024. Subscriptions will last until May 29, 2024.

**Must register with ZyBook.
Name in CANVAS and ZyBook must match**

[Home](#)[Announcements](#)[Assignments](#)[Discussions](#)[Grades](#)[People](#)[Pages](#)[Files](#)

Detailed Syllabus

DATE		
01.17.2024 (1 - We)	ZyBook SET 1 - 1	Overview CANVAS and ZyBook Set 1 Lecture 1 Introduction
01.22.2024 (2 - Mo)	ZyBook SET 1 - 2	Set 1 LECTURE 2 DATA MODELING - WHAT - ERIC
01.24.2024 (3 - We)	ZyBook SET 1 - 3	Set 1 LECTURE 3 DATA MODELING - HOW - RELATIONAL
01.29.2024 (4 - Mo)	ZyBook SET 1 - 4	Set 1 LECTURE 4 ERD to RELATIONAL
01.31.2024 (5 - We)	ZyBook SET 1 - 5	Set 1 LECTURE 5 NORMALIZATION

CANVAS Structure.

► COSC 3380 SPRING 2024 M & W 4:00 TO 5:30 PM



► EXAMS [50 points]



► ZYBOOK [30 points]



► CLASS PARTICIPATION [20 points]



► LECTURES





H 2024 Spring 1

Search for Assignment

+ Group

+ Assignment



Home

Announcements

Assignments

Discussions

Grades

People

Pages

Files

Syllabus

Outcomes



Rubrics

Quizzes

Modules

Collaborations

EXAMS [50 points]

50% of Total

No assignments in this group

DATABASE SYSTEMS - ZYBOOK [30 points]

30% of Total



SET 1: Sections 1 - 6 [ZyBook] (100 points) (complete it anytime before EXAM REVIEW class)

DATABASE SYSTEMS - ZYBOOK [30 points] Module | Not available until Jan 17 at 5:30pm | Due Feb 7 at 4pm 100 pts



SET 2: Sections 8 - 11 [ZyBook] (100 points) (complete it any time before EXAM REVIEW class)

DATABASE SYSTEMS - ZYBOOK [30 points] Module | Not available until Feb 12 at 5:30pm | Due Mar 4 at 4pm 100 pts



SET 3: Sections 13 - 16 [ZyBook] (100 points) (complete it any time before EXAM REVIEW class)

DATABASE SYSTEMS - ZYBOOK [30 points] Module | Not available until Mar 6 at 5:30pm | Due Apr 3 at 4pm 100 pts



SET 4: Sections 18 - 21 [ZyBook] (100 points) (complete it any time before EXAM REVIEW class)

DATABASE SYSTEMS - ZYBOOK [30 points] Module | Not available until Apr 8 at 5:30pm | Due Apr 24 at 4pm 100 pts



CANVAS EMAI or TEAMS Chat

The screenshot shows the Canvas sidebar on the left and the Microsoft Teams Chat interface on the right. The sidebar includes links for Account, Dashboard, Courses, Calendar, and Inbox, with the Inbox item highlighted by a blue rectangle. The Teams interface shows a header with a user icon, search bar, and navigation controls. Below the header are sections for Activity, Chat, and a message input field labeled 'To: Enter name, email, group or tag'.

Please use CANVAS – MUST LOGIN) email to communicate with
your TA (Teaching Assistants):
Process:

1. Email your TA(A through H – Fernando; I through P – Alvaro; Q through Z – Jordan).

SET 1

01.17.2024 (1 - We)	ZyBook SET 1 - 1	Overview CANVAS and ZyBook Set 1 Lecture 1 Introduction
01.22.2024 (2 - Mo)	ZyBook SET 1 - 2	Set 1 LECTURE 2 DATA MODELING - WHAT - ERD MODEL
01.24.2024 (3 - We)	ZyBook SET 1 - 3	Set 1 LECTURE 3 DATA MODELING - HOW - RELATIONAL MODEL
01.29.2024 (4 - Mo)	ZyBook SET 1 - 4	Set 1 LECTURE 4 ERD to RELATIONAL
01.31.2024 (5 - We)	ZyBook SET 1 - 5	Set 1 LECTURE 5 NORMALIZATION
02.05.2024 (6 - Mo)		EXAM 1 Practice (PART of 20 points)
02.07.2024 (7 - We)	TA Download ZyBook SET 1 Sections (4 PM) (PART of 30 points)	EXAM 1 Review (PART of 20 points)
02.12.2024 (8 - Mo)		EXAM 1 (PART of 50 points)

SET 2

02.14.2024 (9 - We)	ZyBook SET 2 - 1	Set 2 LECTURE 6 SQL 1
02.19.2024 (10 - Mo)	ZyBook SET 2 - 2	Set 2 LECTURE 7 SQL II
02.21.2024 (11 - We)	ZyBook SET 2 - 3	Set 2 LECTURE 10 APPLICATIONS
02.26.2024 (12 - Mo)	ZyBook SET 2 - 4	Set 2 LECTURE 11 WEB APPLICATIONS
02.28.2024 (13 - We)		EXAM 2 Practice (PART of 20 points)
03.04.2024 (14 - Mo)	TA Download ZyBook SET 2 Sections (4 PM) (PART of 30 points)	EXAM 2 Review (PART of 20 points)
03.06.2024 (15 - We)		EXAM 2 (PART of 50 points)

SET 3

03.18.2024 (16 - Mo)	ZyBook SET 3 - 1	Set 3 LECTURE 12 STORAGE I
03.20.2024 (17 - We)	ZyBook SET 3 - 2	Set 3 LECTURE 13 STORAGE II
03.25.2024 (18 - Mo)	ZyBook SET 3 - 3	Set 3 LECTURE 14 STORAGE III B TREE INDEX
03.27.2024 (19 - We)	ZyBook SET 3 - 4	Set 3 LECTURE 15 STORAGE IV HASH INDEX
04.01.2024 (20 - Mo)		EXAM 3 Practice (PART of 20 points)
04.03.2024 (21 - We)	TA Download ZyBook SET 3 Sections (4 PM) (PART of 30 points)	EXAM 3 Review (PART of 20 points)
04.08.2024 (22 - Mo)		EXAM 3 (PART of 50 points)

SET 4

04.10.2024 (23 – We)	ZyBook SET 4 - 1	Set 4 LECTURE 16 TRANSACTIONS
04.15.2024 (24 – Mo)	ZyBook SET 4 - 2	Set 4 LECTURE 17 CONCURRENCY CONTROL
04.17.2024 (25 – We)	ZyBook SET 4 - 3	Set 4 LECTURE 18 CRASH RECOVERY LECTURE 19 SECURITY and AUTHORIZATION
04.22.2024 (26 – Mo)		EXAM 4 Practice (PART of 20 points)
04.24.2024 (27 – We)	TA Download ZyBook SET 4 Sections (4 PM) (PART of 30 points)	EXAM 4 Review (PART of 20 points)
04.29.2024 (28 – Mo)		EXAM 4 (PART of 50 points)

[Home](#)[Announcements](#)[Assignments](#)[Discussions](#)[Grades](#)[People](#)[Pages](#)[Files](#)

Detailed Syllabus

DATE		
01.17.2024 (1 - We)	ZyBook SET 1 - 1	Overview CANVAS and ZyBook Set 1 Lecture 1 Introduction
01.22.2024 (2 - Mo)	ZyBook SET 1 - 2	Set 1 LECTURE 2 DATA MODELING - WHAT - ERIC
01.24.2024 (3 - We)	ZyBook SET 1 - 3	Set 1 LECTURE 3 DATA MODELING - HOW - RELATIONAL
01.29.2024 (4 - Mo)	ZyBook SET 1 - 4	Set 1 LECTURE 4 ERD to RELATIONAL
01.31.2024 (5 - We)	ZyBook SET 1 - 5	Set 1 LECTURE 5 NORMALIZATION



Welcome to your class zyBook

Instructions for your students

- Students will access zyBooks directly.
- Students will access zyBooks through links in an LMS (Blackboard, Canvas, etc.)

Please provide the following instructions to your students. Copy into your syllabus, discussion board, etc.

[Copy instructions to clipboard](#)

1. Sign in or create an account at learn.zybooks.com
2. Enter zyBook code

UHCOSC3380HilfordSpring2024

3. Subscribe

A subscription is **\$89**. Students may begin subscribing on Jan 01, 2024 and the cutoff to subscribe is Apr 29, 2024. Subscriptions will last until May 29, 2024.

**Must register with ZyBook.
Name in CANVAS and ZyBook must match**

<input type="checkbox"/> 1. SET 1		Empty	▼
<input type="checkbox"/> 2. SET 1 - 1:INTRODUCTION	Hidden	0%	0% ▼
<input type="checkbox"/> 3. SET 1 - 2:DATA MODELING - WHAT - ERD MODEL	Hidden	0%	0% ▼
<input type="checkbox"/> 4. SET 1 - 3:DATA MODELING HOW - RELATIONAL MODELS	Hidden	0%	0% ▼
<input type="checkbox"/> 5. SET 1 - 4:ERD to RELATIONAL	Hidden	0%	0% ▼
<input type="checkbox"/> 6. SET 1 - 5:NORMALIZATION	Hidden	0%	0% ▼
<input type="checkbox"/> 7. SET 2		Empty	▼
<input type="checkbox"/> 8. SET 2 - 1:SQL I	Hidden	0%	0% ▼
<input type="checkbox"/> 9. SET 2 - 2:SQL II	Hidden	0%	0% ▼
<input type="checkbox"/> 10. SET 2 - 3:APPLICATIONS	Hidden	0%	0% ▼
<input type="checkbox"/> 11. SET 2 - 4:WEB APPLICATIONS	Hidden	0%	0% ▼
<input type="checkbox"/> 12. SET 3		Empty	▼
<input type="checkbox"/> 13. SET 3 - 1:STORAGE I	Hidden	0%	0% ▼
<input type="checkbox"/> 14. SET 3 - 2:STORAGE II	Hidden	0%	0% ▼
<input type="checkbox"/> 15. SET 3 - 3:STORAGE III B TREE INDEX	Hidden	0%	0% ▼
<input type="checkbox"/> 16. SET 3 - 4:STORAGE IV HASH INDEX	Hidden	0%	0% ▼
<input type="checkbox"/> 17. SET 4		Empty	▼
<input type="checkbox"/> 18. SET 4 - 1:TRANSACTIONS	Hidden	0%	0% ▼
<input type="checkbox"/> 19. SET 4 - 2:CONCURRENCY CONTROL	Hidden	0%	0% ▼
<input type="checkbox"/> 20. SET 4 - 3:CRASH RECOVERY	Hidden	0%	0% ▼
<input type="checkbox"/> 21. SET 4 - 4:SECURITY and AUTHORIZATION		Empty	▼
<input type="checkbox"/> 22. ADDITIONAL MATERIALS I - Database Architectures	0%	0%	▼
<input type="checkbox"/> 23. ADDITIONAL MATERIALS II - NoSQL Databases	0%	0%	▼
<input type="checkbox"/> 24. ADDITIONAL MATERIALS III - MySQL DBMS		0%	▼

Mapping ZyBook SET 1 to the my Lectures

Table of contents 

About this material

1. SET 1

Overview BB and ZyBook

2. SET 1 - 1: INTRODUCTION

Lecture 1 Introduction

3. SET 1 - 2: DATA MODELING - WHAT - ERD MODEL

LECTURE 2 DATA MODELING - WHAT - ERD MODEL

4. SET 1 - 3: DATA MODELING HOW - RELATIONAL MODELS

LECTURE 3 DATA MODELING HOW - RELATIONAL MODELS

5. SET 1 - 4: ERD to RELATIONAL

LECTURE 4 ERD to RELATIONAL

6. SET 1 - 5: NORMALIZATION

LECTURE 5 NORMALIZATION

Same for SET 2, SET 3, SET 4

TA, Fernando Ramirez (A – L).

TA, Alvaro Urtaza (M – Z).

