Fundamentals of Data Engineering
(COE848)

Midterm Exam

Feb. 26, 2019

This is a CLOSED BOOK exam. Textbooks, notes, laptops, calculators, personal digital assistants, cell phones, and Internet access are NOT allowed.

This is a 120-minute exam.

There are 5 questions with a total of 100 marks.

Please read each question carefully and write your answers legibly in the space provided. You may do the questions in any order you wish, but please
USE YOUR TIME WISELY.

When you are finished, please hand in your exam paper and make sure you are **signed out**.
Good luck!

Student Name: _____
Student ID: _____
Score: _____ %

Question	Maximum Mark	Received
1	15	
2	20	
3	20	
4	20	
5	25	
Total	100	

Question 1: Explain the following terms in two or three sentences. [15 marks: 2.5 marks each]

a) Data Modeling

The process in which a real world entity transforms to a computable entity. It includes three levels of abstraction: i) Conceptual Level, ii) Logical Level, and iii) Physical Level

b) Transaction

A group one or more operations into a single unit of work.

c) Data Modification Language (DML)

Set of commands to manipulate data such as select, insert, update, and delete, also called "query language"

d) Database Management System (DBMS)

A software system that provides general purpose, efficient, convenient, & safe multi-user storage of and access to massive amounts of persistent data in which ACID properties are guaranteed.

e) Surrogate Key

One-attribute key whose value is system-generated, e.g., AUTONUMBER, i.e., 1, 2, 3, ..., usually named Id | RowId

f) Candidate Key

Candidate key is a possible key set whose attributes can accept no value (NULL)

Question 2: Explain ACID properties (Atomicity, Consistency, Isolation, Durability) with an example for each. [20 marks: 5 marks each]

I) *Atomicity: all-or-nothing execution of transaction, e.g., money transfer in online banking system where money withdrawal and deposit must be atomic.*

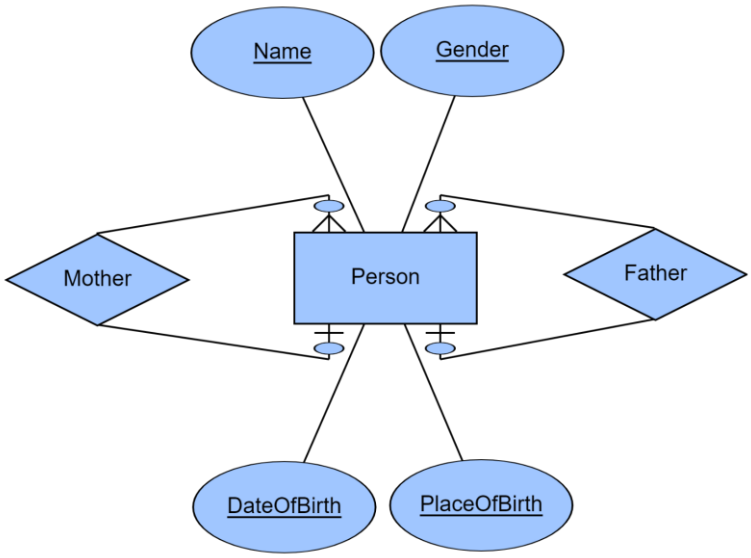
II) *Consistency: respecting constraints or expectations among data instances, e.g., if there is a constraint that every movie must have at least one director, then a transaction to insert a movie without a director must be rolled back.*

III) *Isolation: transaction appear to be executed as if no other transaction is executing at the same time, e.g., in multi-user environment where two users want to insert the same director if the director does not exist must result in only one insertion not two.*

IV) *Durability: once a transaction has committed, the effect must never be lost in case of function exceptions, program crash, or even system crash, e.g., after inserting a movie and its director, the information must be persistent in case of power loss.*

