شُروع اَللہ کے پاک نام سے جو بڑا مہر بان نہایت رحم والا ہے

Dr. Abid Sohail Bhutta

abidbhutta@cuilahore.edu.pk

Department of Computer Science,

COMSATS University , Lahore Campus

Database Systems

Lecture 17

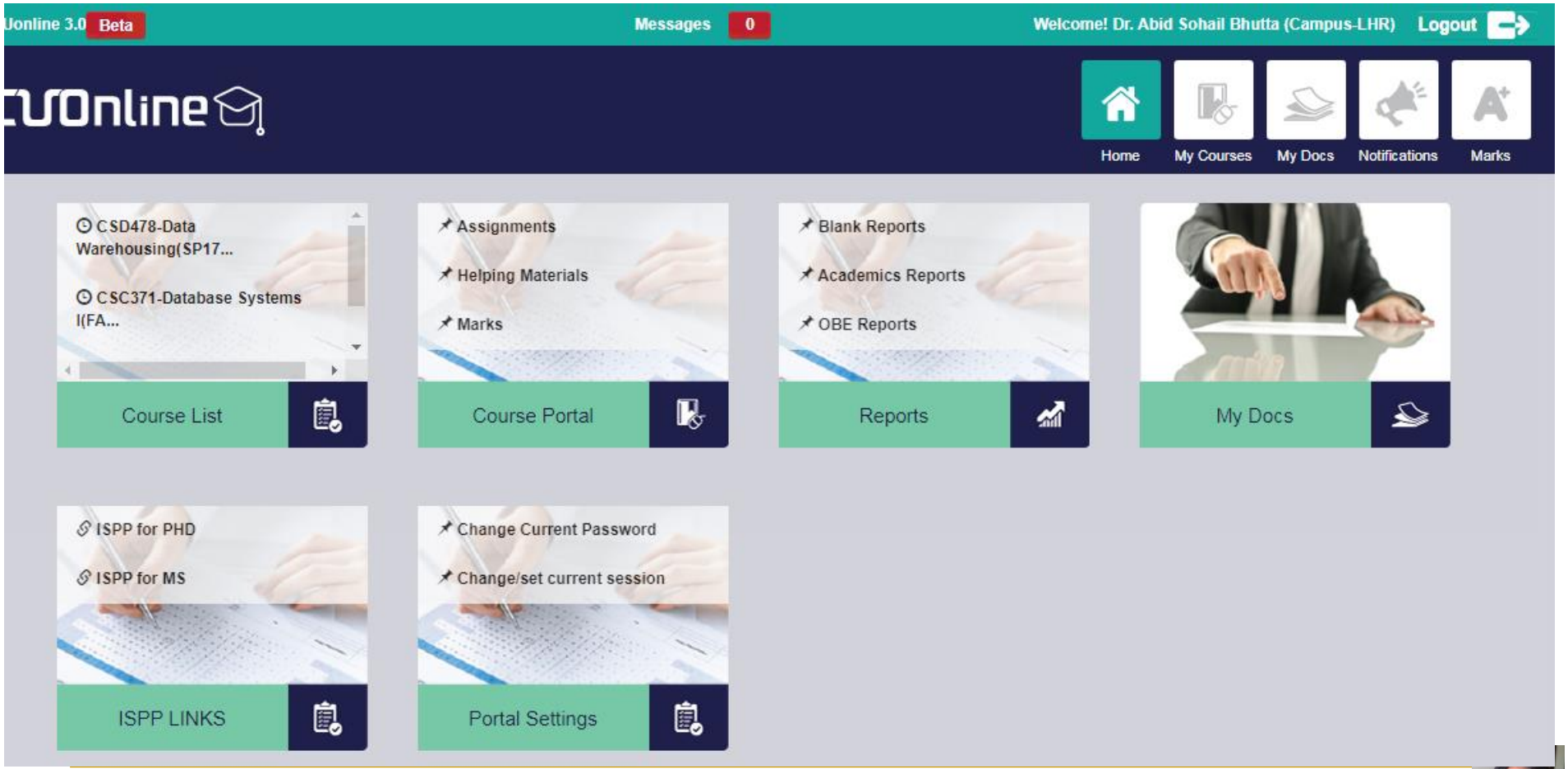
Enhanced Entity Relationship Diagram (EERD) and Inheritance in Schema Modeling (Part 1)