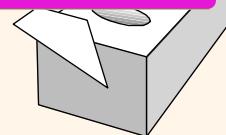
**Please compare CANVAS vs. TEAMS Attendance.
Print screens of students in CANVAS but not in the TEAMS meeting.
(1.17.2024 Attendance X missing LastName.docx)**

[Home](#)[Announcements](#)[Assignments](#)[Discussions](#)[Grades](#)[People](#)[Pages](#)[Files](#)

Detailed Syllabus

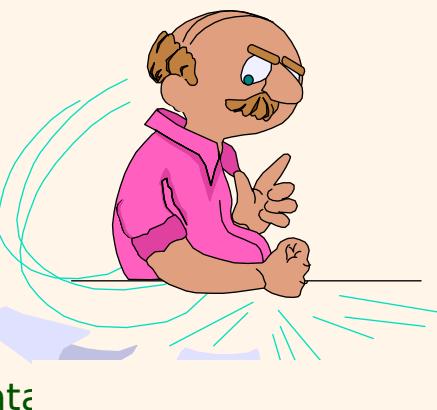
DATE		
01.17.2024 (1 - We)	ZyBook SET 1 - 1	Overview CANVAS and ZyBook Set 1 Lecture 1 Introduction
01.22.2024 (2 - Mo)	ZyBook SET 1 - 2	LECTURE 2 DATA MODELING - WHAT - ERIC
01.24.2024 (3 - We)	ZyBook SET 1 - 3	LECTURE 3 DATA MODELING - HOW - RELATIONAL
01.29.2024 (4 - Mo)	ZyBook SET 1 - 4	LECTURE 4 ERD to RELATIONAL
01.31.2024 (5 - We)	ZyBook SET 1 - 5	LECTURE 5 NORMALIZATION

From 4:15 to 5:05 PM – 50 minutes.

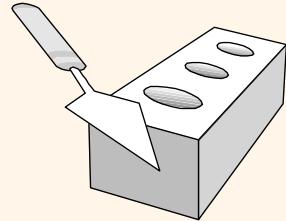


Lecture 1

Introduction DataBase Management Systems (DBMS)



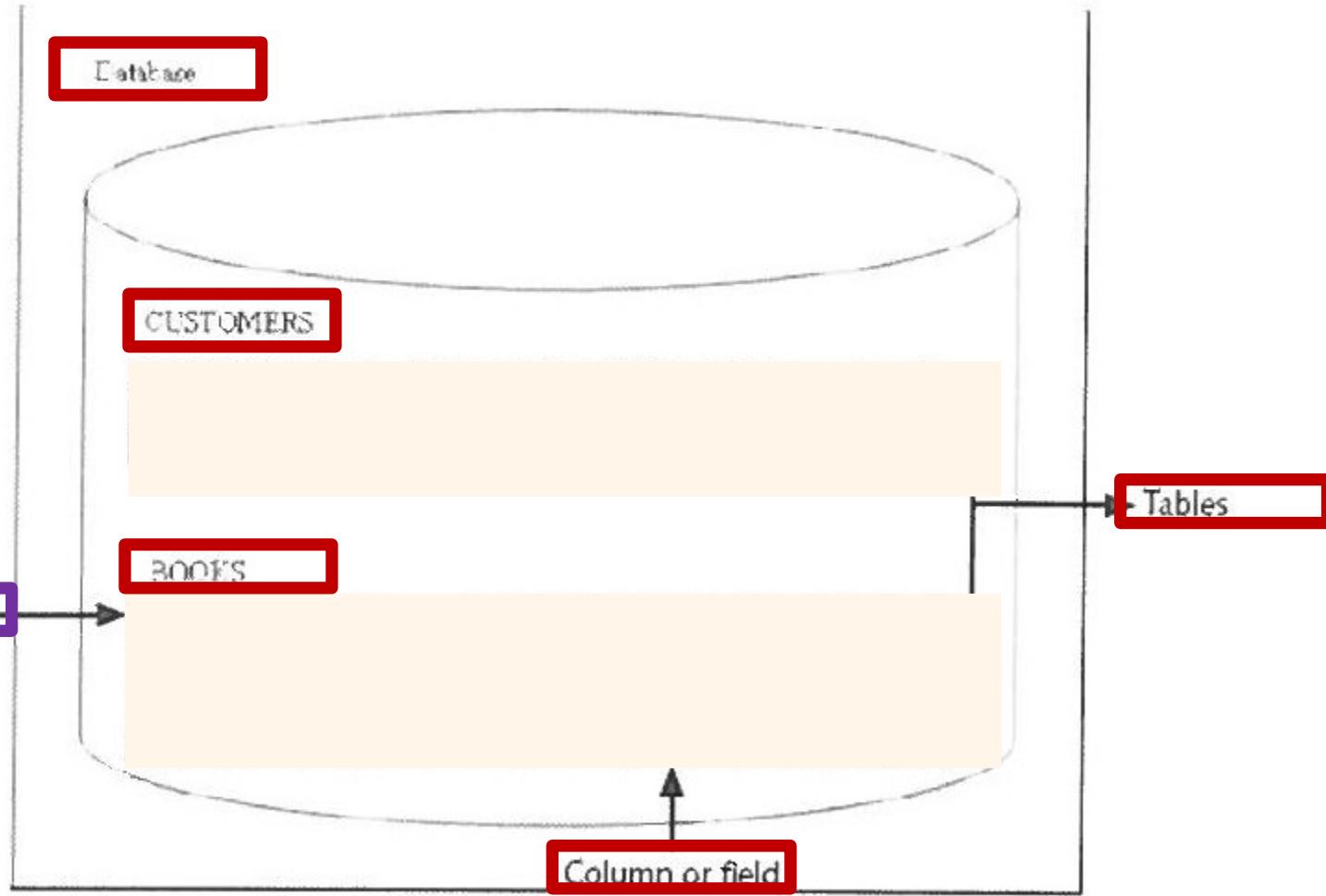
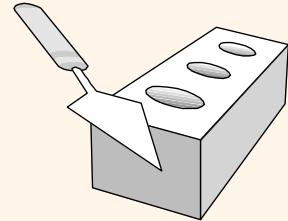
What Is a **Database**?



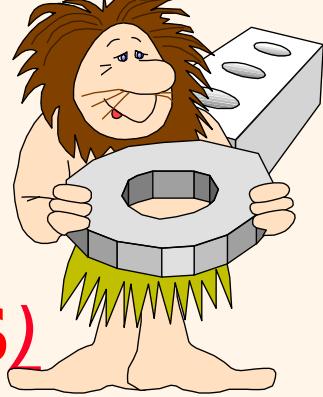
University Systems: A university wants to store **Data** about Faculty, Students, and Courses so that various aspects of this enterprise can be easily described.

- ❖ A very large, integrated collection of **Data**.
- ❖ Models real-world *enterprise*.
 - **Entities** (e.g., Faculty, Students, Courses)
 - **Relationships** (e.g., Madonna **is taking** CS564)

What is a **Relational** **Database**?



What Is a **DBMS**?



A **DataBase Management System (DBMS)**

is a **software** package designed to store and manage

~~Database~~

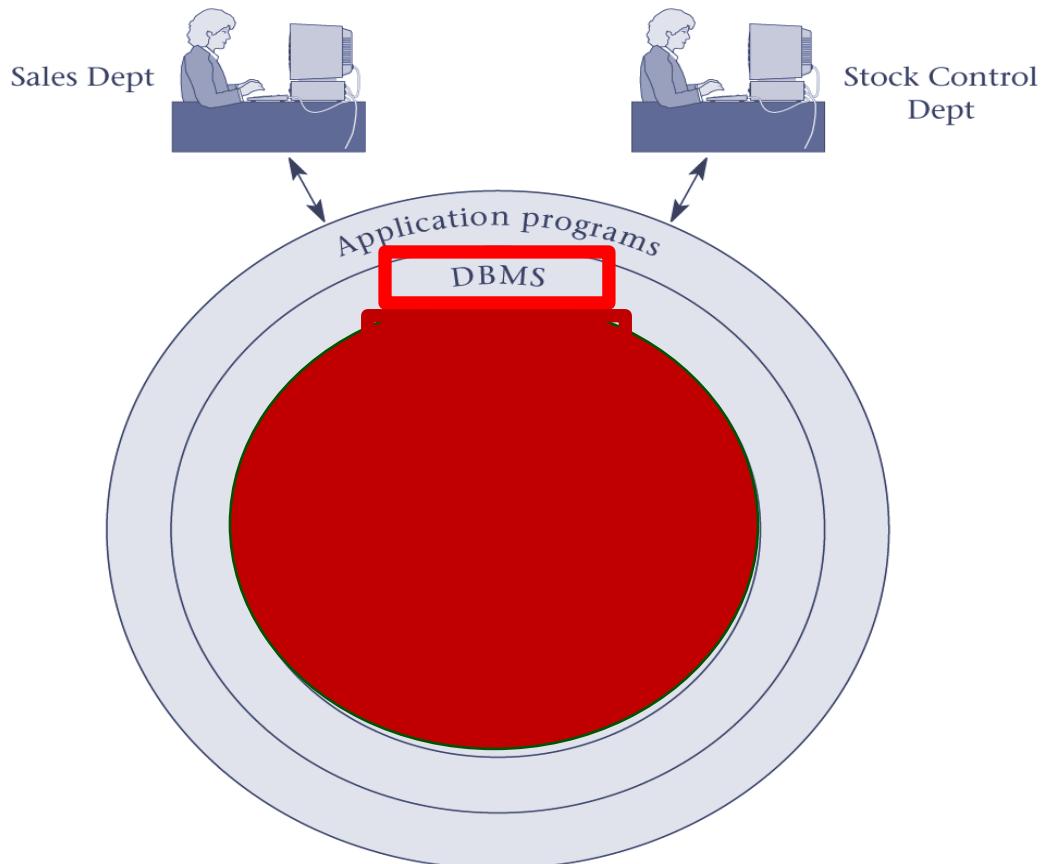
Oracle,
Microsoft SQL Server,
mySQL

What is a **DBMS**?



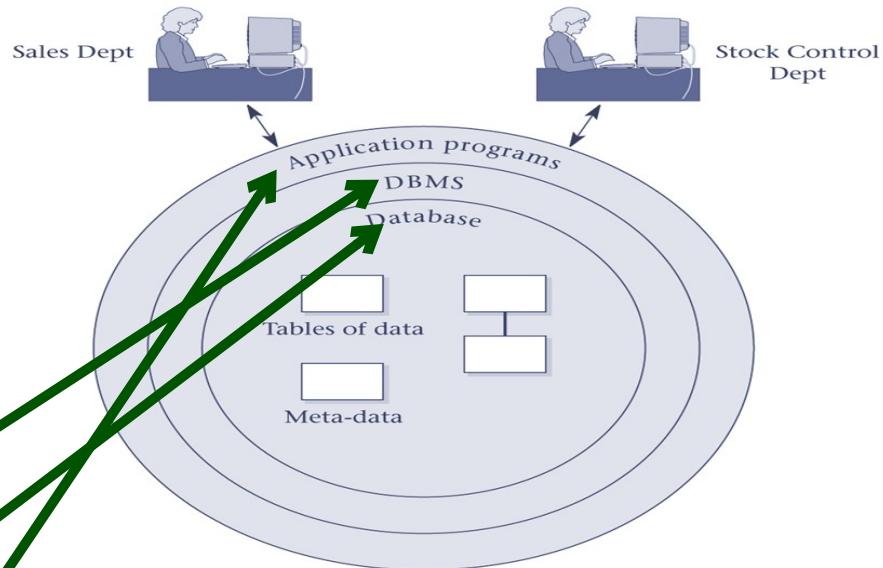
A ***DataBase Management System (DBMS)***

is a **software** package designed to store and manage
Data



Several Areas

...



Database System Implementation. How does one build a **DBMS**, including such matters as query processing, transaction processing and organizing storage for efficient access?

Design of Databases. How does one develop a useful **Database**? What kind of **information** go into the **Database**? How is the **information** structured? How do **Data** items connect?

