

Search content

Search

[View content explorer](#)

[Configure zyBook](#)

1. SET 1

Empty

2. SET 1 - 1:INTRODUCTION

63%

3. SET 1 - 2:DATA MODELING - WHAT - ERD MODEL

45%

46%

4. SET 1 - 3:DATA MODELING HOW - RELATIONAL MO...

12%

15%

5. SET 1 - 4:ERD to RELATIONAL

Hidden

6. SET 1 - 5:NORMALIZATION

Hidden

## COSC 3380: Database Systems



Spring 2024

Class information

100 students subscribed [Subscription instructions](#)

[Manage](#)

Add teaching assistants, sections, email domains, & drop/manage student sections

Instructors

Victoria Hilford

Teaching assistants

Jordan Yu

Alvaro Urtaza

Fernando Ramirez

Class sections

No class sections

**COSC 3380 Spring 2024**

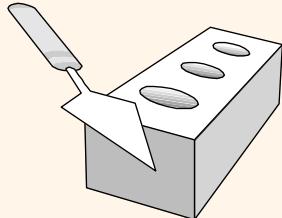
**Database Systems**

**M & W 4:00 to 5:30 PM**

Prof. **Victoria Hilford**

**PLEASE TURN your webcam ON (must have)**

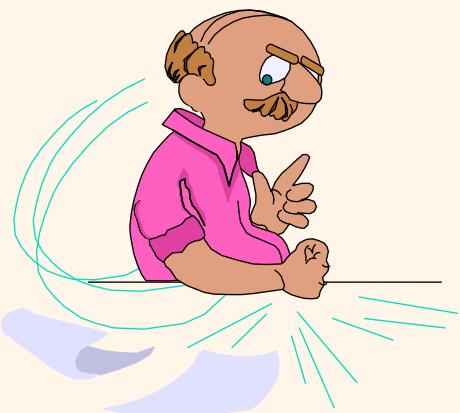
**NO CHATTING during LECTURE**



# COSC 3380

## 4 to 5:30

**PLEASE  
LOG IN  
CANVAS**



Please close all other windows.

01.24.2024

ZyBook SET 1 - 3

(3 - We)

Set 1

LECTURE 3 DATA MODELING - HOW - RELATIONAL MODEL

01.29.2024

ZyBook SET 1 - 4

(4 - Mo)

Set 1

LECTURE 4 ERD to RELATIONAL

01.31.2024

ZyBook SET 1 - 5

(5 - We)

Set 1

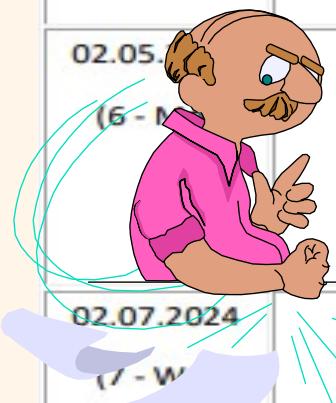
LECTURE 5 NORMALIZATION

02.05.

(6 - Th)

EXAM 1 Practice

(PART of 20 points)



02.07.2024

(7 - Fr)

TA Download

ZyBook SET 1 Sections

(4 PM)

(PART of 30 points)

EXAM 1 Review

(PART of 20 points)

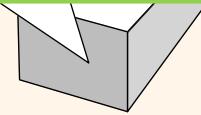
02.12.2024

(8 - Mo)

EXAM 1

(PART of 50 points)

# From 4:00 to 4:07 PM – 7 minutes.



01.24.2024  ZyBook SET 1 - 3 Set 1  
(3 - We) LECTURE 3 DATA MODELING - HOW - RELATIONAL MODEL

 CLASS PARTICIPATION 20 points 20% of Total + :

## CLASS 3

 Class 3 BEGIN PARTICIPATION  :

Not available until Jan 24 at 4:00pm | Due Jan 24 at 4:07pm | 100 pts

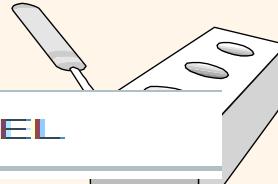
**VH, publish**

This is an synchronous online class.

Attendance is required.

Recording or distribution of class materials is prohibited.

1. At the beginning of selected classes there is an assessment in the first 10 minutes. (beige BOX in the Detailed Syllabus)
2. At the end of selected classes there is an assessment in the last 10 minutes. (blue BOX in the Detailed Syllabus)
3. ZyBook sections will be downloaded and used for 30% of Total Score on the dates specified in the Detailed Syllabus.
4. EXAMS are in CANVAS. No late EXAMS.



## Practice Questions

A database designer interviews a sales team that will be using a new database. During the interview, the designer documents entities as \_\_\_\_\_, and relationships as \_\_\_\_\_.

- a. nouns, nouns
- b. verbs, nouns
- c. nouns, verbs
- d. verbs, verbs

?????



## Practice Questions

Which of the following is a set of things in entity-relationship modeling?

- a. Attribute
- b. Relationship
- c. Entity
- d. Repository

?????



## Practice Questions

A database team plans the creation of a new database. During which phase does the team capture data requirements?

- a. Modeling
- b. Implementation
- c. Normalization
- d. Analysis

?????

## Practice Questions

When an entity-relationship model diagram is implemented within SQL,

**entities** typically become \_\_\_, and **relationships** typically become \_\_\_.

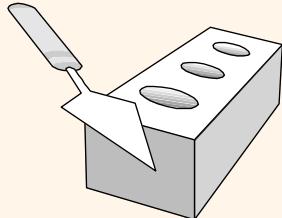
a. tables, primary keys

b. tables, foreign keys

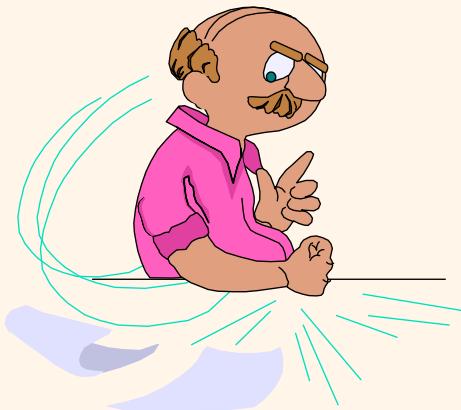
c. rows, tables

d. foreign keys, rows

?????



# COSC 3380



## Class 3

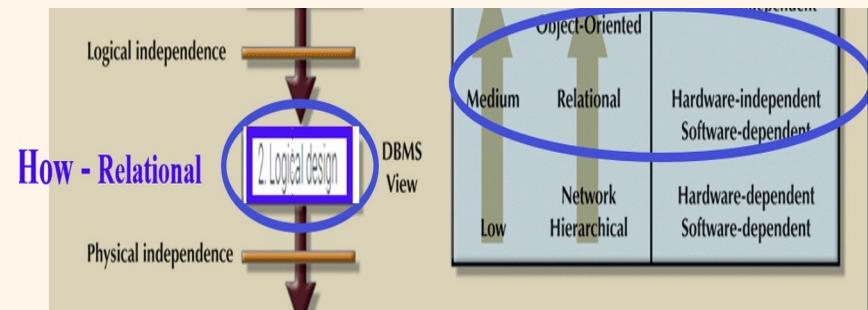
01.24.2024

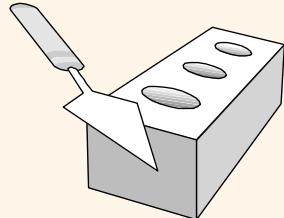
ZyBook SET 1 - 3

(3 - We)

Set 1

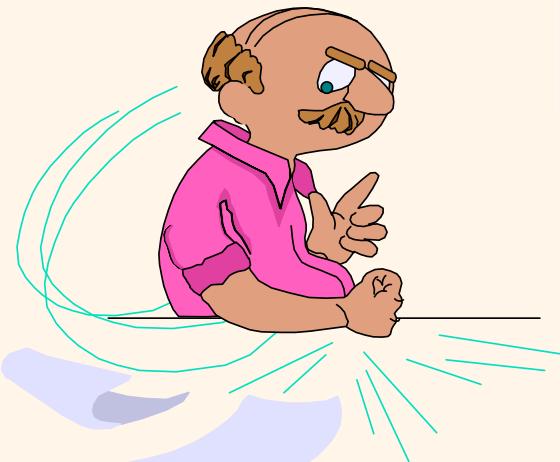
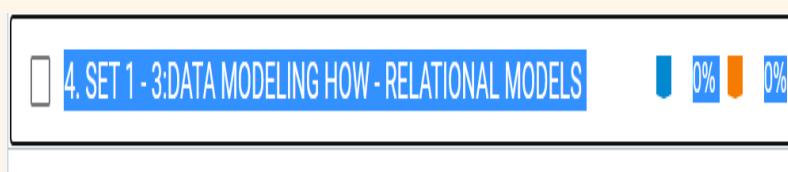
LECTURE 3 DATA MODELING - HOW - RELATIONAL MODEL



