**CS 4750: Database System**

External Level:

* Users can see each other
* External schema needed for front end

Logical Level:

* All the possible description of data that is available
* Can transfer to external level

Internal and Physical Level

* Internal – physical implementation
* Physical – managed by operating system

Data Independence

* Upper and lower levels are unaffected
* Logical indepdendence – change to logical model does not impact external model

Data Model

* Collection of concepts or notations for describing data into databases
* Three parts
  + Structure
  + Integrity
  + Manipulation

Relational Data Model

* Represents data in tabular format
* Attributes = columns
* Tuple, row, report = rows
* Order does not matter
* Datatypes of each attribute does not change
* Cannot have duplicates

Schema

* Logical design of database
* Instance – data stored in database at given time

Keys in Relations

* Used to uniquely identify a tuple
* Super key: attribute that can uniquely identify a tuple
* Candidate key: minimal super
* Primary key: candidate key that is most important key
  + Unique
  + Not empty or NULL
  + Not change
  + Meaningful
  + Most important candidate key
  + Autogenerated incremental
    - Problems
      * Won’t know which key is associated with each tuple
      * Too simple and security is weak
* Composite key: not one single attribute that can uniquely identify each tuple
  + Use a combination of multiple attributes
* Foreign key: attribute from one relation is mapped to some tuple in another relation

Entity-Relation Diagrams

* E-R Diagram: high-level design model representing a database as a collection of entities and relationships among entities
* Entity: real-world object; instance
* Entity-set: collection of entities of a particular type
* Instance: actual occurrence of entity (row)
* Attribute: a field (property of those entities)
  + Single-valued attribute
  + Multi-value
  + Derived
  + Composite