Recall Lecture 16

- Database Schema Designing
  - Entity Relationship Diagram (ERD)
  - Entities Classification
    - Transactional Entities
    - Component Entities
    - Classifying Entities

ERD of CUONLINE

- Common Mistakes



ERD of CUONLINE

CUonline 3.0 Beta Messages 0 Welcome! Dr. Abid Sohail Bhutta (Campus-LHR) Logout

CUOnline

Home | Attendance | My Docs | Groups | Reports | Notifications | Assignments

My Courses

Please select a class for which the attendance is to be marked.

Courses List

Course No	Title	Program	Total Students
CSD478	Data Warehousing	BSE	41
CSC371	Database Systems I	BCS	51
CSC371	Database Systems I	BCS	38

Today's Lecture

- EER (Enhanced Entity Relationship)

What is an EER Model?

- Enhanced Entity Relationship (EER) – Data Modeling
- EER shows complex relationships between objects in a database (multimedia, geographical).
- Concepts of subclasses and superclasses, specialization and generalizations.
- Put OOD Modeling concepts in ER diagram to form an EER model

Why EER Diagrams

- Overall, an EER diagram builds off of an ER diagram by including elements that allow for aggregating, generalizing, and specializing.
- EER diagrams, on the other hand, are perfect for taking a more detailed look at your information.
- When your database contains a larger amount of data it is best to turn to an enhanced model to more deeply understand your model.

Overall Process of Modeling

- Abstraction
- Use of some modeling discipline (Data Model)
- Use of a representation technique
  - Language
  - Diagramming
  - Tools
- Analysis of business rules/semantic constraints (these are typically beyond the capability of the data model)

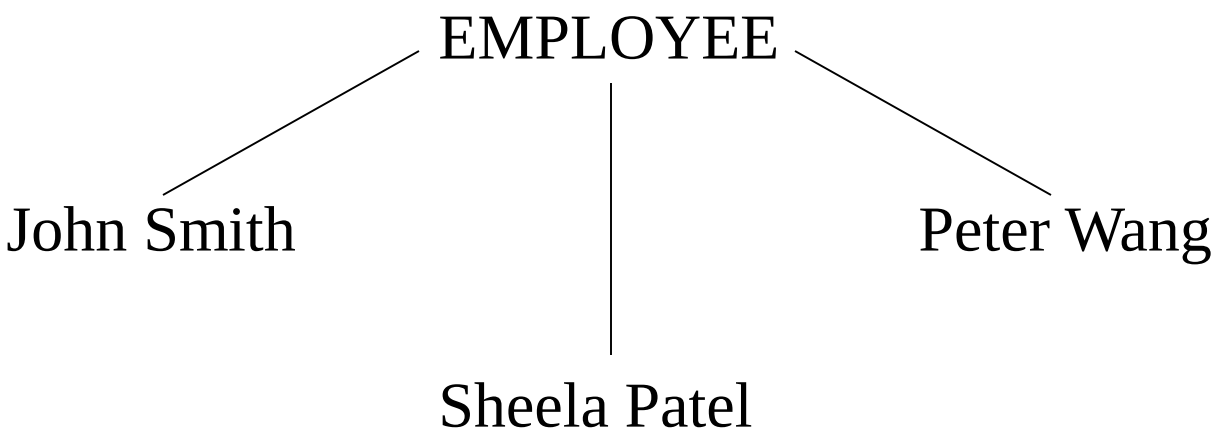
Types of Abstractions

Classification	A is a member of class B
Aggregation	B,C,D are aggregated into A A is made of/composed of B,C,D
Generalisation	B,C,D can be generalised into A, **B is-an A, C is- an A, D is-an A**
Specialisation	A can be specialised into B,C,D B,C,D (special cases of A)

