Object-Oriented Data Model

Why introduced?
- characteristics of traditional database applications
  1. large # of similarly structured data with same size
  2. consists of fixed-length records (usually short)
  3. atomic fields: 1NF
  4. short transactions: execution time in the range of sec
  5. static conceptual schemas

New applications
  1. complex objects and complex internal structures
  2. behavioral data: distinct objects respond differently to the same command
  3. meta knowledge: rules in addition to tuples
  4. long-lived transactions, human interaction possible
  5. CAD, CAE, CAM, multimedia DB, etc

Notions in Object-Oriented Data Model

- An adaptation of OO programming paradigms to databases
- Encapsulating data and code in objects
  - interface between objects: message to invoke method
- Class
  - a means of grouping all objects that share the same set of attributes and methods
  - an object belongs to only one class as an instance
  - enhance the system integrity by type checking
- Class hierarchy and inheritance
  - allow deriving a new class from existing classes
  - subclass inherits all attributes and methods of the superclass
  - single inheritance: one superclass
  - multiple inheritance: more than one superclass
  - ambiguity: potential name conflicts among classes