Question 3: Genealogy, also known as family tree, is the study of families and the tracing of their lineages and history. Assuming each person has one or unknown *biological* father and mother, do data modeling at *i)* conceptual (ER) and *ii*) logical (relational) levels in order to store genealogies in a relational database management system. Each person is uniquely identified by his/her name, gender, date of birth, and place of birth. [20 marks]

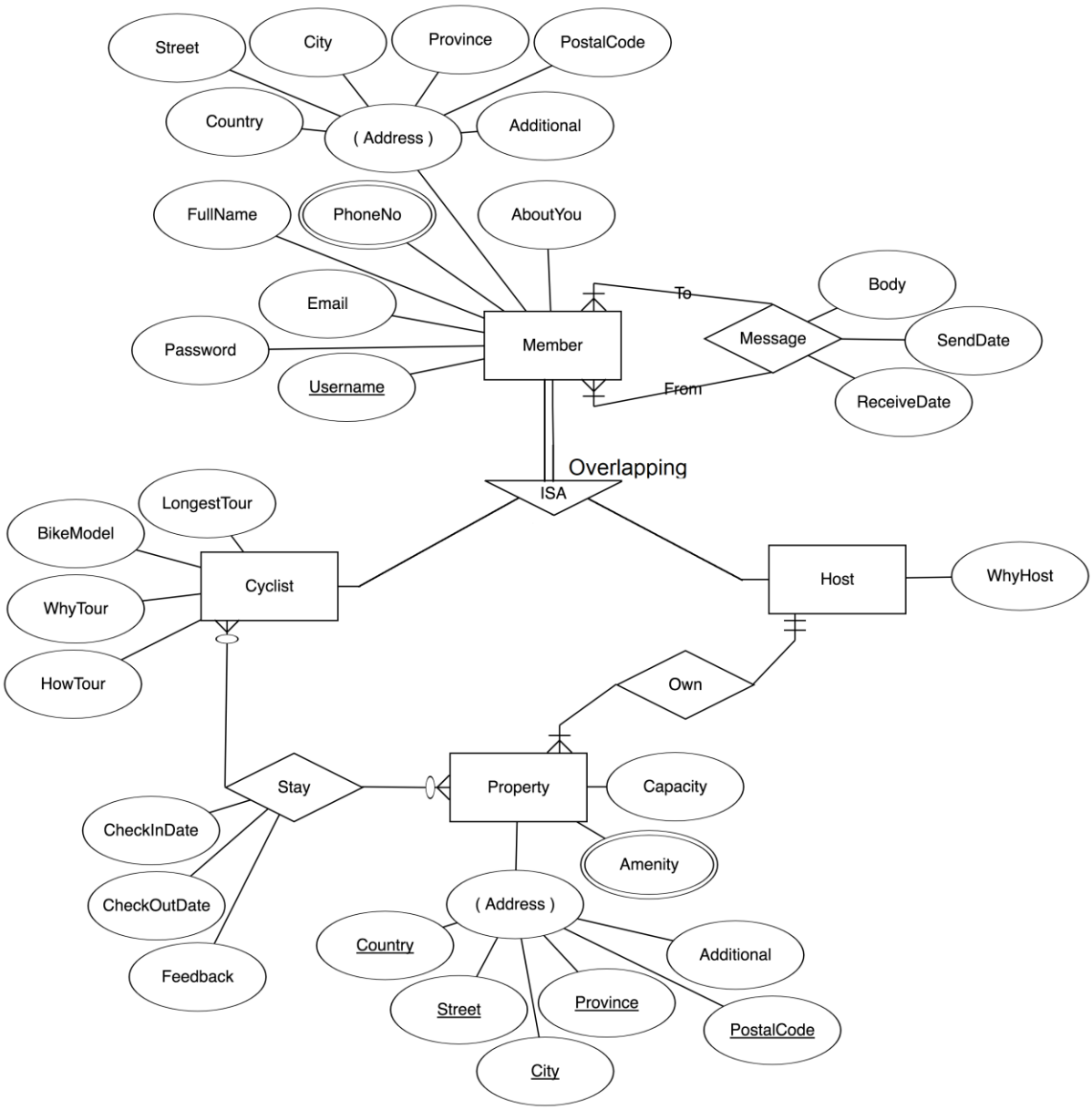
10 marks for ER diagram



10 marks for relational model

R1: Person (Id, Name, Gender, DateOfBirth, PlaceOfBirth, ***FatherId, MotherId****)*
CK= {Name, Gender, DateOfBirth, PlaceOfBirth}
FK1={FatherId}, FK2={MotherId}