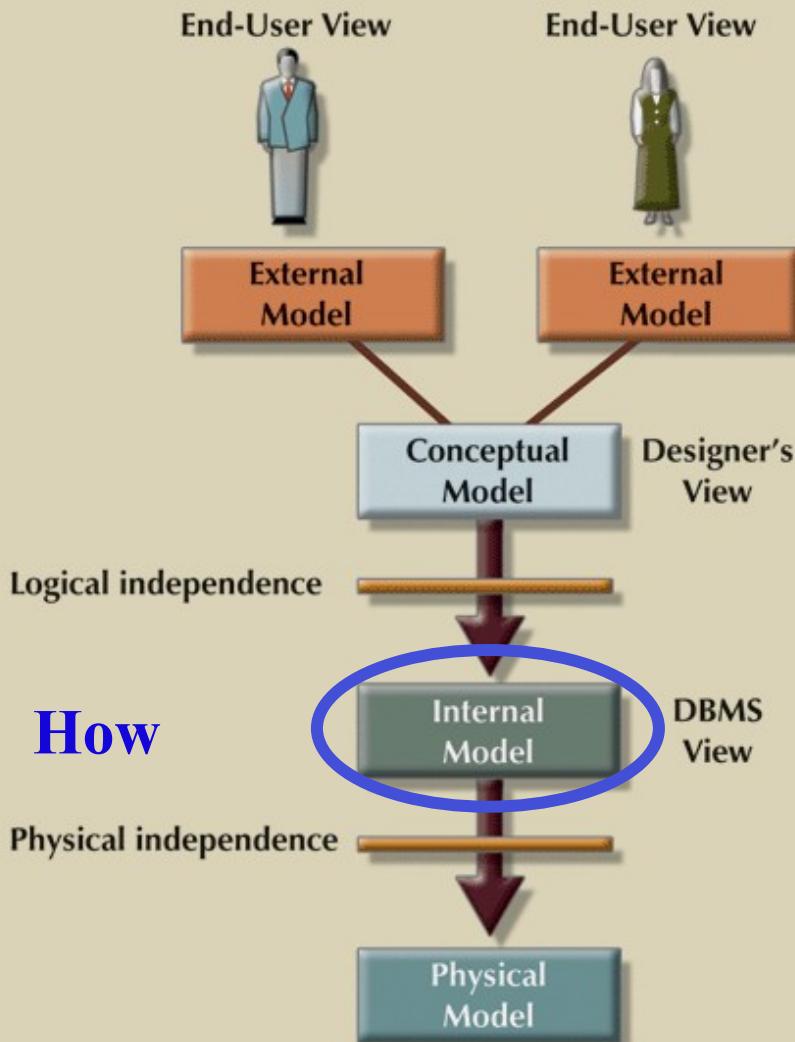


## Lecture 3

# The *Relational* Model **HOW** Data

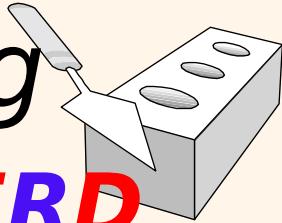


## Data abstraction levels



Degree of Abstraction	Characteristics
High ER Object-Oriented	Hardware-independent Software-independent
Medium Relational	Hardware-independent Software-dependent
Low Network Hierarchical	Hardware-dependent Software-dependent

# HOW Data Modeling



1. **MUST:** Start **FROM** the **PREVIOUS ERD**
2. **MUST:** USE **BELLOW** Relational Language

## **Relation:Schema :**

Students (sid: string, name: string, login: string, age: integer, gpa: real)

Enrolled(cid: string, grade: string, stuid :string)

OR

## **Pictorial View: Students**

>	<u>sid</u>	name	login	age	gpa
Enrolled					
	<u>cid</u>	grade	<u>stuid</u>		

OR

TABLE Employees (ssn CHAR(11),  
name CHAR(30),  
lot INTEGER,  
PRIMARY KEY (ssn))

TABLE Departments (did INTEGER,  
dname CHAR(20),  
budget REAL,  
PRIMARY KEY (did))

BEST

# Relations and Their Characteristics

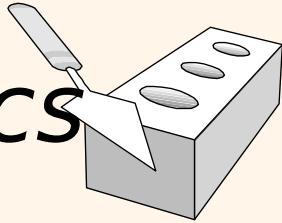


FIGURE 3.1 STUDENT table attribute values

Database name: Ch03\_TinyCollege

Table name: STUDENT

STU_NUM	STU_LNAME	STU_FNAME	STU_INIT	STU_DOB	STU_HRS	STU_CLASS
321452	Bowser	William	C	12-Feb-1975	42	So
324257	Smithson	Anne	K	15-Nov-1981	81	Jr
324258	Brewer	Juliette		23-Aug-1969	36	So
324269	Oblonski	Walter	H	16-Sep-1976	66	Jr
324273	Smith	John	D	30-Dec-1958	102	Sr
324274	Katinga	Raphael	P	21-Oct-1979	114	Sr
324291	Robertson	Gerald	T	08-Apr-1973	120	Sr
324299	Smith	John	B	30-Nov-1986	15	Fr

STUDENT table,  
continued



	STU_GPA	STU_TRANSFER	DEPT_CODE	STU_PHONE	PROF_NUM
►	2.84	No	BIOL	2134	205
	3.27	Yes	CIS	2256	222
	2.26	Yes	ACCT	2256	228
	3.09	No	CIS	2114	222
	2.11	Yes	ENGL	2231	199
	3.15	No	ACCT	2267	228
	3.87	No	EDU	2267	311
	2.92	No	ACCT	2315	230

STU\_HRS = Credit hours earned  
STU\_CLASS = Student classification  
STU\_DOB = Student date of birth

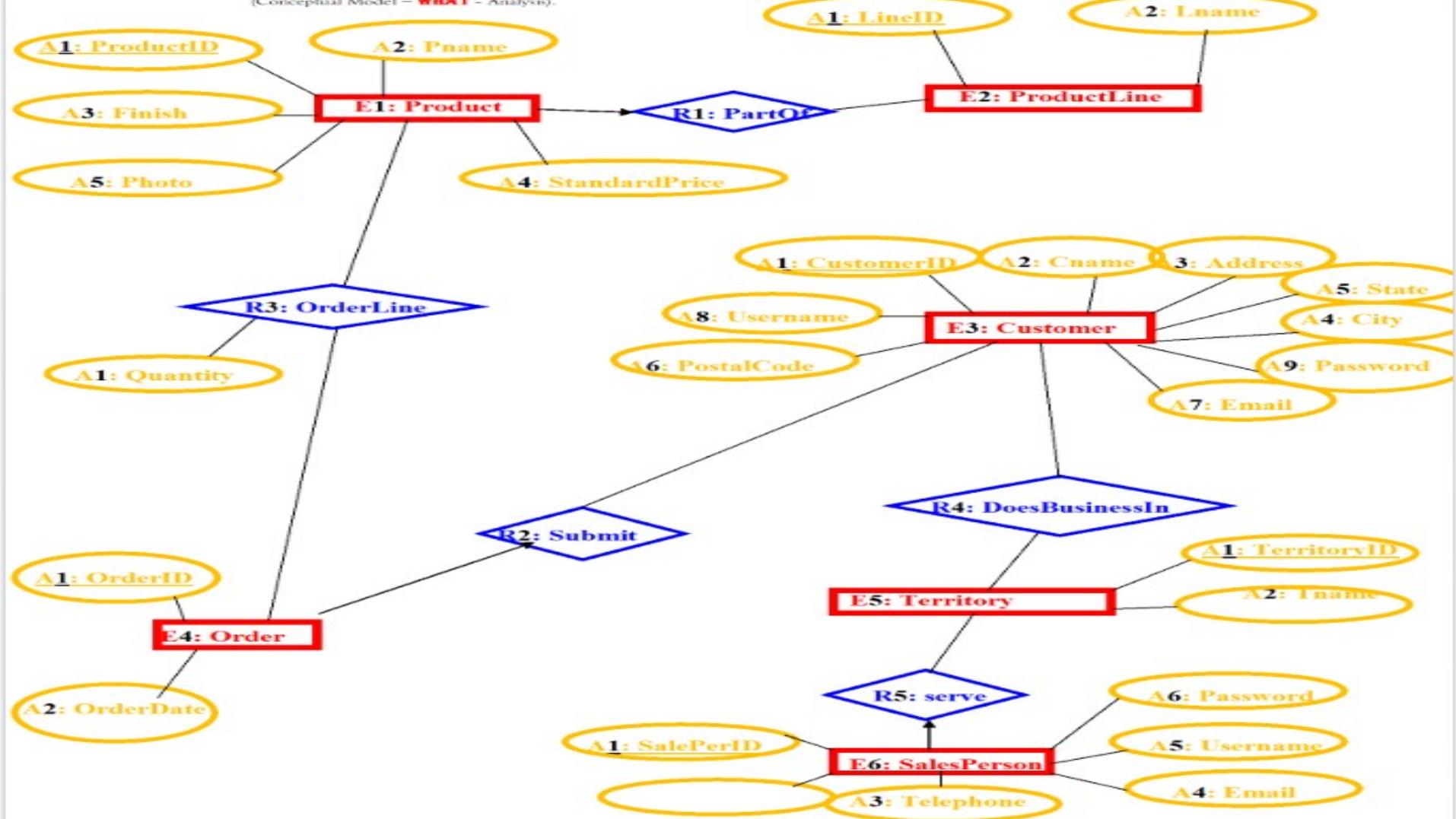
STU\_GPA = Grade point average  
STU\_PHONE = 4-digit campus phone extension  
PROF\_NUM = Number of the professor  
who is the student's advisor

How many attributes?

# ERD Model – WHAT



a. (50 points – separate document) Using the above **COSC3380 Data Requirements** create the **ERD** Model (must fit on one page). (Conceptual Model – **WHAT** – Analysis).



ERD Model – WHAT 6 Es; 5 Rs,  $(5 + 2 + 9 + 2 + 2 + 6) = 26$  As

Sent: Sunday, February 03, 2013 2:15 PM

To: Hilford, Victoria

Hi Dr. H,

I had to ask my brother about the details. I remembered telling you about it, but didn't remember the scenario completely.

