Chapter 1:

define *database*: a collection of related data - known facts that can be recorded and have implicit meaning

define *data*: known recordable facts - with meaning

>   Ex: University database in which info is kept on students - such as name, year of grad, birthdate, etc.

characterisitcs:
- represents some aspects of the real world
- logically coherent collection of data - some inherent meaning
- data to be used for a specific purpose

define *database management system (DBMS)*: a collection of programs that enables users to create and manitain a database. General purpose software for:
- defining: specify data types, structures and constraints
- constructing: storing on some storage medium
- manipulating: querying, updating, and generating reports

define *database system*: database together with the DBMS
(picture from text)

Example: Toy Catalog Database (handout)
- briefly explain the application - Catalog company sells toys
- explain what each file is used for - show distinction among files, records, data
        elements (fields)

define: specify data types (ex: age - int, name - string, etc)
construct: insert values (populate db)
manipulate: query or update
        example queries:

                1) Which customers have children under 3?
                2) List any outstanding orders.
                3) Add Fischer-Price Pony Pickup to the toy file
                4) Change address of George Grant customer

Characteristics that differ from traditional file processing:
- self-describing nature of a database
        description of data is part of the database - stored in a catalog
        called "meta-data" - data about data
        used by DBMS software and db users to get info about db structure
        data definition separate from application programs

        - point out meta-data in example
        - used to form queries

- program-data independence
        application programs written independent of db structure - change in structure
                will not affect existing programs

- ex: Add field to TOY relation - Colors available

- data abstraction
        users (app progs) have a conceptual representation of data but don't know about
                how data is stored

                - ex: ages of children in Cust DB can be stored in a linked list or array
        definition: *data model* - a type of data abstraction - provides conceptual
                representation using logical constructs not related to actual storage of
                data (more on this later)

- multiple views of data
        different users may require different perspectives (views) of the same data
        a view may be
                a) a subset of the data in the database
                b) may include virtual data - derived from data in db but not stored
                there

- data sharing
        a db allows multiple users - data must be shared
        access to shared data must be monitored to ensure that multiple accesses do not
                corrupt the data - *concurrency control*

Database personnel
- database administrator - administers the database and DBMS resources
responsible for authorizing access, coordinating and managing use, and
acquiring software and hardware as needed
- database designers - designs the structure of the database
identifies the data to be stored and how to store it
communicate with users to determine what data to store and what views to
provide
- end users - access data through app software, queries, updates and report generation
- programmers/analysts - design and produce the application software for canned
programs to be used by end users

Uses of a DBMS
- controlling redundancy
data is stored in a central place for use by many applications and users
single storage location helps maintain consistency of data
- security
allows only authorized users to access certain data - views are designed to
provide only the data that is permitted to the particular user
- persistent storage
unlike in programming languages - dbms data remains persistent (exists) after
the program using it is complete
- database inferencing
define rules for deriving data - data can change while rules stay the same
- multiple user interfaces - to fit the sophistication of the user
query languages, application programs, menu driven interfaces, etc.
- representing complex relationships
dbms represents related data and retrieves and updates related data efficiently
- enforcing integrity constraints
constraints can be defined that must hold in order for the data to be considered
correct
dbms enforces constraints
- backup and recovery - in case of system failure
recover to a consistent state using backup data

Advantages of Using a DBMS:

-enforcing standards
naming, formats, report structures, etc.
form many systems and appls in compant
facilitates coordination among depts
- reduced appl developement time
initial added time for building db is offset by simpler appl developement
- flexibility - easier to restructure data
- available information
one user's update is available immediately to another user - w/ concurrency
control and recovery

Disadvantages:

- high initial investment - equipment, training, software
- not special purpose - general

- overhead for security, CC, recovery, etc. - all things that make DBMS beneficial have a cost

Don't use a DBMS if:

- simple application - one-time use
- stringent real-time requirements (we'll see more on this later)
- multi-user access not needed