Database Programming. How does one express queries and other operations on the **Database**? How does one use other capabilities of a **DBMS**, such as transactions or constraints, in an



2. SET 1 - 1: INTRODUCTION



2.1 Database basics



2.2 Database systems



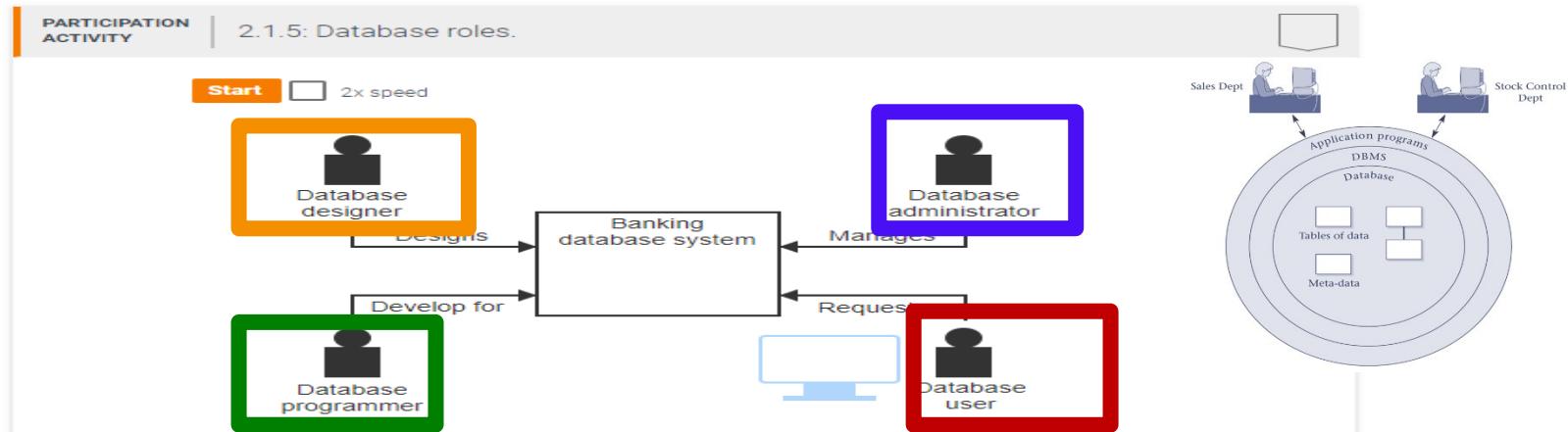
2.3 Database design and programming

Database roles

People interact with databases in a variety of roles:

- A **database administrator** is responsible for securing the database system against unauthorized users. A database administrator enforces procedures for user access and database system availability.
- A **database designer** determines the format of each data element and the overall database structure. Database designers must balance several priorities, including storage, response time, and support for rules that govern the data. Since these priorities often conflict, database design is technically challenging.
- A **database programmer** develops computer programs that utilize a database. Database programmers write applications that combine database query languages and general-purpose programming languages. Query languages and general-purpose languages have significant differences, so database programming is a specialized challenge.
- A **database user** is a consumer of data in a database. Database users request, update, or use stored data to generate reports or information. Database users usually access the database via applications but can also submit queries directly to the database system.

For simple databases with a limited amount of data and few users, one person may assume several roles. Ex: The database administrator might also be a database designer or database programmer. For large, complex databases, each person usually takes on just one role.



PARTICIPATION ACTIVITY

2.1.6: Roles.

1) Which role is responsible for providing access to the database?

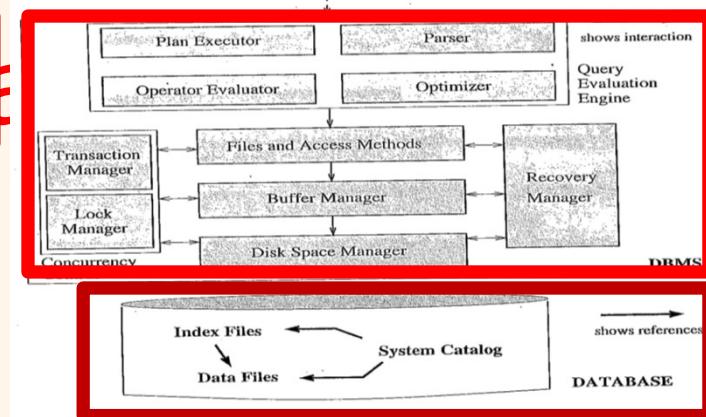
- Database administrator
- Database designer
- Database programmer
- Database users

Correct

Database administrators establish procedures for database security and user access.

?????

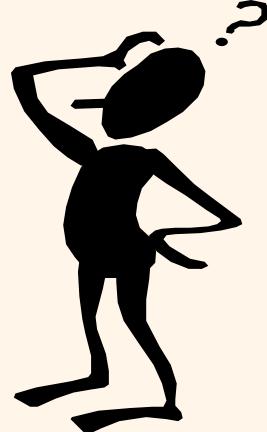
DBMS tools & Database



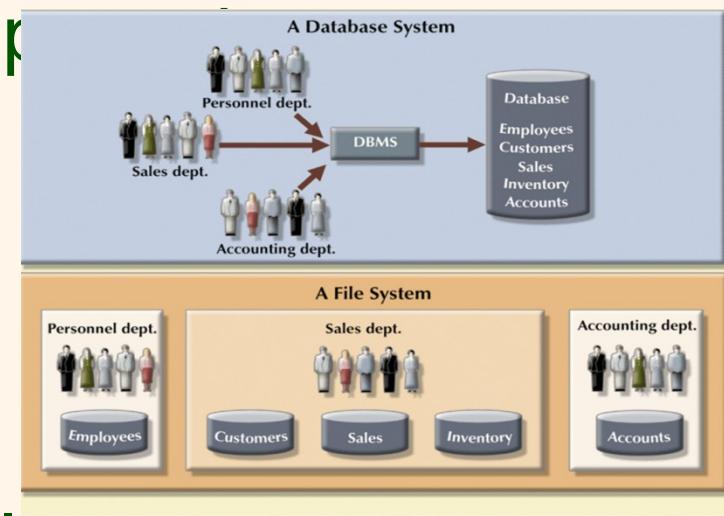
A **DBMS** gives you the tools to:

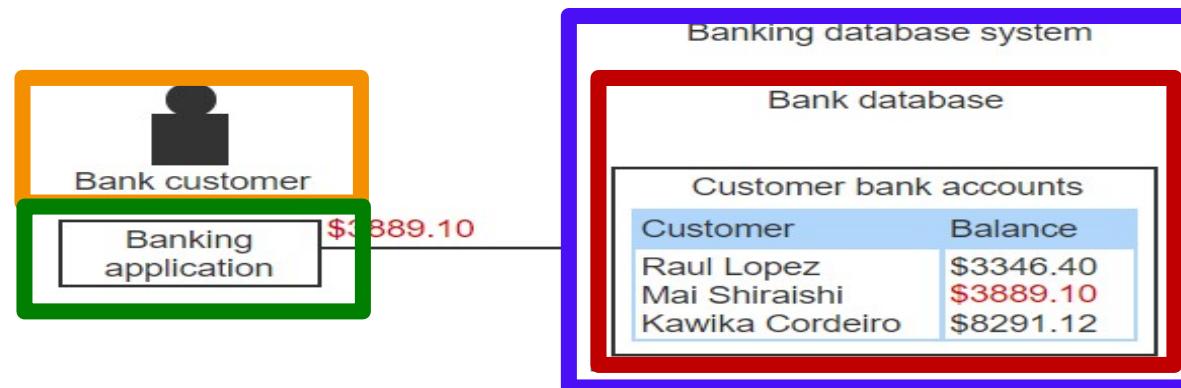
- **Design** the structure of your Database.
- **Create** data entry forms so you can get **information** into the Database.
- **Validate** the **Data** entered and check for inconsistencies.
- **Sort** and manipulate the **Data** in the Database.
- **Query** the Database (that is, ask questions about the **Data**).
- **Produce flexible reports**, both on screen and paper, that make it easy to comprehend the **information** stored in the Database.

Why use a **DBMS**?



- ❖ **Data** independence and efficient access.
- ❖ Reduced **application** development time.
- ❖ **Data** integrity and security.
- ❖ Uniform **Data** administration.
- ❖ Concurrent access, Recovery from crashes.



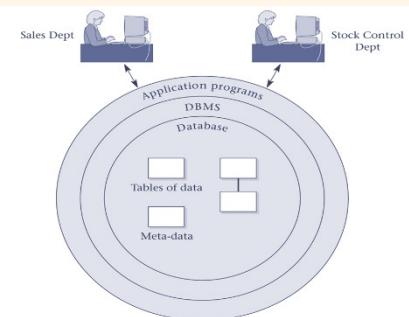
PARTICIPATION
ACTIVITY

2.1.4: Database in a database system.

- In the Banking database system
- In the customer banking application
- In the Bank database

A customer's banking data is stored in a database.

?????



Database vs. File Systems

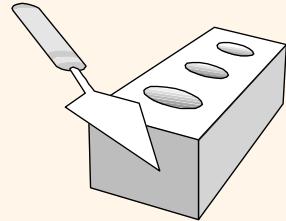
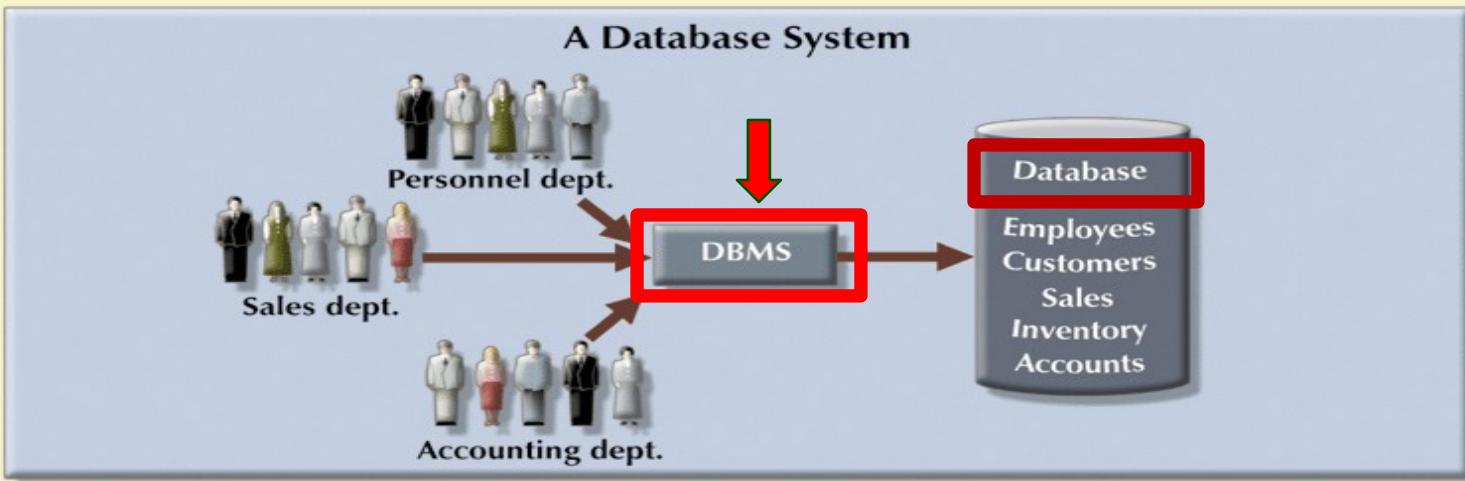
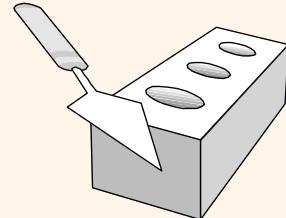


FIGURE
1.6

Contrasting database and file systems

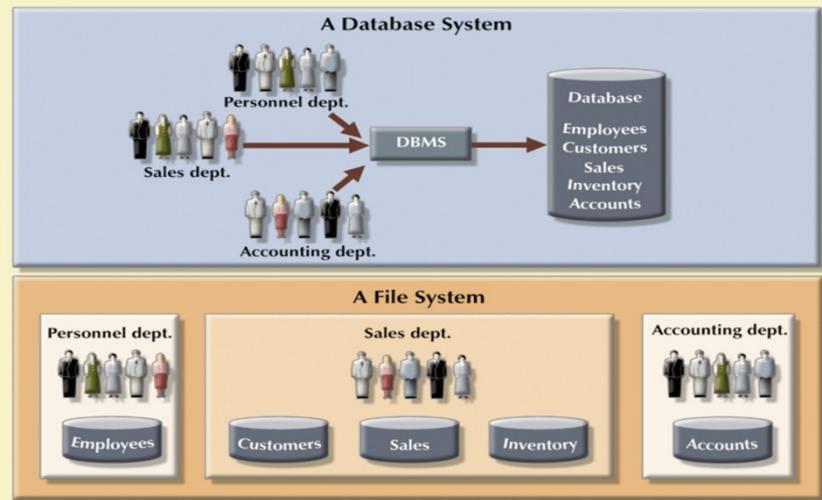


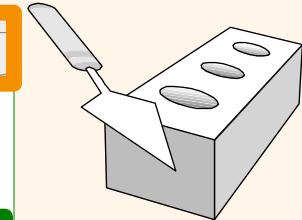


DBMS vs. Files

- ❖ Application must **move** large **Data**sets between main memory and secondary storage (e.g., buffering, page-oriented access, 32-bit addressing, etc.)
- ❖ Special **code** for different queries
- ❖ Must protect **Data** from **inconsistency** due to multiple concurrent users
- ❖ Crash Recovery
- ❖ Security and Access Cont

FIGURE
1.6 Contrasting database and file systems





- 1) If program A writes to bookkeeping.txt the same time that program B reads from bookkeeping.txt, what can potentially go wrong?