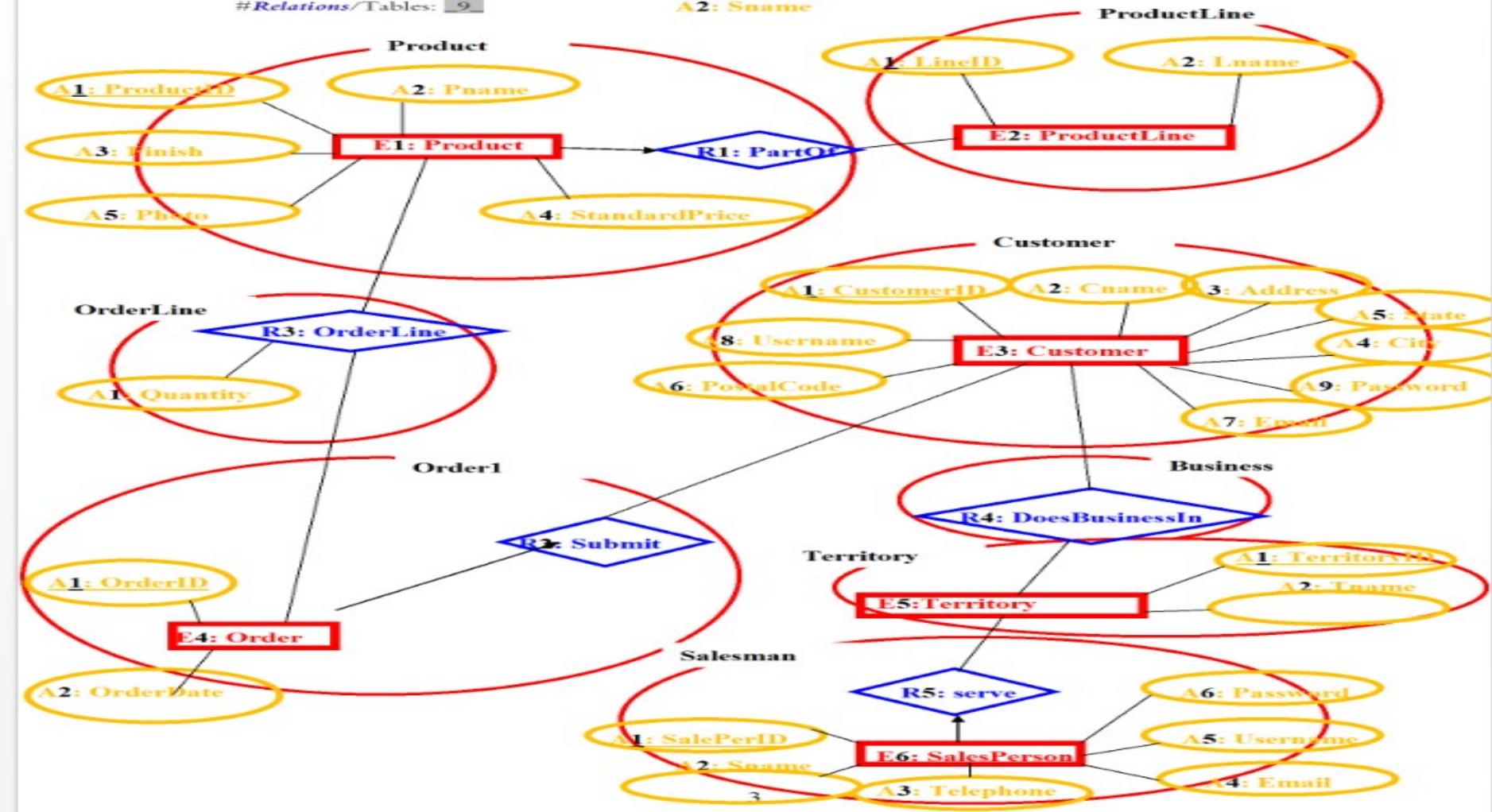
The contract was to sell a particular data set to the FBI. The contract value was \$70,000 per month, \$840k per year. This type of contract can be renewed or extended for years and years.

The data model was to be constructed per the specifications of the bureau and joined with one or more other data sets. Care was not taken to review the data model, so it did not reflect the process the bureau was modeling. When I told you about it, my brother thought they were going to be able to correct it and eventually make the sale, with four to six months of delay from the original starting date. They were not successful in their effort. The contract was cancelled. The economic damage was a minimum of \$840,000 for a very small firm.

# ERD Model – WHAT TO Relational Model - HOW mapping Next class – Step 1

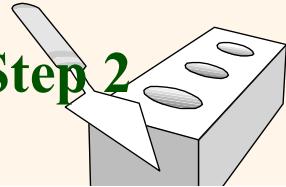


b. (30 points) **COSC3380 Relational Model** (must fit on one page)  
 (Internal Model – **HOW** – Design).  
 #Relations/Tables: 9



## Relational Model – HOW TABLES/RELATIONS

# Relational Model – HOW – Next class Step 2



## TABLE format

```
TABLE Customer (
    CustomerID      CHAR(20),
    Cname           CHAR(20),
    Address          CHAR(20),
    State            CHAR(20),
    City             CHAR(20),
    PostalCode       CHAR(15),
    Email            CHAR(20),
    UserName         CHAR(20),
    Password         CHAR(20),
    PRIMARY KEY      (CustomerID))

TABLE ProductLine (LineID           CHAR(30),
                    Lname            CHAR(30),
                    PRIMARY KEY(LineID))

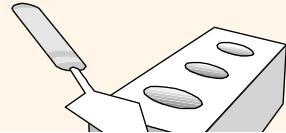
TABLE Order (
    OrderID          CHAR(20),
    orderDate        DATE,
    CustomerID       CHAR(20),
    PRIMARY KEY      (OrderID),
    FOREIGN KEY      (CustomerID) REFERENCES Customer)

TABLE Product (
    ProductID        CHAR(20),
    Pname            CHAR(20),
    Finish           CHAR(20),
    StandartPrice   FLOAT,
    Photo             IMAGE,
    LineID           CHAR(20),
    PRIMARY KEY      (ProductID),
    FOREIGN KEY      (LineID) REFERENCES ProductLine)

TABLE OrderLine (
    Quantity          INTEGER,
    SalePrice         FLOAT,
    ProductID         CHAR(20),
    OrderID           CHAR(20),
    PRIMARY KEY      (ProductID, OrderID),
    FOREIGN KEY      (ProductID) REFERENCES Product,
    FOREIGN KEY      (ProductLineID) REFERENCES Order)
```

## Relational Model – HOW TABLES/RELATIONS

# Relational Model – HOW – Next class Step 2



```
TABLE Territory (
    TerritoryID      CHAR(30),
    Tname            CHAR(30),
    PRIMARY KEY (TerritoryID))

TABLE Business (
    CustomerID       CHAR(30),
    TerritoryID     CHAR(30),
    PRIMARY KEY (CustomerID, TerritoryID),
    FOREIGN KEY (CustomerID) REFERENCES Customer,
    FOREIGN KEY (TerritoryID) REFERENCES Territory)

TABLE SalesMan (
    SalePerID        CHAR(30),
    Sname             CHAR(30),
    Telephone         CHAR(15),
    Email              CHAR(30),
    Username          CHAR(30),
    Password          CHAR(30),
    TerritoryID       CHAR(30),
    PRIMARY KEY        (SalePerID),
    FOREIGN KEY (TerritoryID) REFERENCES Territory)

TABLE PriceUpdates (
    Uname             CHAR(30),
    changeDate        DATE,
    OldPrice          FLOAT,
    NewPrice          FLOAT,
    PRIMARY KEY        (Uname, OldPrice))
```

# Relational Model – HOW

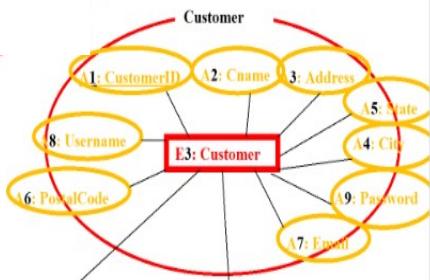


TABLE format

```
TABLE Customer (
    CustomerID      CHAR(20), A1
    Cname           CHAR(20), A2
    Address          CHAR(20), A3
    State            CHAR(20), A4
    City             CHAR(20), A5
    PostalCode       CHAR(15), A6
    Email            CHAR(20), A7
    UserName         CHAR(20), A8
    Password         CHAR(20), A9
    PRIMARY KEY      (CustomerID))
```

# Relational Model – HOW

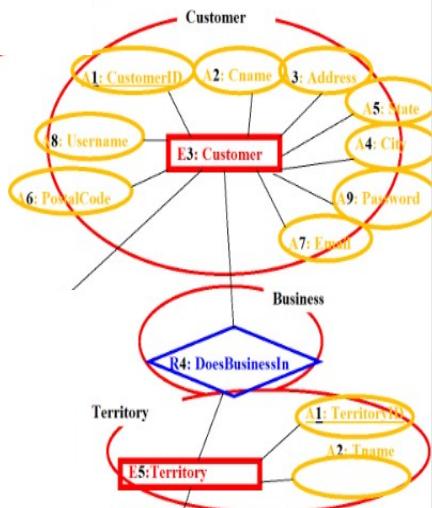
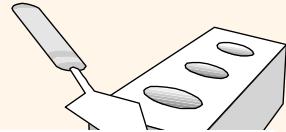


TABLE Business (

```
CustomerID      CHAR(30),    A1  
TerritoryID     CHAR(30),    A1  
PRIMARY KEY (CustomerID, TerritoryID),  
FOREIGN KEY (CustomerID) REFERENCES Customer,  
FOREIGN KEY (TerritoryID) REFERENCES Territory)
```

## 4. SET 1 - 3:DATA MODELING HOW - RELATIONAL MODELS



### 4.1 Relational model



### 4.2 Tables, columns, and rows



### 4.3 Null values



### 4.4 Primary and foreign keys



### 4.5 Referential integrity



## Relational data structure

The **relational model** is a database model based on mathematical principles, with three parts:

1. A data structure that prescribes how data is organized.
2. Operations that manipulate data structures.
3. Rules that govern valid relational data.