Question 4: (www.warmshowers.org) The Warm Showers Community is a free worldwide hospitality exchange for touring cyclists. Members are either *i*) cyclists who do long distance touring with bike, or *ii*) hosts who offer homestays and *Warm Shower* for free, or both. In order to arrange homestays, cyclist directly request lodging from hosts. The platform is a gift economy; hosts are not allowed to charge for lodging. Members set up an online identity, and after leaving comments on their experiences with lodging (feedback), develop a reputation for hosts. Suppose you are given the data modeling at conceptual level (ER diagram) for such software system. Model data for the system in relational database management system (RDBMS) at logical level. That is, convert the given ER diagram to the relational model. Use surrogate key as primary key. List candidate keys (CK) and foreign keys (FK) for each relation, if any, in your relational model. [20 marks]



Note: *There are different approaches to convert ISA relationship. Herein, the "relation-per-all" approach has been used and superclass primary key becomes foreign key to subclasses. Each relation receives 2 marks plus 2 marks for ISA conversion.*

Entity Sets:

R1: Member (Id, Username, Password, Email, FullName, AddressCountry, AddressStreet, AddressCity, AddressProvince, AddressPostalCode, AddressAdditional, AboutYou),
CK={Username}

R2: Cyclist (Id, HowTour, WhyTour, LongestTour, BikeModel, MemberId),
FK= {MemberId},
CK= {MemberId}

R3: Host (Id, WhyHost, MemberId),
FK= {MemberId},
CK= {MemberId}

R4: Property (Id, AddressCountry, AddressStreet, AddressCity, AddressProvince, AddressPostalCode, AddressAdditional, Capacity, ***HostId****),*
FK= {HostId} & Mandatory
CK= {AddressCountry, AddressStreet, AddressCity, AddressProvince, AddressPostalCode}

Multivalued Attributes:

R5: MemberPhone(Id, PhoneNo, ***MemberId****),*
FK={MemberId},
CK= {PhoneNo, MemberId}

R6: Amenity(Id, Title),
CK= {Title}

R7: PropertyAmenity(Id, ***PropertyId, AmenityId****),*
FK1={PropertyId}, FK2={AmenityId},

Relationships:

R8: Message(Id, Body, SendDate, ReceiveDate, ***FromMemberId****,* ***ToMemberId****),*
FK1={FromMemberId}, FK2={ToMemberId},
CK= {SendDate, FromMemberId, ToMemberId}

R9: Stay(Id, CheckInDate, CheckOutDate, Feedback, ***CyclistId****,* ***PropertyId****),*
FK1={CyclistId}, FK2={PropertyId},
CK= {CheckInDate, CyclistId, PropertyId}