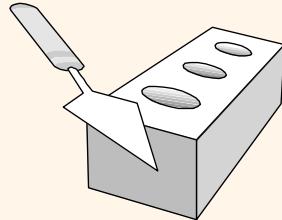
- Nothing can go wrong.
- Program A may be writing only partial data.
- Program B may be reading only partial data.

Correct

Program B may miss a recent update made to the file by Program A. A database can manage reading and writing to shared data to avoid such problems.

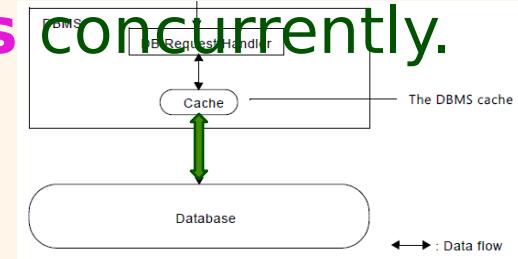
?????

Concurrency Control



Concurrent execution of **user programs** is essential for good **DBMS performance**.

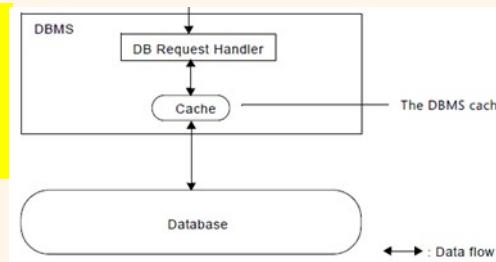
- Because **disk accesses** are frequent, and **relatively slow**, it is important to keep the **cpu** humming by working on several **user programs** **concurrently**.



Interleaving actions of different **user programs** can lead to **inconsistency**: e.g., check is cleared while account balance is being computed.

DBMS ensures such problems don't arise: **users** can pretend they are using a **single-user** system.
(ACID^{ta}Serializability)

Transactio n

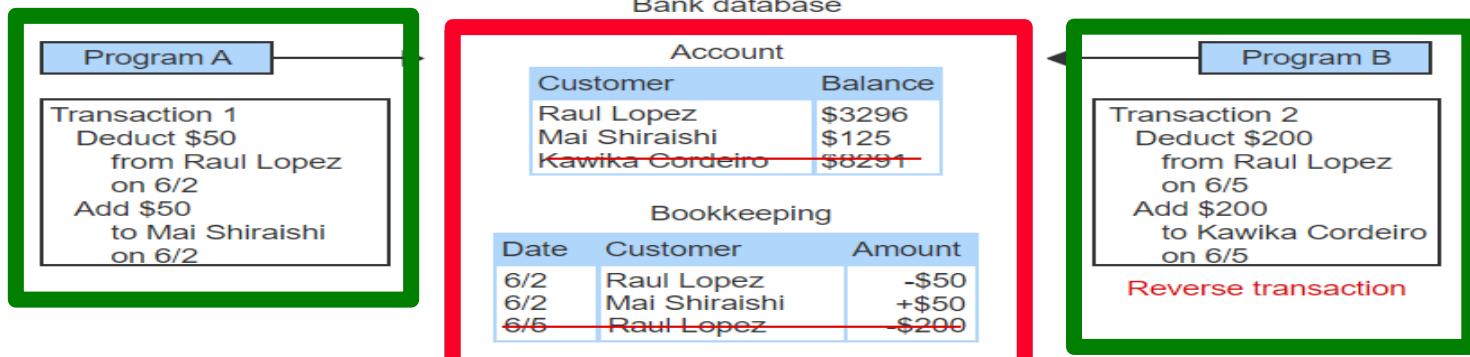
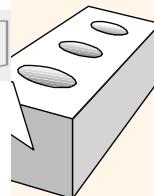


```
BEGIN;  
    INSERT INTO INVOICE  
        VALUES (1009,10016,'18-Jan-2006',256.99,20.56,277.55,'cred',0.00,277.55);  
    INSERT INTO LINE  
        VALUES (1009, '89-WRE-Q', 1, 256.99,256.99);  
    UPDATE PRODUCT  
        SET PROD_QOD = PROD_QOH -1  
        WHERE PROD_CODE = '89-WRE-Q';  
    UPDATE CUSTOMER  
        SET CUST_BALANCE = CUST_BALANCE +277.55  
        WHERE CUST_NUMBER = 10016;  
    INSERT INTO ACCT_TRANSACTION  
        VALUES (10007,'18-Jan-06',10016,'charge',277.55);  
COMMIT;  
END;
```

Key concept is ***transaction***, which is an ***atomic*** sequence of **Database actions (reads/writes)**.

Each ***transaction***, executed completely, must leave the **Database** in a ***consistent state iff*** **Database** is ***consistent*** when the ***transaction*** begins.

- **Users** can specify some simple ***integrity constraints*** on the **Data**, and the **DBMS** will enforce these constraints.
- Beyond this, the **DBMS** does not really understand the semantics of the **Data**. (e.g., it does not understand how the interest on a bank account is computed).
- Thus, ensuring that a ***transaction***(run alone) preserves



Match the transaction behavior to the described situation.

Ensure transaction results are never lost

Transaction processed completely or not at all

Prevent conflicts between concurrent transactions

Prevent conflicts between concurrent transactions

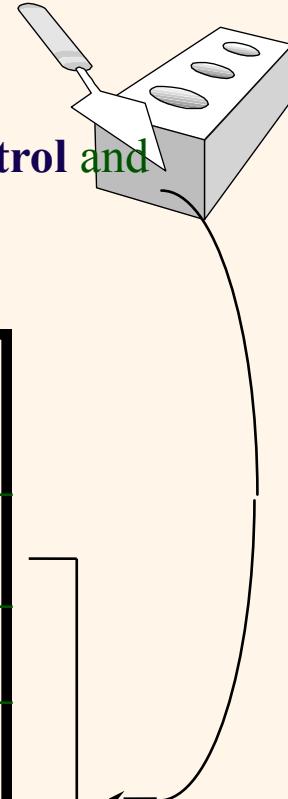
A program is adding a penalty fee to an account that is below \$1000 while another program is adding \$2000 to the same account.

Databases must ensure that transactions that access the same data at the same time do not produce a conflict. The fee is not added if the \$2000 is added to the account first.

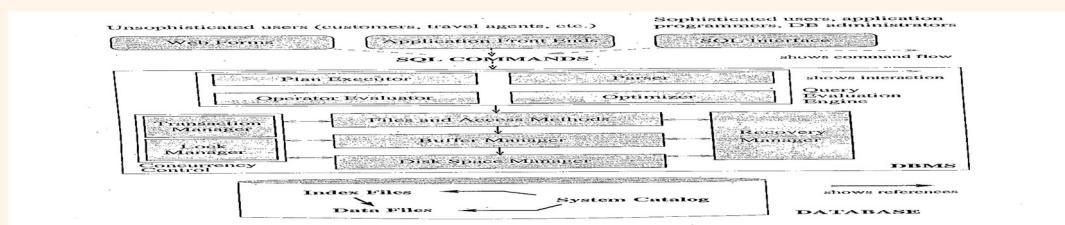
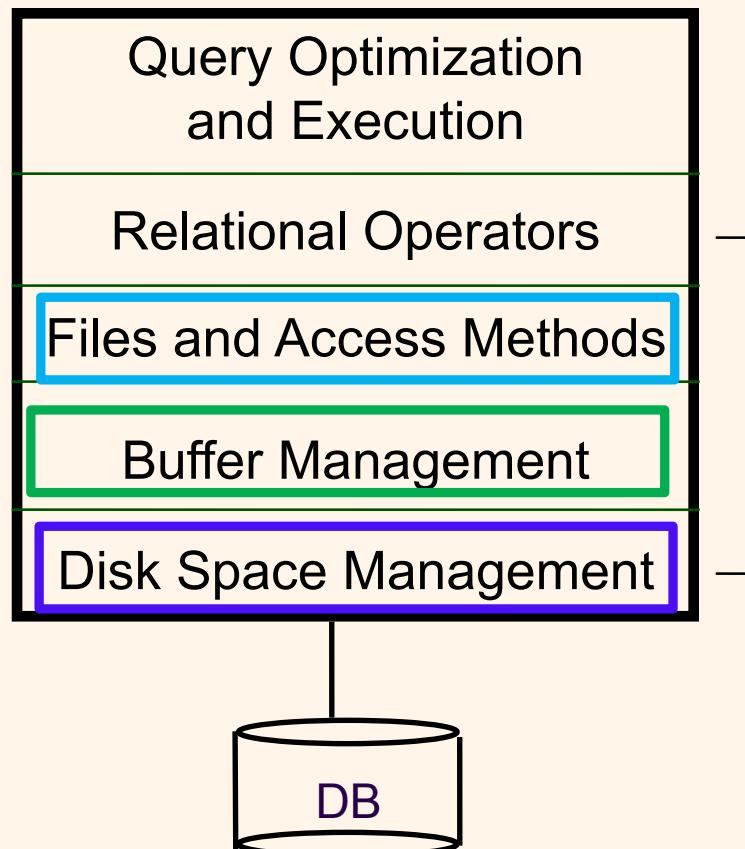
?????

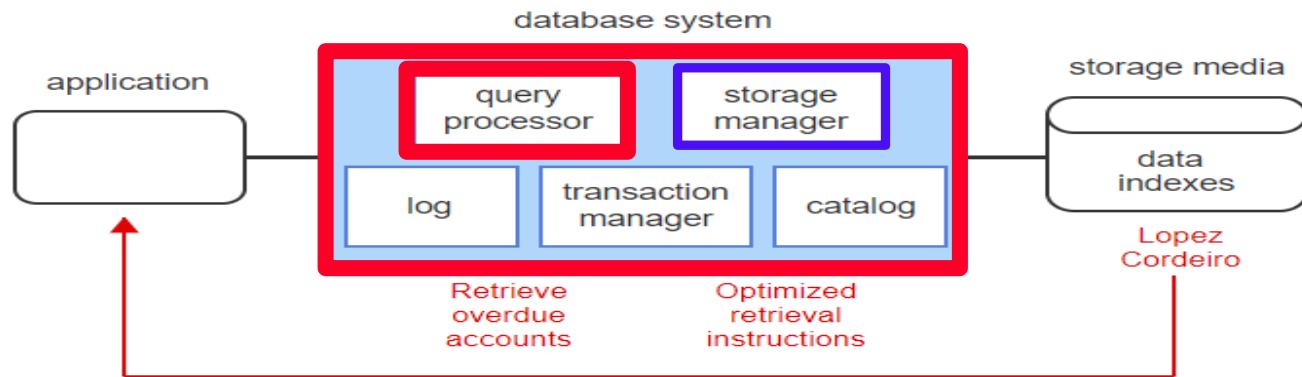
Structure of a DBMS

These layers must consider **Concurrency Control** and **Recovery**



- ❖ A typical **DBMS** has a layered **architecture**.
- ❖ The figure does not show the **Concurrency Control** and **Recovery** components.
- ❖ This is one of several possible **architectures**; each system has its own variations.



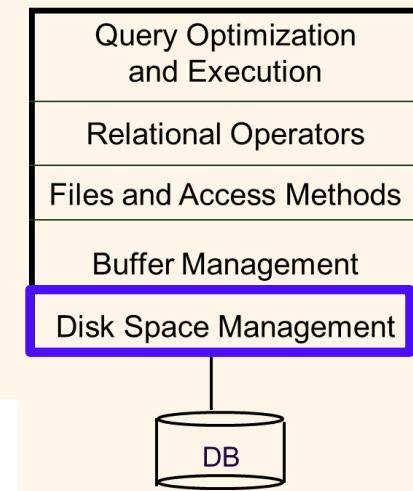


- 1) The query processor has direct access to the database data on storage media.

True

False

All access to the database data must go through the storage manager.