The relational data structure and operations are based on set theory. A **set** is a collection of values, or elements, with no inherent order. Sets are denoted with braces. Ex: {apple, banana, lemon} is the set containing three kinds of fruit. Since sets have no order, {apple, banana, lemon} is the same set as {lemon, banana, apple}.

The relational data structure is based on three mathematical concepts:

- A **domain** is a named set of possible database values, such as integers, dictionary words, or logical values TRUE and FALSE.
- A **tuple** is a finite sequence of values, each drawn from a fixed domain. Ex: (3, apple, TRUE) is a tuple drawn from domains (Integers, DictionaryWords, LogicalValues).
- A **relation** is a named set of tuples, all drawn from the same sequence of domains. Ex: The relation below is named **Grocery** and contains three tuples.

Since a relation is a set, the relation tuples have no inherent order.

PARTICIPATION  
ACTIVITY

4.1.2: Relations have no inherent order.

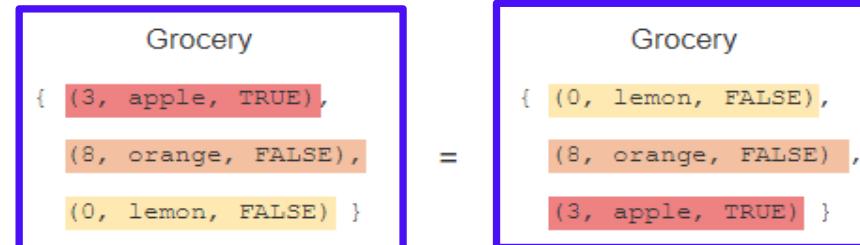




Table 4.1.2: Equivalent relational terms.

Mathematical	Database	Files
Domain	Data type	Data type
Tuple	Row	Record
Relation	Table	File
Attribute	Column	Field

[Feedback?](#)

## Terminology

Relational tuples differ from mathematical tuples in one respect. In mathematics, the order of tuple values is significant. Ex: The tuples (7, 2) and (2, 7) are different. In the relational model, the order of tuple values is not significant, since each value is associated with a different attribute name.

### PARTICIPATION ACTIVITY

#### 4.1.3: Relational data structure.

1) Which terms are commonly used in relational database processing?

- Tuple, relation, attribute
- Row, table, column
- Record, file, field

.....

The most commonly used terms in relational database are row, column, and table. Occasionally tuple, relation, attribute and record, file, field are also used.

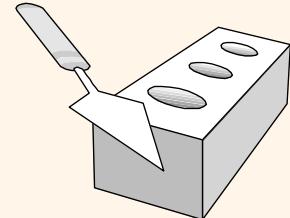
?????

**TA, Fernando Ramirez (A – L).**

**TA, Jordan, Yu (M – Z).**

**Please compare CANVAS vs. TEAMS Attendance.  
Print screens of students in CANVAS but not in the TEAMS meeting.  
(1.24.2024 Attendance X missing LastName.docx)**

# What is the **Relational Data** Model?

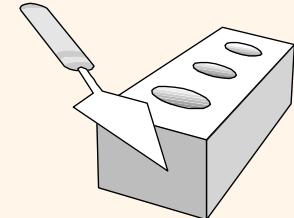


- ❖ A **Database** is a collection of one or more **Relations**, where each **Relation** is a **Table** with rows and columns.
- ❖ This simple tabular representation enables even **novice users** to understand the contents of a **Database**, and it permits the use of **simple, high-level languages to query the Data**.

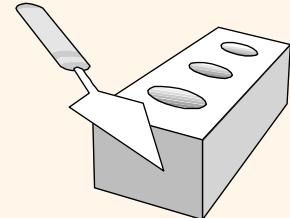


# **Relational Database :**

## *Definitions*



- ❖ **Relation:** made up of 2 parts:
  - **Schema** : specifies name of **Relation**, plus **name** and **type** of each **attribute**:  
  
Students( **key**: string, **name**: string, **login**: string, **age**: integer, **gpa**: real)
  - **Instance** : a **Table**, with rows and columns.  
**#rows = cardinality, #fields = degree / arity.**
  - ❖ Think of a **Relation Instance** as a **set** of **rows/tuples**.



# ***Relation: Schema :***

Students (sid: string, name: string, login: string, age: integer, gpa: real)

## ***Relation Instance of Students***

### **Relation Students**

key				
<u>sid</u>	name	login	age	gpa
53666	Jones	jones@cs	18	3.4
53688	Smith	smith@eeecs	18	3.2
53650	Smith	smith@math	19	3.8

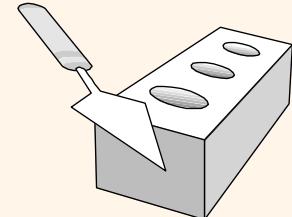
a *set* of rows/tuples

## **Relation Instance S1**

❖ Cardinality = 3, Degree = 5, all rows

# **Relational Database :**

## **Definitions**



- ❖ **Relational Database** : a set of **Relations**
- ❖ **Relational Database schema** = all the **Relations' Schemas**.

The University Enterprise **Database schema**:

Students(~~sid: string, name: string, login: string, age: integer, gpa: real~~)

Faculty(~~fid: string, fname: string, sal: real~~)

Courses(~~cid: string, cname: string, credits: integer~~)

Rooms(~~rno:integer, address: string, capacity: integer~~)

Enrolled(~~sid:integer, cid: integer, grade: string~~)

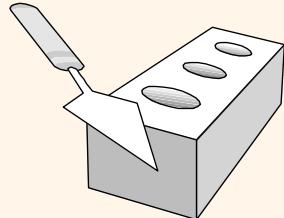
Teaches(~~fid: string, cid: string~~)

Meets(~~sid: string, rno:integer time: string~~)

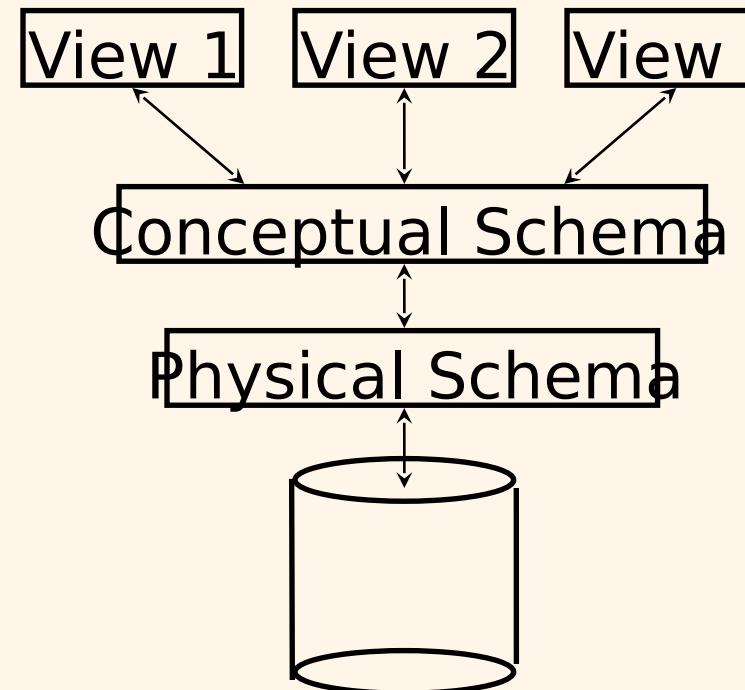
What is missing?

Keys

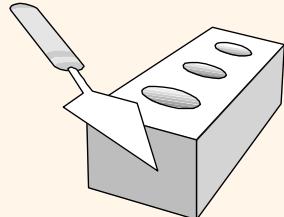
# Views



- ❖ A **view** is just a **Relation**, but we store a *definition*, rather than a set of **tuples**.
- ❖ Provides **logical** data independence.



- ❖ **Views** can be used to *present necessary information* (or a summary), while hiding details in underlying physical **Relation(s)**.