Classification Abstraction

- Relationship between a class and its members

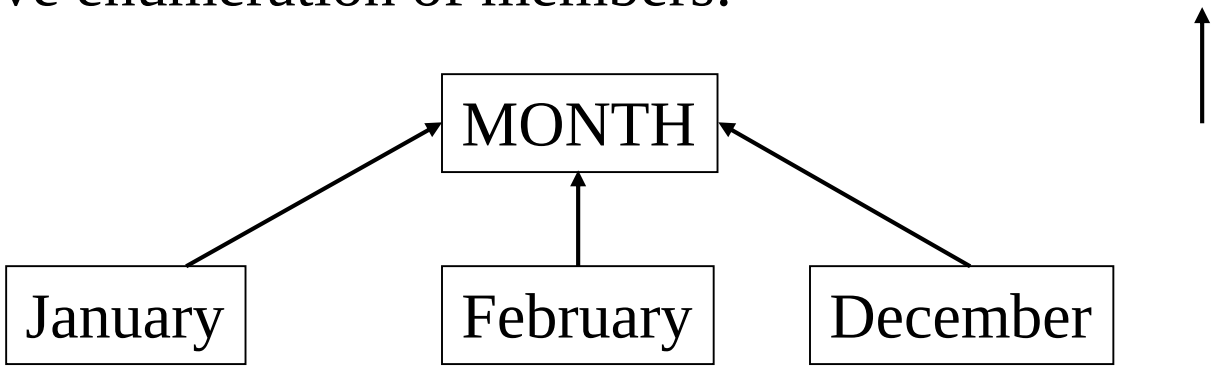
  John Smith, Sheela Patel, and Peter Wang are all employees. They are all members of a class: EMPLOYEE class



- Each individual employee is a member of the class EMPLOYEE

Classification Abstraction (contd.)

Exhaustive enumeration of members:



January, February etc. are members of the class "MONTH"

Represents "member-of" relationship

In object-oriented modeling :

MONTH : an Object type or class

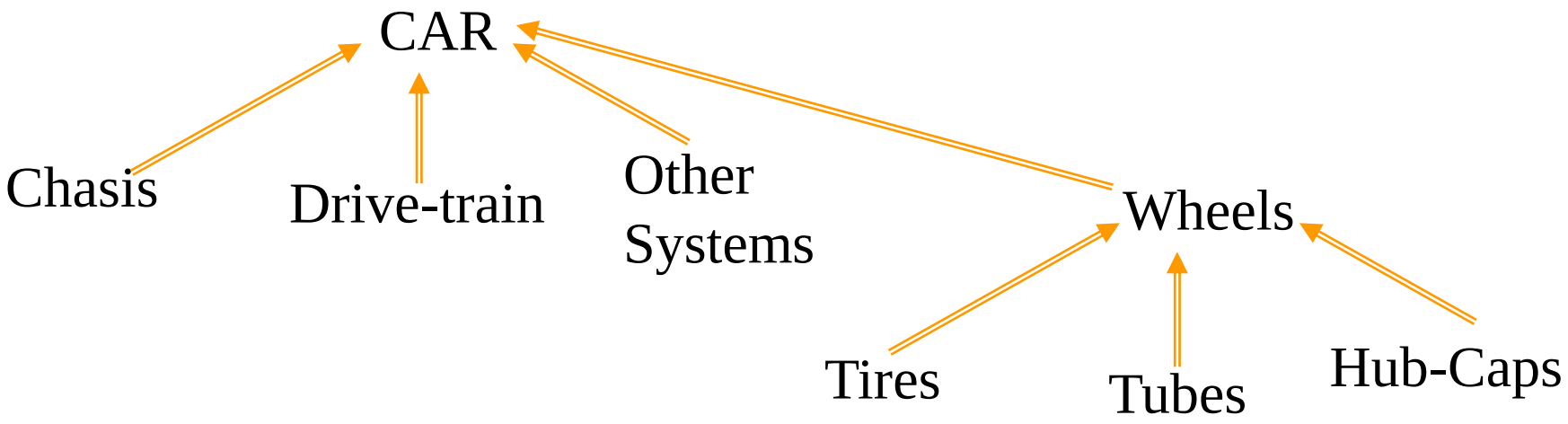
January … December : objects that belong to class MONTH