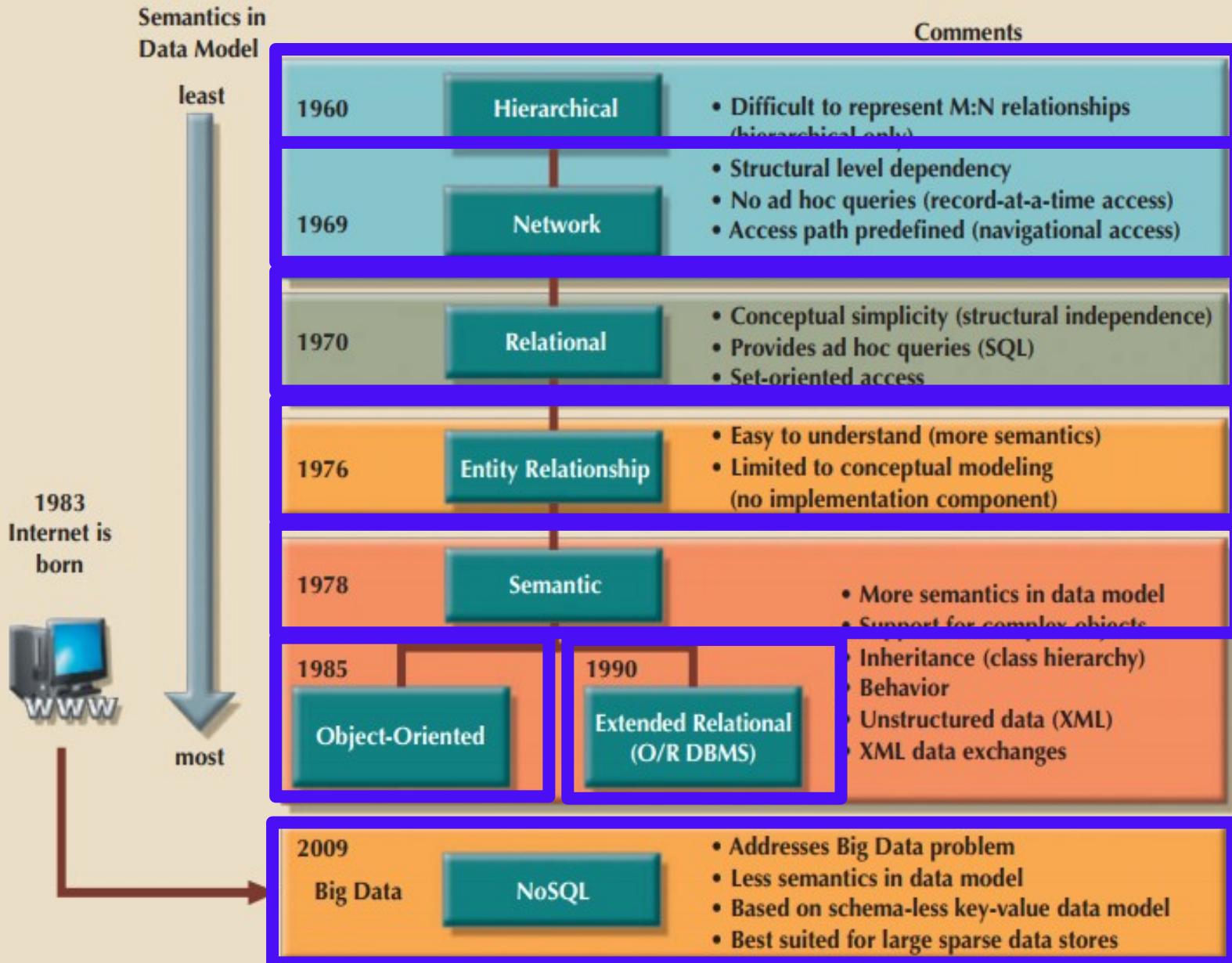


?????

EVOLUTION OF MAJOR DATA MODELS

GENERATION	TIME	DATA MODEL	EXAMPLES	COMMENTS
First	1960s–1970s	File system	VMS/VSAM	Used mainly on IBM mainframe systems Managed records, not relationships
Second	1970s	Hierarchical and network	IMS, ADABAS, IDS-II	Early database systems Navigational access
Third	Mid-1970s	Relational	DB2 Oracle MS SQL Server MySQL	Conceptual simplicity Entity relationship (ER) modeling and support for relational data modeling
Fourth	Mid-1980s	Object-oriented Object/relational (O/R)	Versant Objectivity/DB DB2 UDB Oracle 12c	Object/relational supports object data types Star Schema support for data warehousing Web databases become common
Fifth	Mid-1990s	XML Hybrid DBMS	dbXML Tamino DB2 UDB Oracle 12c MS SQL Server	Unstructured data support O/R model supports XML documents Hybrid DBMS adds object front end to relational databases Support large databases (terabyte size)
Emerging Models: NoSQL	Early 2000s to present	Key-value store Column store	SimpleDB (Amazon) BigTable (Google) Cassandra (Apache) MongoDB Riak	Distributed, highly scalable High performance, fault tolerant Very large storage (petabytes) Suited for sparse data Proprietary application programming interface (API)

THE EVOLUTION OF DATA MODELS



Data models

1970

Relational

- Conceptual simplicity (structural independence)
- Provides ad hoc queries (SQL)
- Set-oriented access

A **Data model** is a collection of concepts for describing **Data**.

A **Schema (type)** is a description of a particular collection of **Data**, using the a given **Data model**.

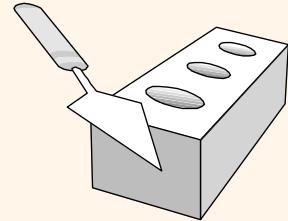
The **Relational Data model** is the most widely used **model**:

Main concept: **Relationships** and columns.

SUPPLIER							Columns (Attributes, Fields)					
Supplier_Number	Supplier_Name	Supplier_Street	Supplier_City	Supplier_State	Supplier_Zip							
8259	CBM Inc.	74 5th Avenue	Dayton	OH	45220							
8261	B. R. Molds	1277 Gandolly Street	Cleveland	OH	49345							
8263	Jackson Composites	8235 Micklin Street	Lexington	KY	56723							
8444	Bryant Corporation	4315 Mill Drive	Rochester	NY	11344							

Students (sid: string, name: string, login: string, age: integer, gpa: real)

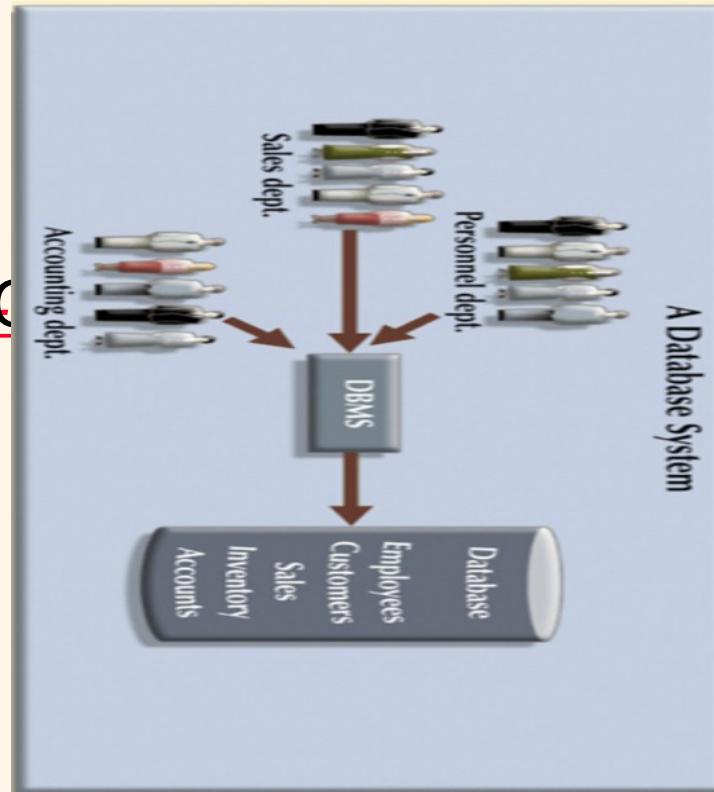
Levels of **Data** Abstraction



Many Views,

single Conceptual (Logical) Schema.

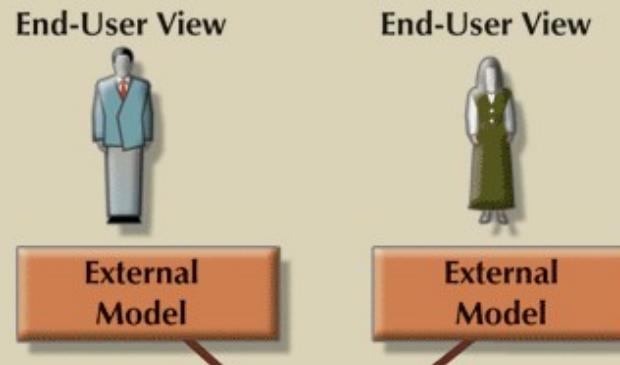
single Physical Schema - HOW.



- Views describe how **users** see the **Data**.
- Conceptual Schema defines **Data** logical structure - **WHAT**
- Physical Schema describes the **Files** and **Indexes** used

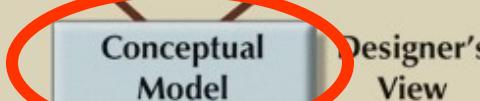
Schemas are defined using **DDL**; **Data** is modified/queried using **DML**

Data abstraction levels



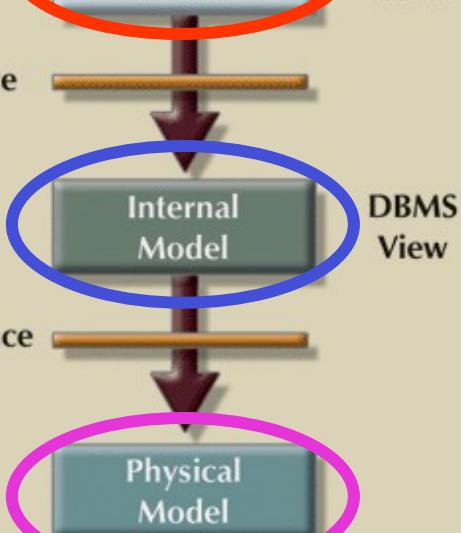
What - ERD

Logical independence



How - Relational

Physical independence



SQL

Degree of Abstraction	Characteristics
High	ER Object-Oriented
Medium	Relational
Low	Network Hierarchical

Example: University Database

External Schema (View):

Course_Info(cid:string, enrollment:integer)

Conceptual Schema - WHAT:

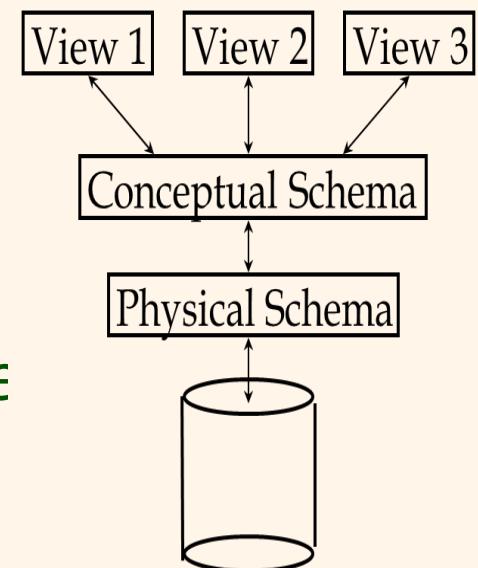
Student(sid:string, name: string, login: string, age: integer, gpa:real)

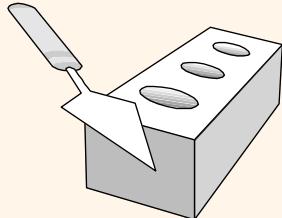
Course(cid:string, cname:string, credits:integer)

Enrolled(sid:string, cid:string, grade:string)

Physical Schema - HOW:

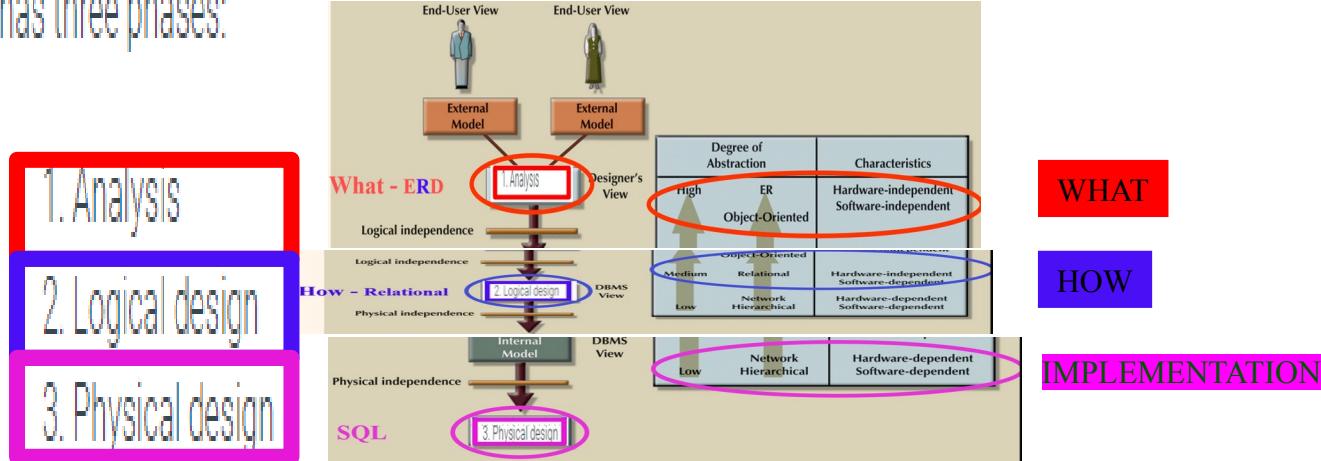
- **Relations** stored as **unordered Files (Heap)**.
- **Index** on first column of **Student**.