# Views

```
VIEW B-Students (name, sid, course)
AS SELECT S.sname, S.sid, E.cid
FROM Students S, Enrolled E
WHERE S.sid = E.studid AND E.grade = 'B'
```

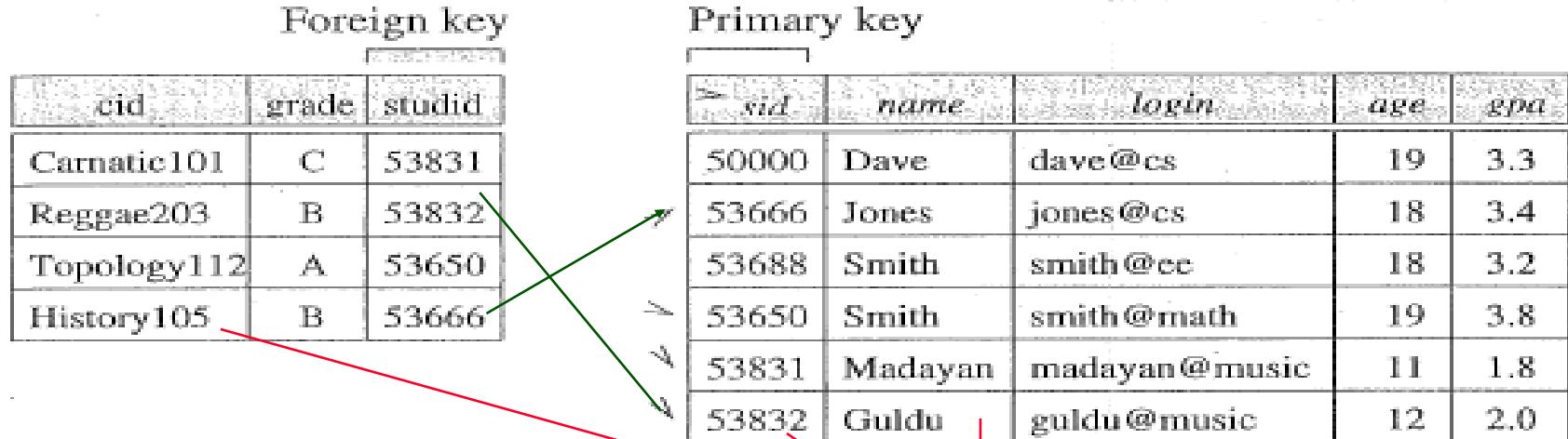


Figure 3.4 Referential Integrity

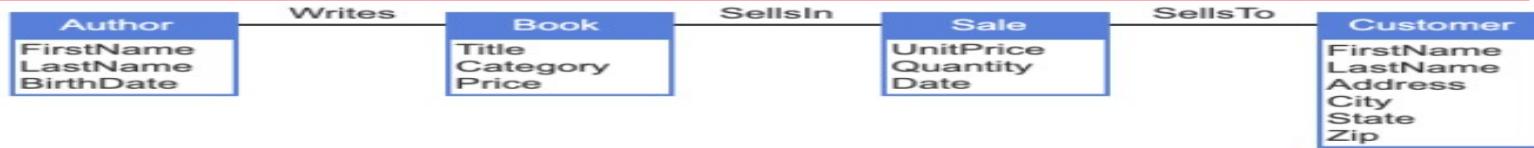
name	sid	course
Jones	53666	History105
Guldu	53832	Reggae203

Figure 3.18 An Instance of the B-Students View



Figure 2.3.1: ER diagram.

WHAT



## Logical design

HOW

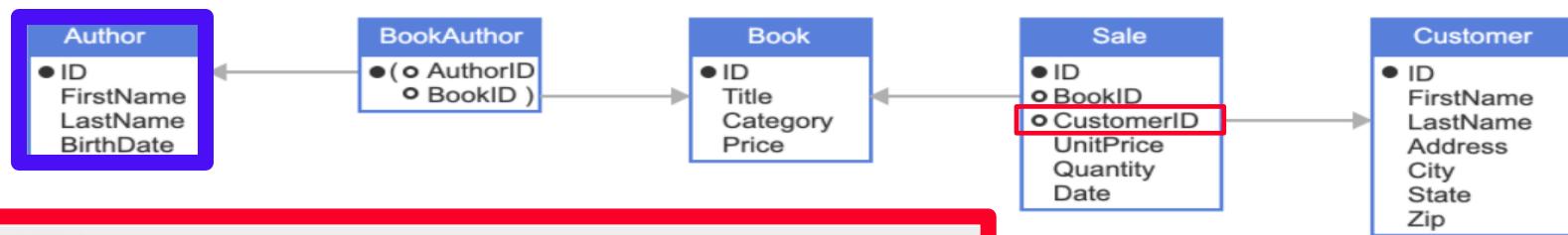
The **logical design** phase implements database requirements in a specific database system. For relational database systems, logical design converts entities, relationships, and attributes into tables, keys, and columns. A **key** is a column used to identify individual rows of a table. Tables, keys, and columns are specified in SQL with CREATE TABLE statements.

The logical design is depicted in a **table diagram**. Table diagrams are similar to ER diagrams but more detailed:

- Rectangles represent tables. Table names appear at the top of rectangles.
- Text within rectangles and below table names represents columns.
- Solid bullets (●) indicate key columns.
- Empty bullets (○) and arrows indicate columns that refer to keys.

The logical design, as specified in SQL and depicted in a table diagram, is called a database **schema**.

Figure 2.3.2: Table diagram.



PARTICIPATION ACTIVITY

2.3.2: Logical design.

1) CustomerID is:

- A table
- A key
- A column that refers to a key

Correct

Columns that refer to a key have an empty bullet prefix.

?????

# (CA 4.4.1 – Step 1 -Primary keys)

## CHALLENGE ACTIVITY

4.4.1: Primary and foreign keys.

• ISOCode3	CountryName	ContinentCode	IndependenceYear
BGR	Bulgaria	EU	1878
CAN	Canada	NA	NULL
PLW	Palau	OC	NULL
CZE	Czech Republic	EU	1918
JPN	Japan	AS	NULL

```
SELECT CountryName  
FROM Country  
WHERE ISOCode3 = 'CZE';
```

What is returned?

Bulgaria

Canada

Palau

Czech Republic

Japan

**CountryName**  
**Czech Republic**

1

2

3

4

**Check**

**Next**

✓ Expected: Czech Republic

'Czech Republic' is the CountryName for which the value of the primary key ISOCode3 is 'CZE'.

## 4. SET 1 - 3:DATA MODELING HOW - RELATIONAL MODELS



### 4.1 Relational model



### 4.2 Tables, columns, and rows



### 4.3 Null values



### 4.4 Primary and foreign keys



### 4.5 Referential integrity



## 4.3 Null values

### NULL

**NULL** is a special value that represents missing data. NULL represents either 'unknown' or 'inapplicable'. In the Compensation table below, the NULL in the BirthDate column means unknown, since all people have a birth date. Also, if engineering employees are never paid a bonus, the NULL in the Bonus column means inapplicable.

NULL is not the same as zero for numeric data types or blanks for character data types. Ex: A zero bonus indicates an employee can, but has not earned a bonus. Thus, bonus is known and applicable, and should not be represented as NULL.

Some columns should never contain a NULL. In the Compensation table, the ID column identifies employees and should always have a valid integer value. To prohibit NULL, a column can be designated NOT NULL in SQL. The database rejects any attempt to insert a row with missing data in a NOT NULL column.

PARTICIPATION ACTIVITY | 4.3.1: NULL values in the Compensation table.

Compensation						
NOT NULL		Name	BirthDate	Salary	Department	Bonus
ID		Lisa Ellison	October 2, 1993	45000	Engineering	NULL
2538		Sam Snead	NULL	32000	Sales	1000
5384		Maria Rodriguez	December 21, 2001	95000	Sales	3000
6381						

PARTICIPATION ACTIVITY | 4.3.2: NULL values.

Compensation				
ID	Name	Department	Salary	Bonus
2538	Lisa Ellison	Engineering	45000	0
5384	Sam Snead	Sales	30500	1000
6381	NULL	Sales	92300	3000

1) What does a NULL in the Name column represent?

- Unknown
- Inapplicable
- Either unknown or inapplicable

Correct

Every employee has a name, so NULL indicates the name exists but has not been entered. The name is unknown to the database.

?????

## 4. SET 1 - 3:DATA MODELING HOW - RELATIONAL MODELS



### 4.1 Relational model



### 4.2 Tables, columns, and rows



### 4.3 Null values



### 4.4 Primary and foreign keys



### 4.5 Referential integrity



## 4.4 Primary and foreign keys

### Primary keys

A **primary key** is a column, or group of columns, used to identify a row. In the Employee table below, the primary key is ID because each employee has a unique employee ID. The primary key usually appears on the table's left side, but the primary key's position is not significant to the database. In this material, a solid circle (●) precedes the primary key in table examples.

Table 4.4.1: Employee table with primary key ID.

Employee		
● ID	Name	Salary
2538	Lisa Ellison	45000
5384	Sam Snead	30500
6381	Maria Rodriguez	92300

### PARTICIPATION ACTIVITY

#### 4.4.1: Primary keys.

Refer to the Employee table.

Employee		
● ID	Name	Salary
2538	Lisa Ellison	45000
5384	Sam Snead	30500
6381	Maria Rodriguez	92300

- 1) Name is a good primary key.

- True  
 False

New employees with the same name might be added to the table. Since the Name column might contain duplicate values, the Name column is not a good primary key.

?????

## Foreign keys

A **foreign key** is a column, or group of columns, that refer to a primary key. Non-NULL foreign key values must match some value of the primary key. The data types of the foreign and primary keys must be the same, but the names may be different. In this material, an empty circle (○) precedes foreign keys in table examples.

The Manager column of the Department table below refers to the ID column of Employee, so Manager is a foreign key. Both columns have an integer data type, but ID and Manager are different names.

### PARTICIPATION ACTIVITY

4.4.4: The foreign key Manager refers to the primary key ID.

Department		Foreign key	Primary key Employee		
• Code	Name	○ Manager	• ID	Name	Salary
44	Engineering	2538	2538	Lisa Ellison	45000
82	Sales	6381	5384	Sam Snead	30500
12	Marketing	6381	6381	Maria Rodriguez	92300
99	Technical support	NULL			

### PARTICIPATION ACTIVITY

4.4.5: Foreign keys.

Refer to the Employee and Department tables below.

#### Department

• Code	Name	○ Manager
44	Engineering	2538
82	Sales	6381
12	Marketing	6381
99	Technical support	NULL

#### Employee

• ID	Name	Salary
2538	Lisa Ellison	45000
5384	Sam Snead	30500
6381	Maria Rodriguez	92300

- 1) The data type of Manager and ID must be the same.