Classification - Class Properties

- Collection of similar entities or concepts into a higher level concept
- EMPLOYEE class collects all employees into one class
- A class has properties called “class properties”
- EMPLOYEE class has **class properties** - e.g., average salary, total number of employees
- Each member has values for own properties (e.g. name, address, salary): called **member properties**

Aggregation Abstraction

Defines a new class from a set of classes which are identified as components of the root class



represents IS-PART-OF (component) relationship

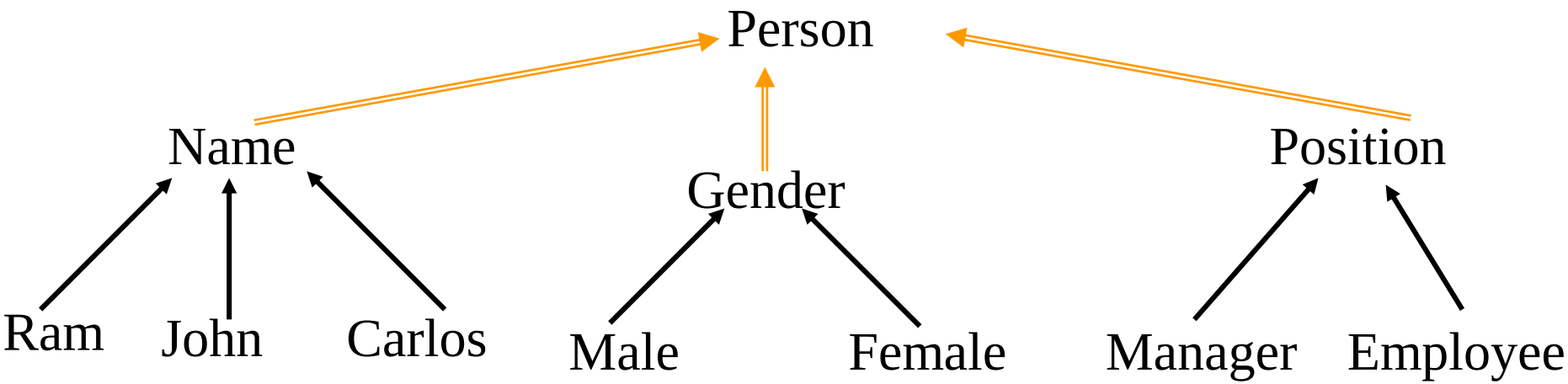
Root class: CAR
Component Classes: Chassis, Drive-Train, Other Systems, Wheels
Root class: Wheels
Component Classes: Tires, Tubes, Hub-Caps

Classification and Aggregation

Classification and Aggregation are used to build schemas

Example: class Person

Representation:



Name, Gender, and Position aggregate into Person. They are classes themselves.