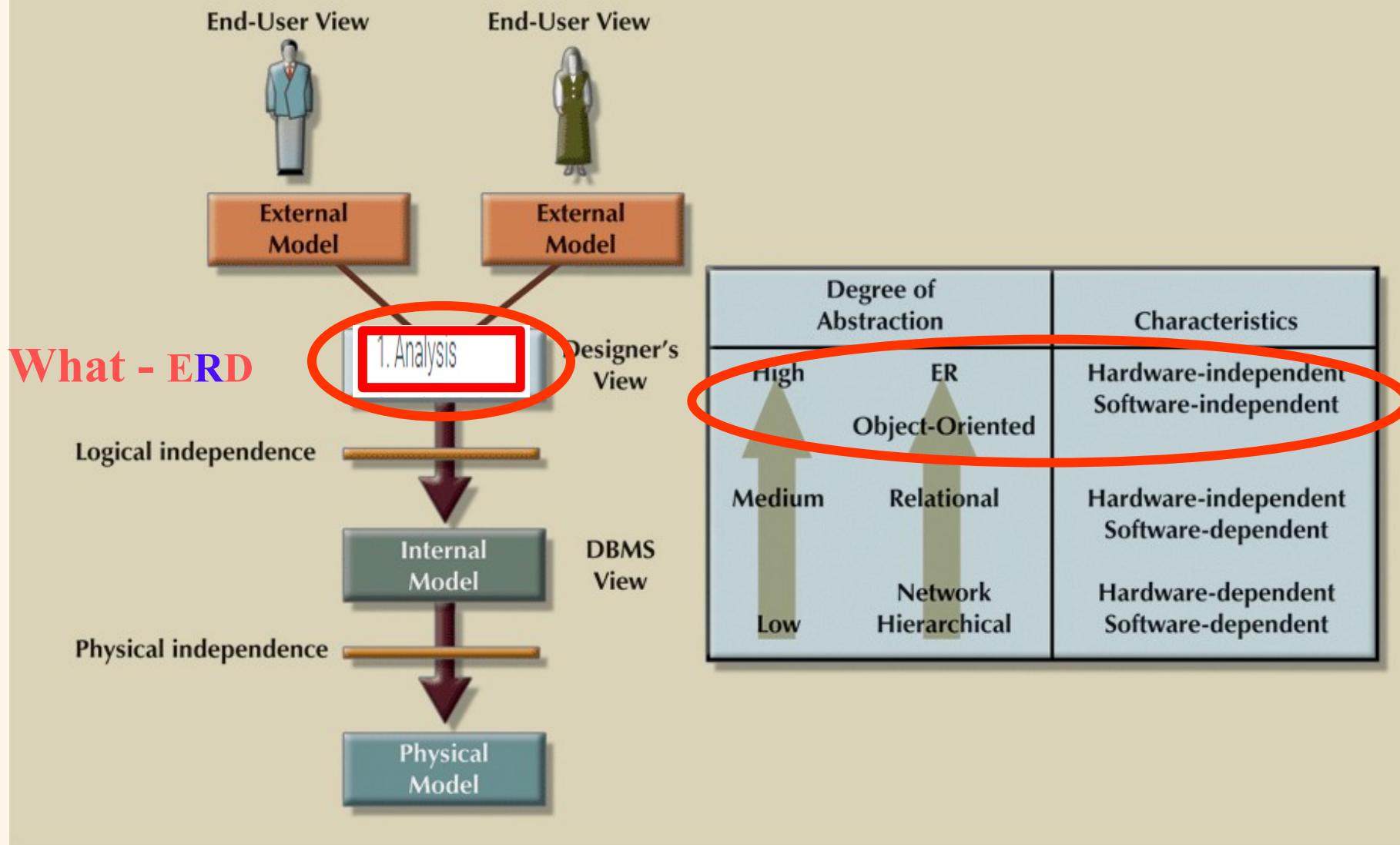


A **database design** is a specification of database objects such as tables, columns, data types, and indexes. Database design also refers to the process used to develop the specification.

For small, simple databases, the database design process can be informal and unstructured. For large, complex databases, the process has three phases:



Data abstraction levels



The **analysis** phase specifies database requirements without regard to a specific database system. Requirements are represented as entities, relationships, and attributes. An entity is a person, place, activity, or thing. A relationship is a link between entities, and an attribute is a descriptive property of an entity.

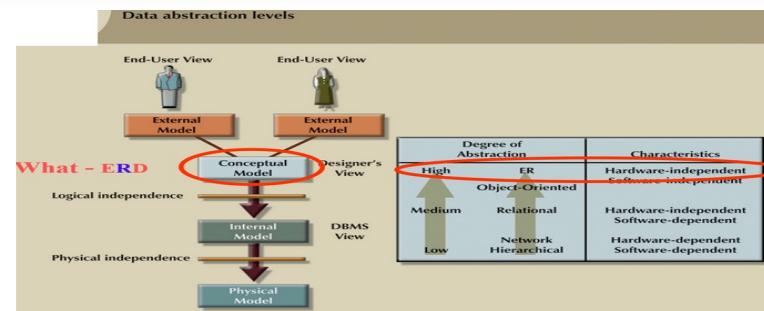
WHAT

Terminology

Analysis has many alternative names, such as conceptual design, entity-relationship modeling, and requirements definition.

Entities, relationships, and attributes are depicted in **ER diagrams**:

- Rectangles represent entities. Entity names appear at the top of rectangles.
- Lines between rectangles represent relationships.
- Text inside rectangles and below entity names represent attributes.



ER diagrams are usually supplemented by textual descriptions of entities, relationships, and attributes.

Figure 2.3.1: ER diagram.

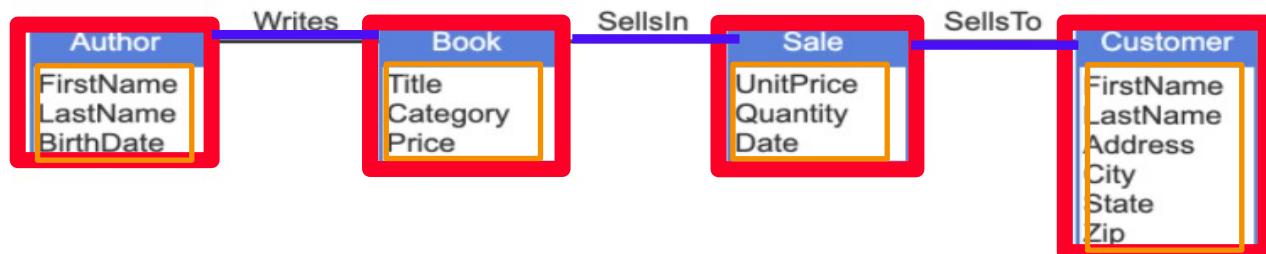


Entities, relationships, and attributes are depicted in **ER diagrams**:

- Rectangles represent entities. Entity names appear at the top of rectangles.
- Lines between rectangles represent relationships.
- Text inside rectangles and below entity names represent attributes.

ER diagrams are usually supplemented by textual descriptions of entities, relationships, and attributes.

Figure 2.3.1: ER diagram.



PARTICIPATION
ACTIVITY

2.3.1: Analysis.

Refer to the ER diagram above.

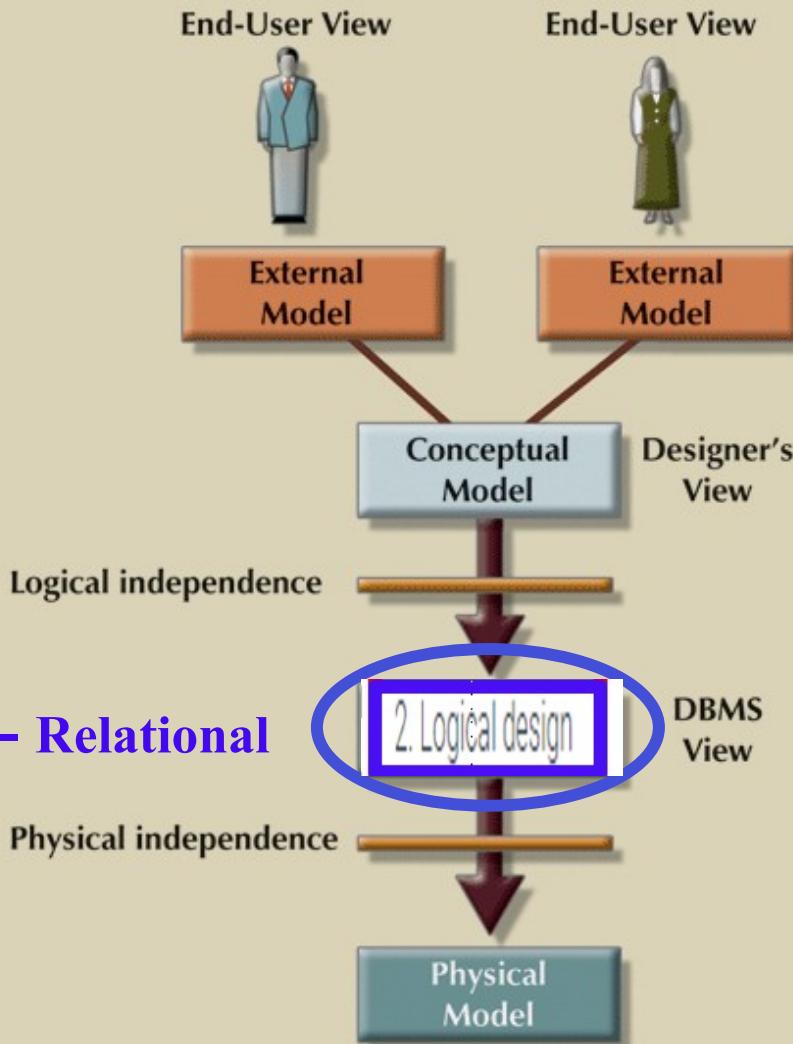
1) What is 'Writes'?

- Entity
- Relationship
- Attribute

Author-Writes-Book is a relationship between the Author and Book entities. Relationships are depicted as lines on ER diagrams.

?????

Data abstraction levels



Degree of Abstraction	Characteristics
High	ER Object-Oriented
Medium	Relational
Low	Network Hierarchical

How - Relational

The **logical design** phase implements database requirements in a specific database system. For relational database systems, logical design converts entities, relationships, and attributes into tables, keys, and columns. A **key** is a column used to identify individual rows of a table. Tables, keys, and columns are specified in SQL with CREATE TABLE statements.

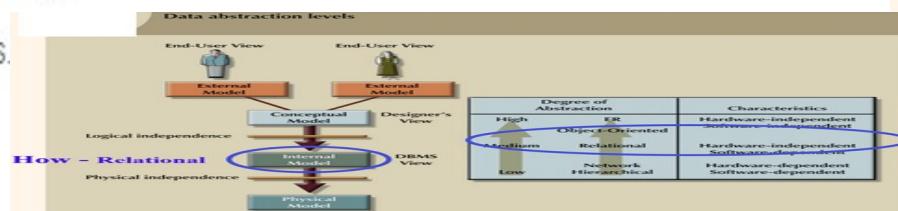
The logical design is depicted in a **table diagram**. Table diagrams are similar to ER diagrams but more detailed:

- Rectangles represent tables. Table names appear at the top of rectangles.

- Text within rectangles and below table names represents columns.