- True
- False

#### Correct

Manager is a foreign key, which refers to the ID primary key. Data types of foreign and primary key must be the same.

?????

# (CA 4.4.1 – Step 3 -Primary and foreign keys)

CHALLENGE  
ACTIVITY

4.4.1: Primary and foreign keys.

SQL ??????

Country			
• ISOCode2	CountryName	Continent	Capital
ZW	Zimbabwe	Africa	Harare
TN	Tunisia	Africa	Tunis
MU	Mauritius	Africa	Port Louis

Geography			
○ Code	● TLD	Area	IndependenceYear
ZW	.zw	149363.62	1980
TN	.tn	59984.83135	1956
MU	.mu	783.7873818	1968

What is the Area of Tunisia?

3

Area  
**59984.83135**

?????

Check

Next

## 4. SET 1 - 3:DATA MODELING HOW - RELATIONAL MODELS



### 4.1 Relational model



### 4.2 Tables, columns, and rows



### 4.3 Null values



### 4.4 Primary and foreign keys



### 4.5 Referential integrity



## Referential integrity rules

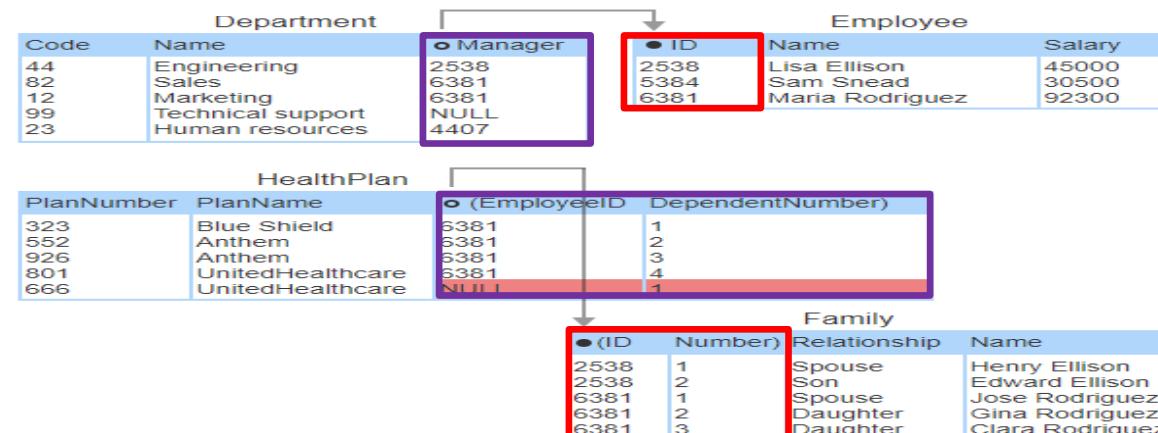
A **fully NULL** foreign key is a simple or composite foreign key in which all columns are NULL. Key values must either be fully NULL or match some primary key value.

**Referential integrity** requires that all foreign

In a relational database, foreign keys must obey referential integrity at all times. Occasionally, data entry errors or incomplete data result in referential integrity violations. Violations must be corrected before data is stored in the database.

### PARTICIPATION ACTIVITY

#### 4.5.1: Referential integrity rules.



### PARTICIPATION ACTIVITY

#### 4.5.2: Referential integrity rules for simple primary keys.

Refer to the Employee and Department tables below.

Department		Manager
Code	Name	ID
44	Engineering	2538
82	Sales	3829
12	Marketing	6381
99	Technical support	NULL

ID	Name	Salary
2538	Lisa Ellison	45000
5384	Sam Snead	30500
6381	Maria Rodriguez	92300

- 1) In the Department table, which foreign key value violates referential integrity?

- 2538
- 3829
- NULL

Since no employee has ID 3829, 3829 violates referential integrity.

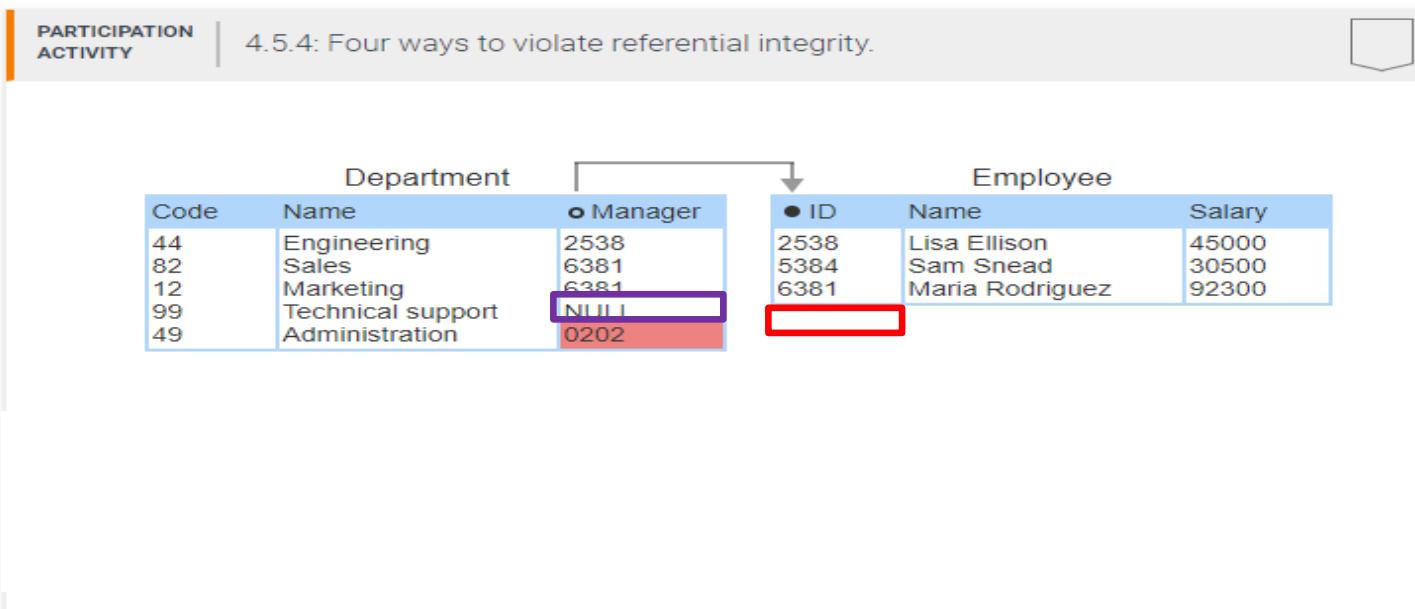
?????

## Referential integrity violations

Referential integrity can be violated in four ways:

1. A primary key is updated.
2. A foreign key is updated.
3. A row containing a primary key is deleted.
4. A row containing a foreign key is inserted.

Only these four operations can violate referential integrity. Primary key inserts and foreign key deletes cannot violate referential integrity.



PARTICIPATION ACTIVITY

4.5.5: Referential integrity violations.

Match the violation type to the database change.

Update a foreign key

Change the Technical Support department manager to 8001.

Changing the Technical Support department manager is a foreign key update. Since no employee ID is 8001, this update causes a violation.

?????

## Referential integrity actions

An insert, update, or delete that violates referential integrity can be corrected manually. A database user can correct an invalid foreign key with an update, or create a matching primary key with an insert. However, manual corrections are time-consuming and error-prone.

Databases can automatically correct referential integrity violations with any of four actions, which are specified in SQL when creating a table with a foreign key:

- **RESTRICT** rejects an insert, update, or delete that violates referential integrity.
- **SET NULL** sets invalid foreign keys to NULL.
- **SET DEFAULT** sets invalid foreign keys to a default primary key value, specified in SQL.
- **CASCADE** propagates primary key changes to foreign keys.

CASCADE behaves differently for primary key updates and deletes. If a primary key is deleted, rows containing matching foreign keys are deleted. If a primary key is updated, matching foreign keys are updated to the same value.

### PARTICIPATION ACTIVITY

#### 4.5.6: Enforcing referential integrity with RESTRICT.

#### RESTRICT

Department		Manager	Employee	
Code	Name	ID	Name	Salary
44	Engineering	2538	Lisa Ellison	45000
82	Sales	6381	Sam Snead	30500
12	Marketing	6381	Maria Rodriguez	92300
99	Technical support	NULL		
49	Administration	0202		

### PARTICIPATION ACTIVITY

#### 4.5.9: Referential integrity actions.

Refer to the Employee and Department tables. What are the results of the actions below?

##### Department

Code	Name	Manager
44	Engineering	2538
82	Sales	6381
12	Marketing	6381
99	Technical support	NULL

##### Employee

ID	Name	Salary
2538	Lisa Ellison	45000
5384	Sam Snead	30500
6381	Maria Rodriguez	92300

3. A row containing a primary key is deleted.

1) RESTRICT, when the row containing Maria Rodriguez is deleted.

- The Sales and Marketing managers are set to NULL.
- The Sales and Marketing departments are deleted.
- The delete is rejected.

##### Correct

RESTRICT rejects referential integrity violations. If Maria Rodriguez is deleted, the managers of Sales and Marketing would not match any ID, so the delete is rejected.

?????

## Referential integrity actions

An insert, update, or delete that violates referential integrity can be corrected manually. A database user can correct an invalid foreign key with an update, or create a matching primary key with an insert. However, manual corrections are time-consuming and error-prone.