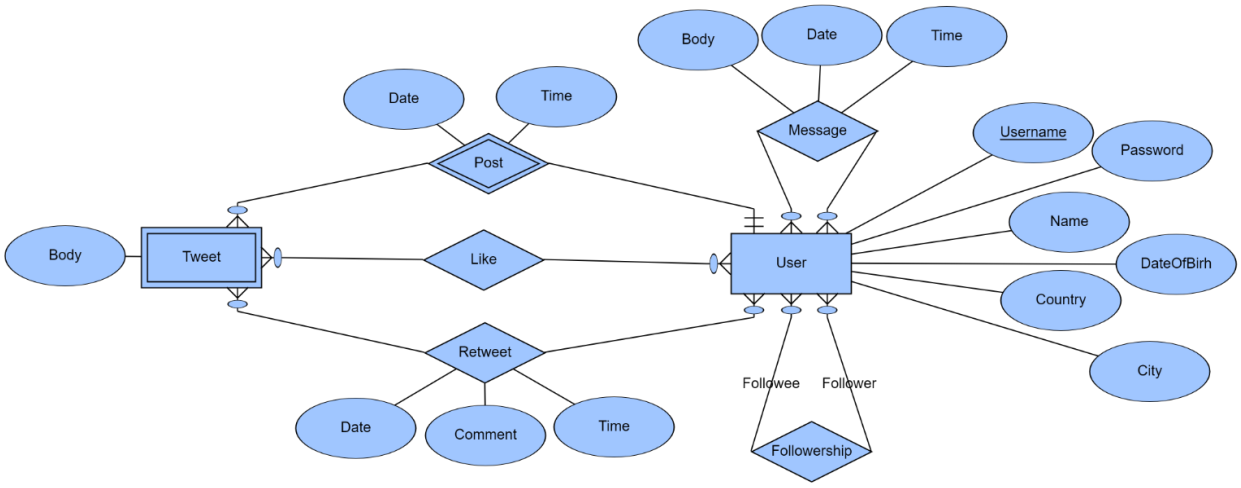
"Own" relationship has been applied in R4 by ***HostId*** *foreign key.*

Question 5: (twitter.com) Twitter is an online social networking service on which users post short messages (140 characters) known as 'tweet'. A tweet might be liked or shared again by a user (retweet) which is a great way to pass along news and interesting discoveries on Twitter. Besides, a user can follow other users (follower) to see their tweets in her own timeline. Users are also able to send direct messages to other users. Suppose you are given the following requirements for this software system:

- (User) A user selects a username, password, name, city, country, date of birth when signing up,
- (Tweet) A tweet has body,
- (Posting) A user can post no tweet or more than one tweet at a date and time,
- (Retweeting) A user can retweet another tweet at a date and time, adding her own comment,
- (Liking) A tweet can be liked by no user or more than one user,
- (Following) A user can follow no or more than one user,
- (Message) A user can send one direct message to one or more than one user at a date and time.

Model data for the system in relational database management system (RDBMS) at conceptual and logical levels, i.e., *i*) draw ER diagram and *ii*) convert ER diagram to relational model. List your assumptions and clearly indicate primary keys (PK) of entity sets and optionality (ordinality) and multiplicity (cardinality) of relationships in your ER diagram. In relational model, use surrogate key and then list candidate keys (CK) and foreign keys (FK) for each relation, if any. [25 marks]

15 marks for ER diagram: 2 marks each strong entity set, 2 marks each relationship, 3 marks for Tweet as weak entity set



10 marks for relational model: 1.5 each relation but 2.5 marks for Tweet relation

Entity sets:
R1: User(Id, Username, Password, Name, DateOfBirth, PlaceOfBirth, City, Country)
CK={Username, Password, Name, DateOfBirth, PlaceOfBirth}

*R2: Tweet(Id, Body, Date, Time, **UserId**)*
FK={UserId}
CK={UserId, Date, Time}

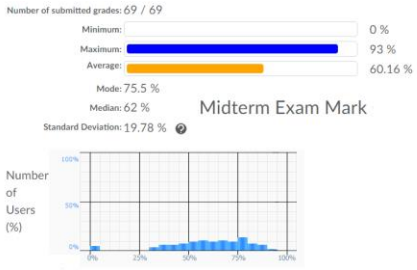
Relationships:
*R3: Like(Id, **UserId**, **TweetId**)*
FK1={UserId}, FK2={TweetId}
CK={UserId, TweetId}