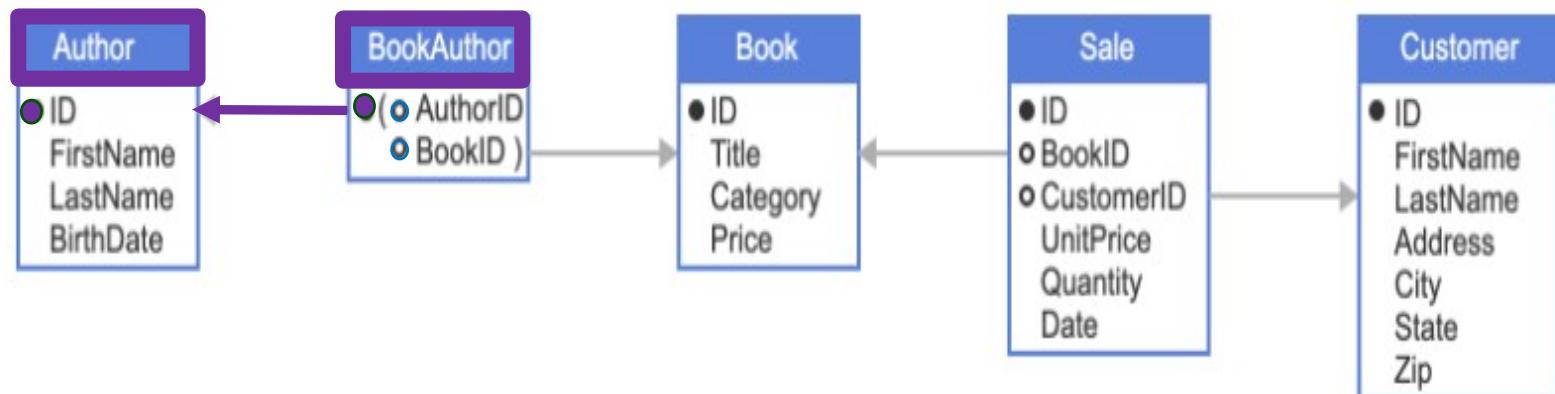
- Solid bullets (●) indicate key columns.

- Empty bullets (○) and arrows indicate columns that refer to keys.



The logical design, as specified in SQL and depicted in a table diagram, is called a database **schema**.

Figure 2.3.2: Table diagram.



The **relational model** is a database model based on mathematical principles, with three parts:

1. A data structure that prescribes how data is organized.
2. Operations that manipulate data structures.
3. Rules that govern valid relational data.

```
Song (
  ID INT,
  Title VARCHAR(60),
  Artist VARCHAR(60),
  GenreCode CHAR(3))
```

The relational data structure and operations are based on set theory. A **set** is a collection of values, or elements, with no inherent order. Sets are denoted with braces. Ex: {apple, banana, lemon} is the set containing three kinds of fruit. Since sets have no order, {apple, banana, lemon} is the same set as {lemon, banana, apple}.
The relational data structure is based on three mathematical concepts:

```
Song
(100, 'Hey Jude', 'Beatles', 'PRC'),
(200, 'You Belong With Me', 'Taylor Swift', NULL),
(300, 'Need You Now', 'Lady Antebellum', 'COU'),
(400, 'Old Town Road', 'Lil Nas X', NULL),
```

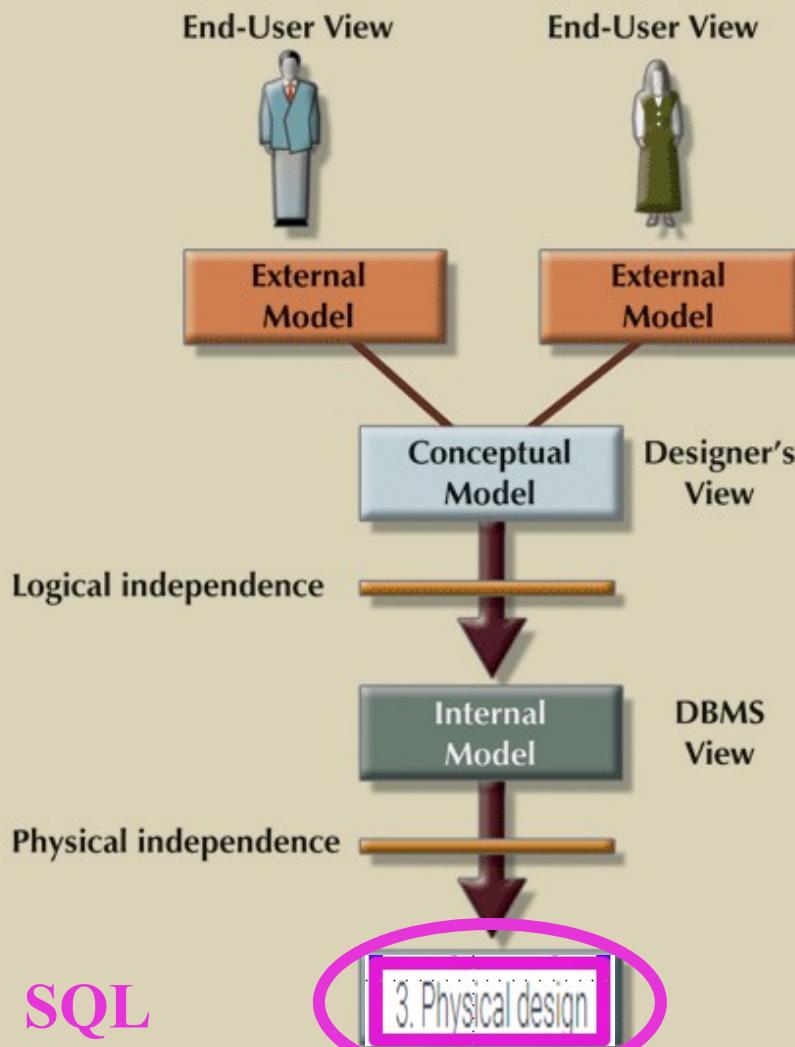
A **domain** is a named set of possible database values, such as integers, dictionary words, or logical values TRUE and FALSE.

A **tuple** is a finite sequence of values, each drawn from a fixed domain. Ex: (3, apple, TRUE) is a tuple drawn from domains (Integers, DictionaryWords, LogicalValues).

A **relation** is a named set of tuples, all drawn from the same sequence of domains. Ex: The relation below is named **Grocery** and contains three tuples.

Since a relation is a set, the relation tuples have no inherent order.

Data abstraction levels



Degree of Abstraction	Characteristics
High	ER Object-Oriented
Medium	Relational
Low	Network Hierarchical

IMPLEMENTATION

hrke

Physical design IMPLEMENTATION

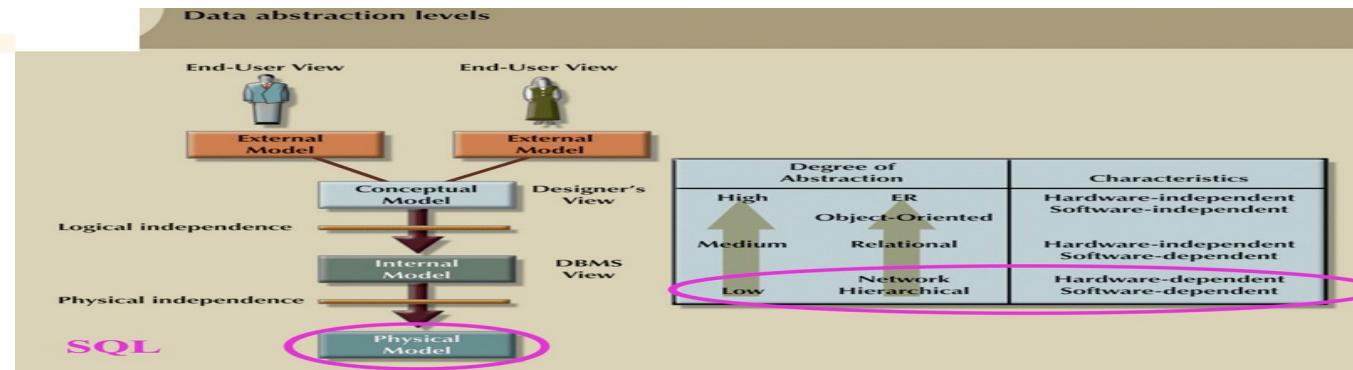
The **physical design** phase adds indexes and specifies how tables are organized on storage media. Ex: Rows of a table may be sorted on the values of a column and stored in sort order. Physical design is specified with SQL statements such as CREATE INDEX and, like logical design, is specific to a database system.

Physical design can be depicted in diagrams. However, logical design is more important for database users and programmers, so physical design diagrams are not commonly used.

In relational databases, logical and physical design affect queries differently. Logical design affects the query result. Physical design affects query processing speed but never affects the query result. The principle that physical design never affects query results is called **data independence**.

Data independence allows database designers to tune query performance without changes to application programs. When database designers modify indexes or row order, applications run faster or slower but always generate the same results.

Prior to relational databases, most database systems did not support data independence. Performance tuning often forced time-consuming changes to applications. Data independence is a major advantage of relational databases and contributed to the rapid adoption of relational technology in the 1980s.

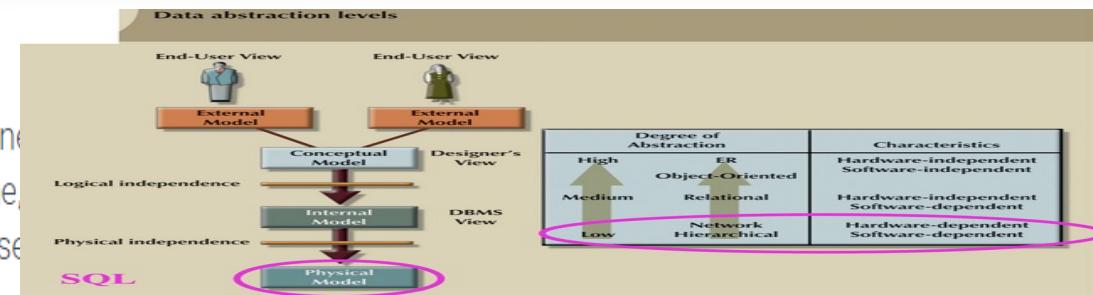


```
1 CREATE TABLE Genre (
2   Code CHAR(3),
3   Name VARCHAR(20),
4   Description VARCHAR(200),
5   PRIMARY KEY(Code)
6 );
7
8 CREATE TABLE Song (
9   ID INT,
10  Title VARCHAR(60),
11  Artist VARCHAR(60),
12  GenreCode CHAR(3),
13  PRIMARY KEY (ID),
14  FOREIGN KEY (GenreCode) REFERENCES Genre(Code)
15 );
16
17 INSERT INTO Genre VALUES
18  ('CLA', 'Classical', 'Orchestral music composed and performed by professionally trained artists'),
19  ('COU', 'Country', 'Developed mostly in southern USA, with roots in traditional folk music, spirituals and'),
20  ('DRO', 'Drone', 'Minimalist music that emphasizes sustained or repeated sounds, notes, or tone clusters'),
21  ('GRU', 'Grunge', 'Alternative rock inspired by hardcore punk, heavy metal, and indie rock'),
22  ('PRC', 'Pop Rock', 'Rock music with less attitude'),
23  ('RAB', 'R&B', 'Modern version of soul and funk African-American pop music'),
24  ('TEC', 'Techno', 'Electronic music');
25
26 INSERT INTO Song VALUES
27  (100, 'Hey Jude', 'Beatles', 'PRC'),
28  (200, 'You Belong With Me', 'Taylor Swift', NULL),
29  (300, 'Need You Now', 'Lady Antebellum', 'COU'),
30  (400, 'Old Town Road', 'Lil Nas X', NULL),
31  (500, 'That\'s The Way Love Goes', 'Janet Jackson', 'RAB'),
32  (600, 'Even Flow', 'Pearl Jam', 'GRU');
```

SQL syntax

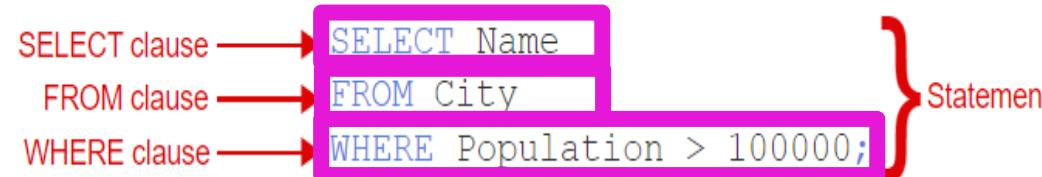
IMPLEMENTATION

An SQL **statement** is a complete command composed of one or more clauses like SELECT, FROM, and WHERE with table names like City, column names like Name, and conditions like Population > 100000. It is better to write each clause on a separate line, but good practice is to write each clause on a single line, but good practice is to write each clause on a single line.



PARTICIPATION ACTIVITY

8.5.3: Three clauses in a SELECT statement.