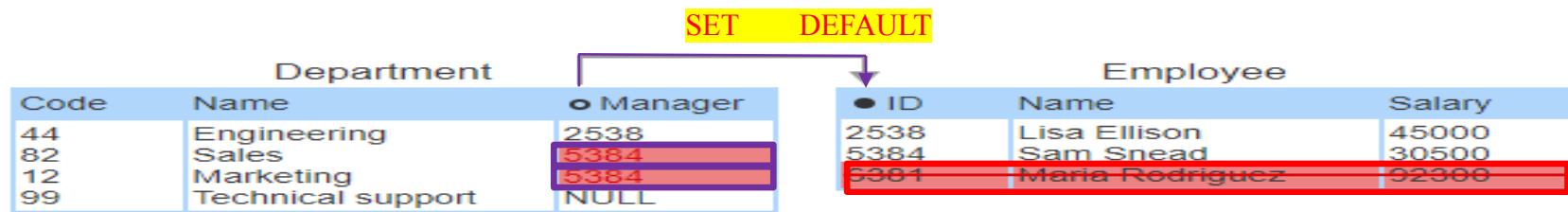
Databases can automatically correct referential integrity violations with any of four actions, which are specified in SQL when creating a table with a foreign key:

- **RESTRICT** rejects an insert, update, or delete that violates referential integrity.
- **SET NULL** sets invalid foreign keys to NULL.
- **SET DEFAULT** sets invalid foreign keys to a default primary key value, specified in SQL.
- **CASCADE** propagates primary key changes to foreign keys.

CASCADE behaves differently for primary key updates and deletes. If a primary key is deleted, rows containing matching foreign keys are deleted. If a primary key is updated, matching foreign keys are updated to the same value.

### PARTICIPATION ACTIVITY

#### 4.5.7: Enforcing referential integrity with SET NULL and SET DEFAULT.



### PARTICIPATION ACTIVITY

#### 4.5.9: Referential integrity actions.

Refer to the Employee and Department tables. What are the results of the actions below?

Code	Name	Manager
44	Engineering	2538
82	Sales	6381
12	Marketing	6381
99	Technical support	NULL

ID	Name	Salary
2538	Lisa Ellison	45000
5384	Sam Snead	30500
6381	Maria Rodriguez	92300

3. A row containing a primary key is deleted.

3) SET DEFAULT, when Lisa Ellison's ID is changed to 1001.

- The Engineering manager is set to NULL.
- The Engineering manager is set to a default value, such as 9999.
- The change is rejected.

Correct

SET DEFAULT sets matching foreign keys to a default value. Since Lisa Ellison manages Engineering, the Engineering manager is set to a default value, which must be a valid primary key in the Employee table.

?????

## Referential integrity actions

An insert, update, or delete that violates referential integrity can be corrected manually. A database user can correct an invalid foreign key with an update, or create a matching primary key with an insert. However, manual corrections are time-consuming and error-prone.

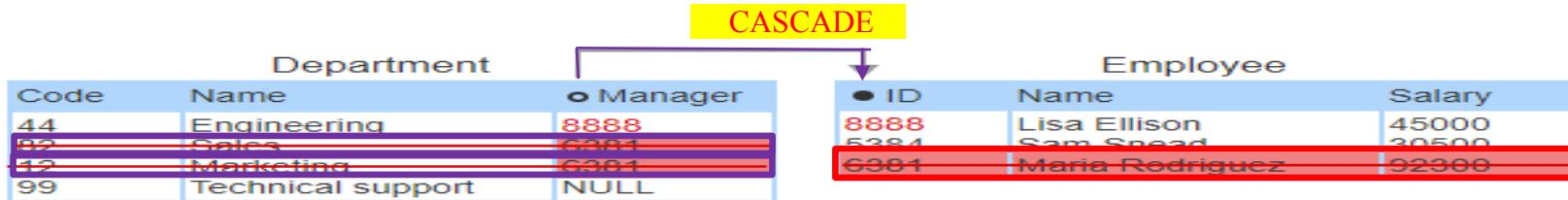
Databases can automatically correct referential integrity violations with any of four actions, which are specified in SQL when creating a table with a foreign key:

- **RESTRICT** rejects an insert, update, or delete that violates referential integrity.
- **SET NULL** sets invalid foreign keys to NULL.
- **SET DEFAULT** sets invalid foreign keys to a default primary key value, specified in SQL.
- **CASCADE** propagates primary key changes to foreign keys.

CASCADE behaves differently for primary key updates and deletes. If a primary key is deleted, rows containing matching foreign keys are deleted. If a primary key is updated, matching foreign keys are updated to the same value.

### PARTICIPATION ACTIVITY

#### 4.5.8: Enforcing referential integrity with CASCADE.



### PARTICIPATION ACTIVITY

#### 4.5.9: Referential integrity actions.

Refer to the Employee and Department tables. What are the results of the actions below?

Department		
Code	Name	Manager
44	Engineering	2538
82	Sales	6381
12	Marketing	6381
99	Technical support	NULL

Employee		
ID	Name	Salary
2538	Lisa Ellison	45000
5384	Sam Sneed	30500
6381	Maria Rodriguez	92300

3. A row containing a primary key is deleted.

5) CASCADE, when Maria Rodriguez is deleted.

- The Sales and Marketing managers are set to NULL.
- The Sales and Marketing departments are deleted.
- The delete is rejected.

CASCADE propagates primary key deletes to matching foreign keys. Maria Rodriguez manages Sales and Marketing, so the Sales and Marketing rows are deleted.

?????

# TA time (Fernando) – ~4 minutes

## (CA 4.5.1 Referential Integrity – Step 1 – RESTRICT)

CHALLENGE  
ACTIVITY

4.5.1: Referential integrity.

Ask Students

Country

• ISOCode2	Name	IndepDate	PopDensity
SY	Syria	1946-04-17	238.4527145
BF	Burkina Faso	1958-12-11	186.9745644
VN	Vietnam	1945-09-02	798.0407234

Geography

○ ID	• ISOCode3	15to64PopPct	Continent
SY	SYR	0.591	Asia
BF	BFA	0.524	Africa
VN	VNM	0.698	Asia

With RESTRICT referential integrity, what happens if the ISOCode2 of Syria is updated to 'XX'?

Select

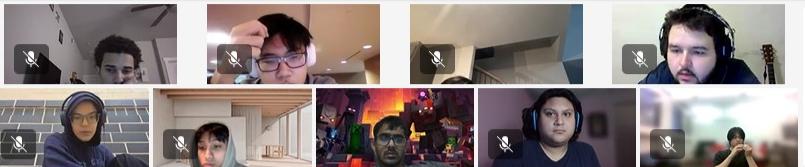
1

Check

Next



01:13:18

Take control
Pop out
Chat
117
People
Raise
View
Rooms
Apps
More
Camera
Mic
Share
Leave


Change your view



FR

Ramirez, Ferna...



View all

Apple
Arc
File
Edit
View
Spaces
Archive
Extensions
Window
Help
544874\_3144414.qx3zq7/
Wed Jan 24 4:58:54 PM

Jump to level 1

Country

• ISOCode2	Name	Capital	IndepYear
SS	South Sudan	Juba	2011
SB	Solomon Islands	Honiara	1978
ES	Spain	Madrid	NULL

Geography

• ISOCode3	ID	Population	Over65PopPct
SSD	SS	8260490	0.034
SLB	SB	652858	0.035
ESP	ES	46796540	0.194

With RESTRICT referential integrity, what happens if the row containing South Sudan is deleted from the Country table?

Delete is rejected

1    2    3    4    5

✓ Expected: Delete is rejected

RESTRICT rejects referential integrity violations.

If the row containing South Sudan is deleted from the Country table, then 'SS' in the Geography table would not match any ISOCode2 in the Country table. Thus, the delete is rejected.

[View solution](#) (Instructors only)

- 1
- 2
- 3
- 4
- 5

Feedback?

Ramirez, Fernando



63°F Light rain

4:58 PM  
1/24/2024

ta

# TA time (Fernando) – ~4 minutes

## (CA 4.5.1 Referential Integrity – Step 2 – SET NULL)

CHALLENGE  
ACTIVITY

4.5.1: Referential integrity.

Ask Students

Country			
• ISOCode3	Name	IndepDate	Area
MUS	Mauritius	1968-03-12	783.7873818
BLR	Belarus	1944-07-03	78374.10797
VUT	Vanuatu	1980-07-30	4706.585313

Geography			
◦ ID	• ISOCode2	Population	Over65PopPct
MUS	MU	1265303	0.109
BLR	BY	9483499	0.148
VUT	VU	292680	0.044

With SET NULL referential integrity, what happens if the ISOCode3 of Mauritius is updated to 'XX'?

What happens to the row with ID MUS in the Geogr

2

Check

Next



01:15:31

In this meeting (118) Mute all

- DP Palacios, David F
- AP Patel, Ayush
- AP Pham, An T
- QP Pham, Quoc Hung
- NP Phung, Nam K
- ZR Rahman, Zayed
- ZR Ramadan, Zeyad
- FR Ramirez, Fernando
- MR Roberts, Mojool T
- CR Robles, Cristian J
- AR Rodrigues, Albamaria V
- MR Rodriguez, Miguel A
- GR Romero Ramirez, Gabriela

Ramirez, Fernando



Type here to search



63°F Light rain

5:01 PM  
1/24/2024

# TA time (Fernando) – ~4 minutes

## (CA 4.5.1 Referential Integrity – Step 3 – SET DEFAULT)

CHALLENGE  
ACTIVITY

4.5.1: Referential integrity.

Ask Students

Country			
• TLD	Name	IndepYear	Area
.so	Somalia	1960	242217.3281
.vn	Vietnam	1945	119718.6963
.bt	Bhutan	NULL	14727.48036

Geography			
○ ID	• ISOCode2	Population	15to64PopPct
.so	SO	15008154	0.508
.vn	VN	95540395	0.698
.bt	BT	754394	0.686

With SET DEFAULT referential integrity, what happens if the TLD of Somalia is updated to 'XX'?