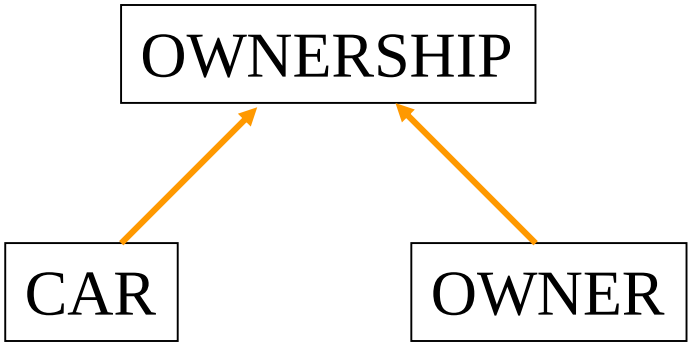
Ram, John, Carlos are classified into Name or Name is a classification of Ram, John, Carlos

Two Contexts for Aggregation

Aggregate two or more classes into a higher level concept. It may be considered a relationship or association between them.

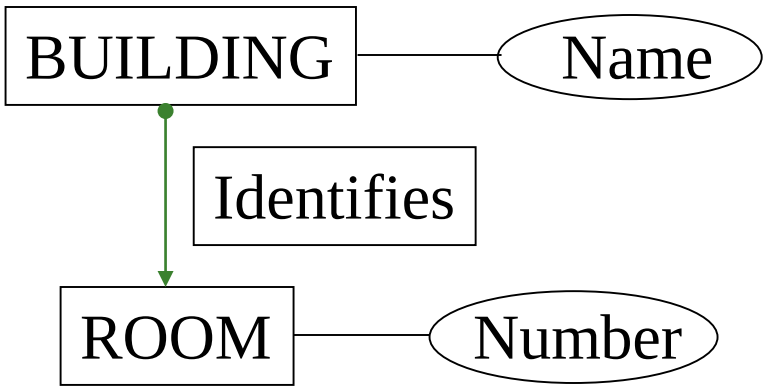
Context1: CAR is an aggregate (composition) of Chassis, Drive-train, Other Systems, Wheels.

Context 2: OWNERSHIP is an aggregate (relationship) of CAR and OWNER



Identification

Identifies one concept (an instance of it) from another concept.



Data Abstraction (contd.)

Process of hiding (suppressing) unnecessary details so that the high level concept can be made more visible.

This enables programmers, designers, etc., To communicate easily and to understand the application's data and functional requirements easily.

TYPES OF ABSTRACTION

Classification:	IS-A-MEMBER-OF
Aggregation:	IS-MADE-OF, IS-ASSOCIATED-WITH
Composition:	IS-MADE-OF (similar to aggregation) (A COMPRISES B,C,D)
Identification:	IS-IDENTIFIED-BY
Generalisation:	IS-A IS-LIKE IS-KIND-OF

Cardinality Constraints

Cardinality Constraint: Quantification of the relationship between two concepts or classes (a constraint on aggregation)

MINIMUM (A,B) = n

At a minimum, one instance of A is related to at least n instances of B.

n = 0	MIN(A,B) = 0	MIN(Person, Car) = 0
n = 1	MIN(A,B) = 1	MIN(Cust, Ship-address) = 1
n = inf.	Min (A,B) = inf.	Not possible
n = x (fixed)	MIN(A,B) = x	MIN(Car, Wheels) = 4

Cardinality Constraints (contd.)

MAXIMUM (A,B) = n

At a maximum, one instance of A is related to at most n instances of B.

n = 0	MAX(A,B) = 0	DOES NOT ARISE
n = 1	MAX(A,B) = 1	MAX(Cust, Ship-address) = 1
n = inf.	MAX(A,B) = inf.	MAX(Cust, Orders) = inf.
n = x (fixed)	MAX(A,B) = x	MAX(Stud, Course) = 6

Participation Constraints

MIN (A,B) = 0	Optional Participation
MIN (A,B) = 1	Mandatory Participation
MAX (A,B) = 0	No Participation
MIN (A,B) = x, MAX (A,B) = y	Range Constrained Participation

In Next Lecture

- Enhanced Entity Relationship Diagram (EERD) and Inheritance in Schema Modeling (Part 2)

Thanks