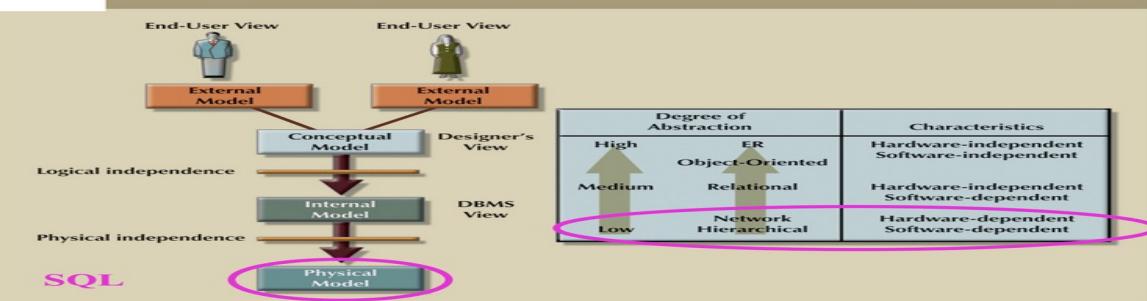
Captions ^

1. The SELECT clause starts the statement. Name is a column name.
2. The FROM clause must follow the SELECT clause. City is a table name.
3. The WHERE clause is optional. When included, the WHERE clause must follow the FROM clause. Population > 100000 is a condition.
4. The three clauses ending in a semicolon is a statement. The statement retrieves the names of all cities that have a population greater than 100,000 people.

Feedback?

2.3.3: Data independence.

Data abstraction levels

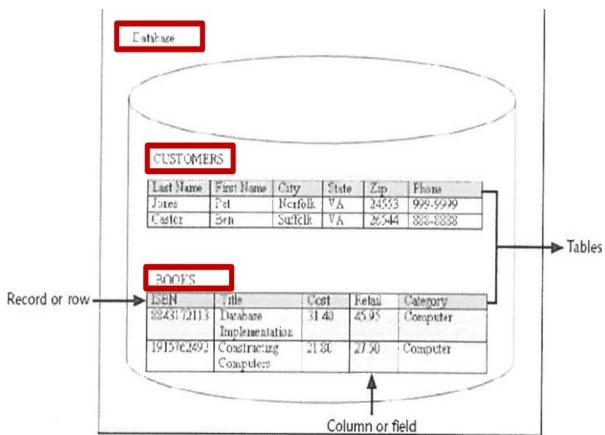


Book

Title Price

8210	Frankenstein	Fiction	5.95
1003	Gone with the Wind	Fiction	29.95
8209	History of Africa	History	12.50
1004	Iliad	Poetry	9.95
1002	Night	AutoBio	8.50
●			
●			
●			

1001	Pride and Prejudice	Fiction	6.00
1005	War and Peace	Fiction	34.95

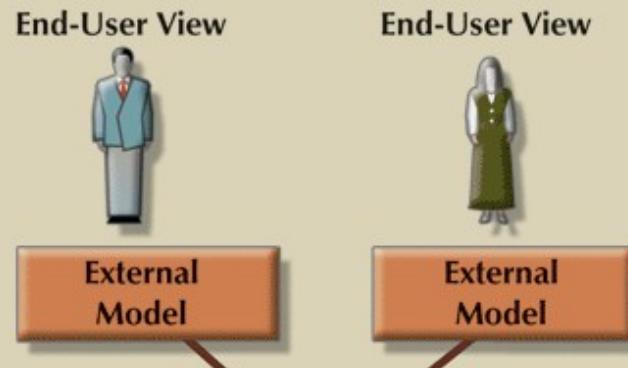


```
SELECT Title
FROM Book
WHERE Price > 20.00;
```

Result

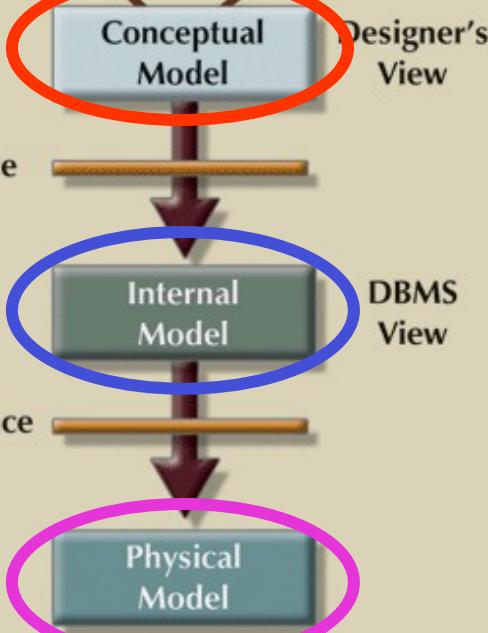
Title
Gone with the Wind
War and Peace

Data abstraction levels



What - ERD

Logical independence



How - Relational

Physical independence

Degree of Abstraction	Characteristics
High	ER Object-Oriented Hardware-independent Software-independent
Medium	Relational Hardware-independent Software-dependent
Low	Network Hierarchical Hardware-dependent Software-dependent

SQL

Logical design

The **logical design** phase implements database requirements in a specific database system. For relational database systems, logical design converts entities, relationships, and attributes into tables, keys, and columns. A **key** is a column used to identify individual rows of a table. Tables, keys, and columns are specified in SQL with CREATE TABLE statements.

The logical design is depicted in a **table diagram**. Table diagrams are similar to ER diagrams but more detailed:

- Rectangles represent tables. Table names appear at the top of rectangles.
- Text within rectangles and below table names represents columns.
- Solid bullets (●) indicate key columns.
- Empty bullets (○) and arrows indicate columns that refer to keys.

The logical design, as specified in SQL and depicted in a table diagram, is called a database **schema**.

Figure 2.3.2: Table diagram.



PARTICIPATION ACTIVITY

2.3.4: Database design process.

Match the term to the description.

Physical Model

Conceptual Model

Internal Model

Physical design

Analysis

Logical design

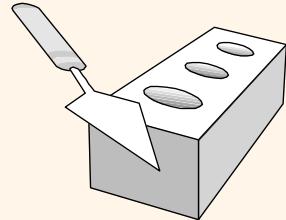
Database design

?????

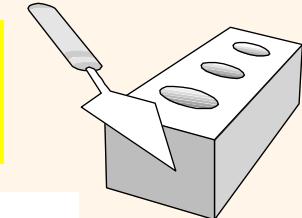
Logical design

Implementation of database requirements as tables, keys, and columns in a specific database system

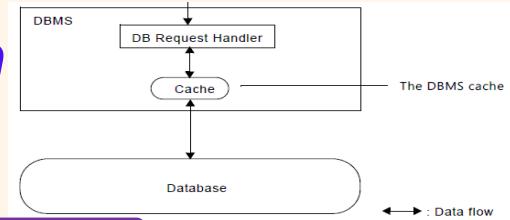
Logical design results in SQL specifications for tables, columns, and keys in a specific database system.



Data**bases** make these folks happy ...



❖ **End users** and **DBMS vendor**

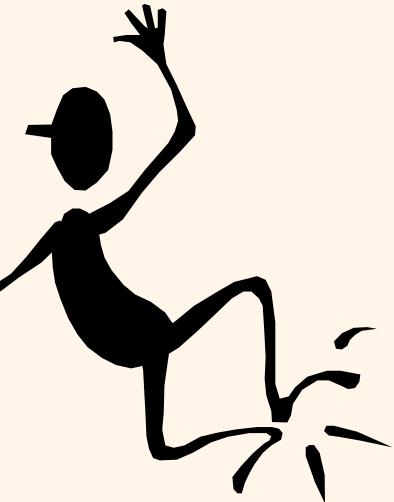


❖ **DB application programmers**

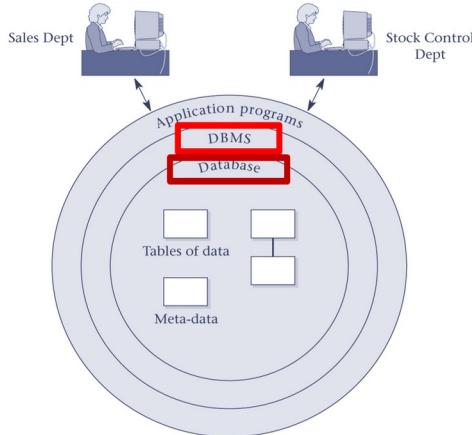
- e.g., smart webmasters

❖ **DataBase Administrator (DBA)**

- Designs logical /physical **Schemas**
- Handles security and authorization
- **Data** availability, crash recovery
- **Database** tuning as needs evolve



Just understand how a **DBMS** works!



**SELECT Title
FROM Book
WHERE Price > 20.00;**

Book

ID	Title	Category	Price
1001	Pride and Prejudice	Fiction	23.00
1002	Night	AutoBio	8.50
1003	Gone with the Wind	Fiction	29.95

Title: Pride and Prejudice
Category: Fiction
Title: Gone with the Wind
Category: Fiction

```
bookCursor = bookDatabaseConnection.cursor()
bookQuery = ('SELECT Title, Category'
             'FROM Book'
             'WHERE Price > 20.00')
bookCursor.execute(bookQuery)

for resultRow in bookCursor.fetchall():
    print('Title:', resultRow[0])
    print('Category:', resultRow[1])
```

Book				
ID	Title	Category	Price	
1001	Pride and Prejudice	Fiction	23.00	
1002	Night	AutoBio	8.50	
1003	Gone with the Wind	Fiction	29.95	

Title: Pride and Prejudice
Category: Fiction
Title: Gone with the Wind
Category: Fiction

- 1) Each host language, such as Java or C++, has a different API.

True

False

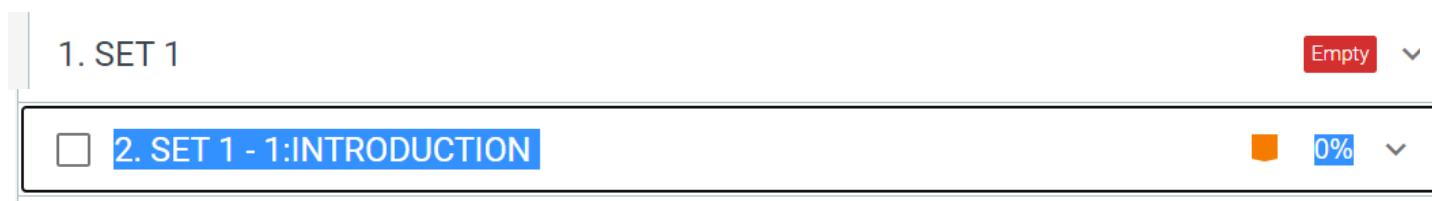
Correct

Since the API is written primarily in the host language,
different host languages must have different APIs.

?????

From 5:05 to 5:10 PM – 5 minutes.

01.17.2024	ZyBook SET 1-1	Overview CANVAS and ZyBook
(1 - We)		Set 1
		Lecture 1 Introduction



Please work on
SET 1 – 1: INTRODUCTION

SET 1 Lecture 1

INTRODUCTION

2. SET 1 - 1: INTRODUCTION

 0% 

2.1 Database basics

 0% 

2.2 Database systems

 0% 

2.3 Database design and programming

 0% 

Next

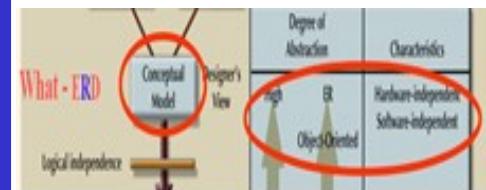
01.22.2024

ZyBook SET 1 - 2

Set 1

(2 - Mo)

LECTURE 2 DATA MODELING - WHAT - ERD MODEL



WHAT

1. SET 1

Empty

3. SET 1 - 2: DATA MODELING - WHAT - ERD MODEL

0% 0%

Please start working on
SET 1 – 2: DATA MODELING – WHAT – ERD MODEL

From 5:10 to 5:20 PM – 10 minutes.

01.17.2024	ZyBook SET 1-1	Overview CANVAS and ZyBook
(1 - We)		Set 1 Lecture 1 Introduction

CLASS PARTICIPATION 20 points 20% of Total + :

I AM IN TEAMS

Class 1 END PARTICIPATION

CLASS PARTICIPATION [20 points] Module | Not available until Jan 17 at 5:10pm | Due Jan 17 at 5:20pm | 100 pts

VH, publish ∅ :

This is an synchronous online class.

Attendance is required.

Recording or distribution of class materials is prohibited.

1. At the beginning of selected classes there is an assessment. (beige BOX in the Detailed Syllabus)
2. At the end of selected classes there is an assessment. (blue BOX in the Detailed Syllabus)
3. ZyBook sections will be downloaded on the dates specified in the Detailed Syllabus at 4 PM.
4. EXAMS are in CANVAS and camera is required. No late EXAMS.
5. I have to be present in TEAMS in order to take ANY graded assignments.

At 5:15 PM.

End Class 1

VH, unhide ZyBook Sections 2 & 3.



VH, Download Attendance Report
Rename it:
1.17.2024 Attendance Report FINAL

VH, upload **Class 1** to CANVAS.