What happens to the row with ID .so in the Geography table?

3

Check

Next

01:16:46

Take control Pop out Chat People Raise View Rooms Apps More Camera Mic Share Leave

In this meeting (118) Mute all

Palacios, David F Patel, Ayush Pham, An T Pham, Quoc Hung Phung, Nam K Rahman, Zayed Ramadan, Zeyad Ramirez, Fernando Roberts, Mojoolu T Robles, Cristian J Rodrigues, Albamaria V Rodriguez, Miguel A Romero Ramirez, Gabriela

**CHALLENGE ACTIVITY** 4.5.1: Referential integrity.

Jump to level 1

Country

ISOCode2	Name	PopDensity	Area
BB	Barbados	1726.504144	166.0239292
QA	Qatar	620.5435277	4482.646061
PK	Pakistan	712.9960623	297638.432

Geography

Code	ID	15to64PopPct	ContinentCode
52	BB	0.66	NA
634	QA	0.848	AS
586	PK	0.607	AS

With **SET DEFAULT** referential integrity, what happens if the ISOCode2 of Barbados is updated to 'XX'?

ISOCode2 of Barbados is updated to 'XX'.

What happens to the row with ID BB in the Geography table?

'BB' in the Geography table is set to a specified default value.

Check Next

✓ Expected: ISOCode2 of Barbados is updated to 'XX'. 'BB' in the Geography table is set to a specified default value.

SET DEFAULT sets invalid foreign keys to a default primary key value specified in SQL.  
If the ISOCode2 of Barbados is updated to 'XX', then 'BB' in the Geography table would not match any ISOCode2 in the Country table. Thus, the foreign key 'BB' in the Geography table is changed to the default value.

Ramirez, Fernando

Type here to search

Cloud 63°F Light rain 5:02 PM 1/24/2024

# TA time (Fernando) – ~4 minutes

## (CA 4.5.1 Referential Integrity – Step 4 – CASCADE)

CHALLENGE  
ACTIVITY

4.5.1: Referential integrity.



Ask Students

- 1
- 2
- 3
- 4

Country			
• ISOCode3	Name	Area	Capital
CPV	Cape Verde	1555.991699	Praia
IRQ	Iraq	167617.7609	Baghdad
SLE	Sierra Leone	27868.8538	Freetown

Geography			
○ ID	• Code	Over65PopPct	15to64PopPct
CPV	132	0.045	0.653
IRQ	368	0.032	0.564
SLE	694	0.025	0.554

With CASCADE referential integrity, what happens if the ISOCode3 of Cape Verde is updated to 'XX'?

Select



What happens to the row with ID CPV in the Geography table?

Select



4

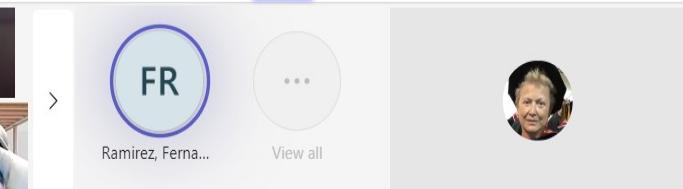
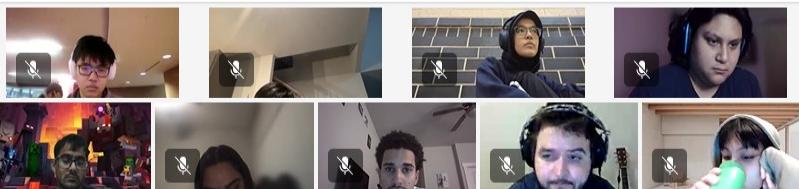
Check

Next



01:18:14

Take control Pop out Chat People Raise View Rooms Apps More Camera Mic Share Leave



< In this meeting (118) Mute all

- DP Palacios, David F
- AP Patel, Ayush
- AP Pham, An T
- QP Pham, Quoc Hung
- NP Phung, Nam K
- ZR Rahman, Zayed
- ZR Ramadan, Zeyad
- FR Ramirez, Fernando
- MR Roberts, Mojoolu T
- CR Robles, Cristian J
- AR Rodrigues, Albamaria V
- MR Rodriguez, Miguel A
- GR Romero Ramirez, Gabriela

Apple Arc File Edit View Spaces Tabs Archive Extensions Window Help

CHALLENGE

ACTIVITY 4.5.1: Referential integrity.

544874.314414.qx3zq7

Jump to level 1

## Country

ISOCode2	Name	IndepDate	PopDensity
EE	Estonia	1918-02-24	78.76477369
MU	Mauritius	1968-03-12	1614.344693
IN	India	1947-08-15	1178.284199

## Geography

Code	ID	15to64PopPct	Population
233	EE	0.642	1321977
480	MU	0.707	1265303
356	IN	0.662	1352617328

- 1
- 2
- 3
- 4
- 5

With CASCADE referential integrity, what happens if the ISOCode2 of Estonia is updated to 'XX'?

ISOCode2 of Estonia is updated to 'XX'

What happens to the row with ID EE in the Geography table?

'EE' in the Geography table is updated to 'XX'

1 2 3 4 5

Check

Next

✓ Expected: ISOCode2 of Estonia is updated to 'XX', 'EE' in the Geography table is updated to 'XX'

CASCADE propagates primary key changes to foreign keys.

If the ISOCode2 of Estonia is updated to 'XX', then 'EE' in the Geography table would not match any ISOCode2 in the Country table. Thus, the foreign key 'EE' in the Geography table is changed to 'XX'.

Ramirez, Fernando

-

+



63°F Light rain

5:03 PM  
1/24/2024

At 5:05 PM .

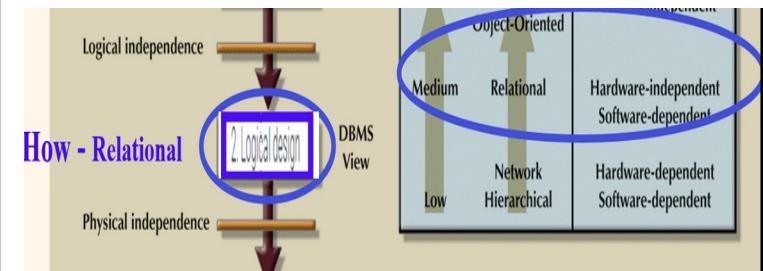
01.24.2024

ZyBook SET 1 - 3

Set 1

(3 - We)

LECTURE 3 DATA MODELING - HOW - RELATIONAL MODEL



HOW

1. SET 1

Empty

4. SET 1 - 3:DATA MODELING HOW - RELATIONAL MODELS

0% 0%

Please work on  
SET 1 – 4: DATA MODELING - HOW - RELATIONAL MODEL

# SET 1 Lecture 3

## Data Modeling HOW – RELATIONAL MODEL

### 4. SET 1 - 3:DATA MODELING HOW - RELATIONAL MO...



6%



0%



#### 4.1 Relational model Hidden



0%



0%



#### 4.2 Tables, columns, and rows Hidden



0%



#### 4.3 Null values Hidden



0%



0%



#### 4.4 Primary and foreign keys Hidden



25%



0%



#### 4.5 Referential integrity Hidden



0%



0%



# Next

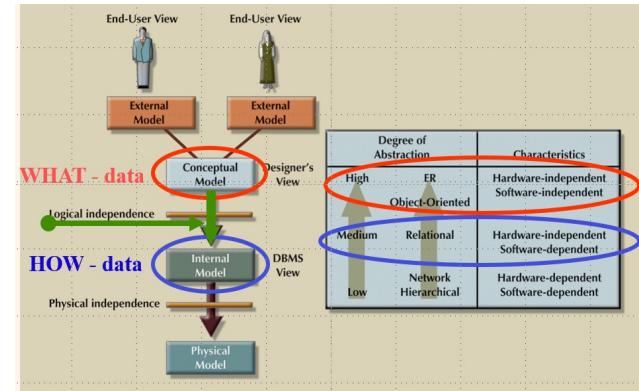
01.29.2024

ZyBook SET 1 - 4

(4 - Mo)

Set 1

LECTURE 4 ERD to RELATIONAL



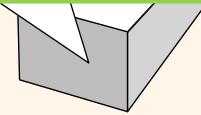
1. SET 1

Empty ▾

5. SET 1 - 4:ERD to RELATIONAL

VH, publish

**From 5:05 to 5:15 PM – 10 minutes.**



01.24.2024  ZyBook SET 1 - 3 

(3 - We) Set 1

LECTURE 3 DATA MODELING - HOW - RELATIONAL MODEL

 CLASS PARTICIPATION 20 points  20% of Total + :

## CLASS 3



### Class 3 END PARTICIPATION

Not available until Jan 24 at 5:05pm | Due Jan 24 at 5:15pm | 100 pts

VH, publish  :

This is an synchronous online class.

Attendance is required.

Recording or distribution of class materials is prohibited.

1. At the beginning of selected classes there is an assessment in the first 10 minutes. (beige BOX in the Detailed Syllabus)
2. At the end of selected classes there is an assessment in the last 10 minutes. (blue BOX in the Detailed Syllabus)
3. ZyBook sections will be downloaded and used for 30% of Total Score on the dates specified in the Detailed Syllabus.
4. EXAMS are in CANVAS. No late EXAMS.

At 5:15 PM.

**End Class 3**

**VH, unhide ZyBook Section 5.**



**VH, Download Attendance Report  
Rename it:  
1.24.2024 Attendance Report FINAL**

**VH, upload Class 3 to CANVAS.**