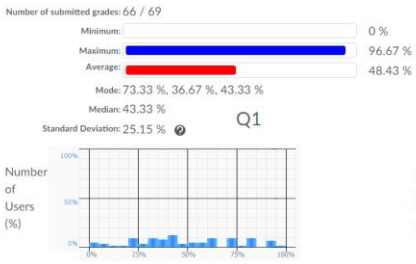
*R4: Retweet(Id, Comment, Date, Time, **UserId**, **TweetId**)*
FK1={UserId}, FK2={TweetId}

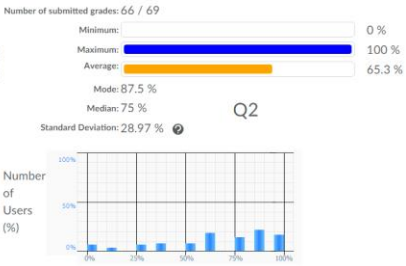
*R5: Followership(Id, **FollwerId**, **FolloweeId**)*
FK1={FollowerId}, FK2={FolloweeId}
CK={FollowerId, FolloweeId}

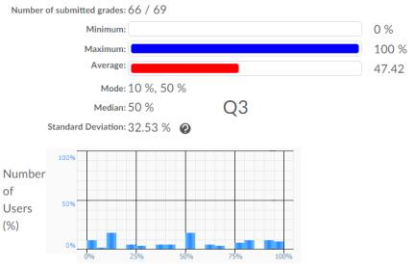
*R6: Message(Id, Body, Date, Time, **SenderId**, **ReceiverId**)*
FK1={SenderId}, FK2={ReceiverId}

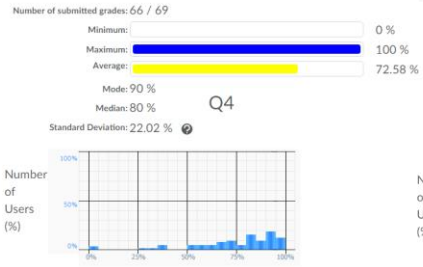
*"Post" relationship has been already applied in R2 by **UserId** foreign key.*

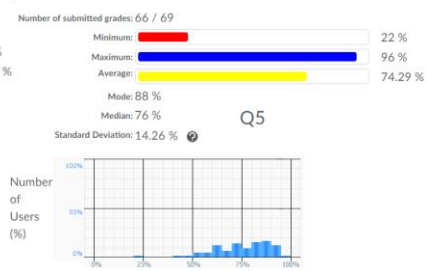

Number of submitted grades: 69 / 69
Minimum: 0 %
Maximum: 93 %
Average: 60.16 %
Mode: 75.5 %
Median: 62 %
Standard Deviation: 19.78 %
Midterm Exam Mark
Number of Users (%)


Number of submitted grades: 66 / 69
Minimum: 0 %
Maximum: 96.67 %
Average: 48.43 %
Mode: 73.33 %, 36.67 %, 43.33 %
Median: 43.33 %
Standard Deviation: 25.15 %
Q1
Number of Users (%)


Number of submitted grades: 66 / 69
Minimum: 0 %
Maximum: 100 %
Average: 65.3 %
Mode: 87.5 %
Median: 75 %
Standard Deviation: 28.97 %
Q2
Number of Users (%)


Number of submitted grades: 66 / 69
Minimum: 0 %
Maximum: 100 %
Average: 47.42
Mode: 10 %, 50 %
Median: 50 %
Standard Deviation: 32.53 %
Q3
Number of Users (%)


Number of submitted grades: 66 / 69
Minimum: 0 %
Maximum: 100 %
Average: 72.58 %
Mode: 90 %
Median: 80 %
Standard Deviation: 22.02 %
Q4
Number of Users (%)


Number of submitted grades: 66 / 69
Minimum: 22 %
Maximum: 96 %
Average: 74.29 %
Mode: 88 %
Median: 76 %
Standard Deviation: 14.26 %
Q5
Number of Users (%)