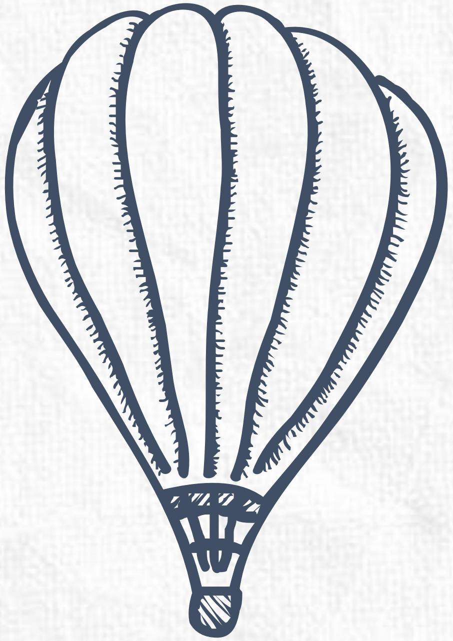
CSC3170 Tutorial 2

Qi, Xixian (120090691)

2024/01/23

CONTENT



1 Relational Schema

2 Keys

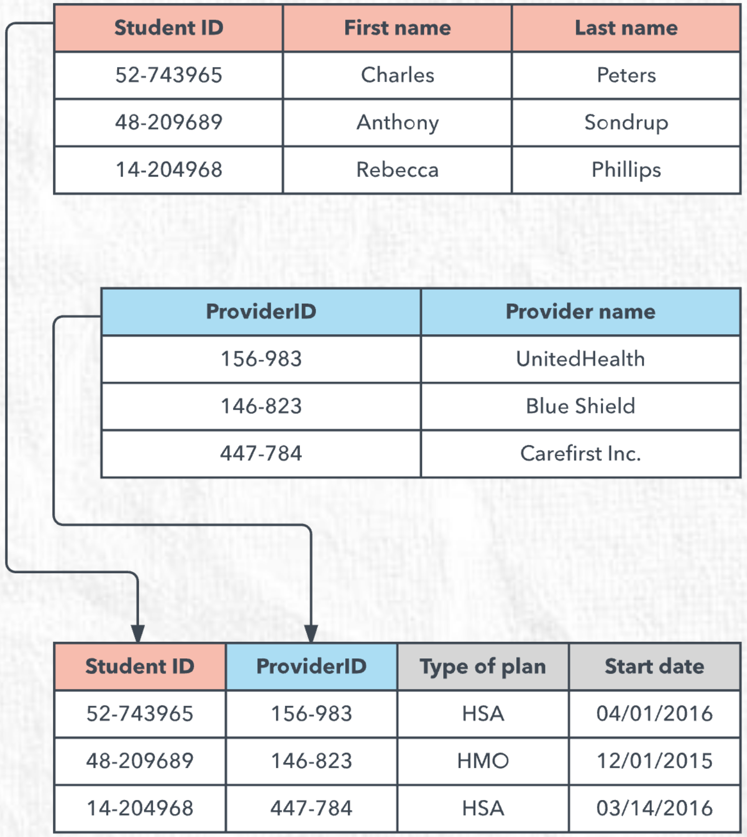
3 Relational Algebra

Relational Schema

01 Database model

A database model shows the logical structure of a database, including the relationships and constraints that determine how data can be stored and accessed. There are many kinds of data models, including:

- Hierarchical database model
- Relational model
- Network model
- Object-oriented model
- Entity-relationship model
- Document model
- Entity-attribute-value model
- Star schema
- Object-relational model



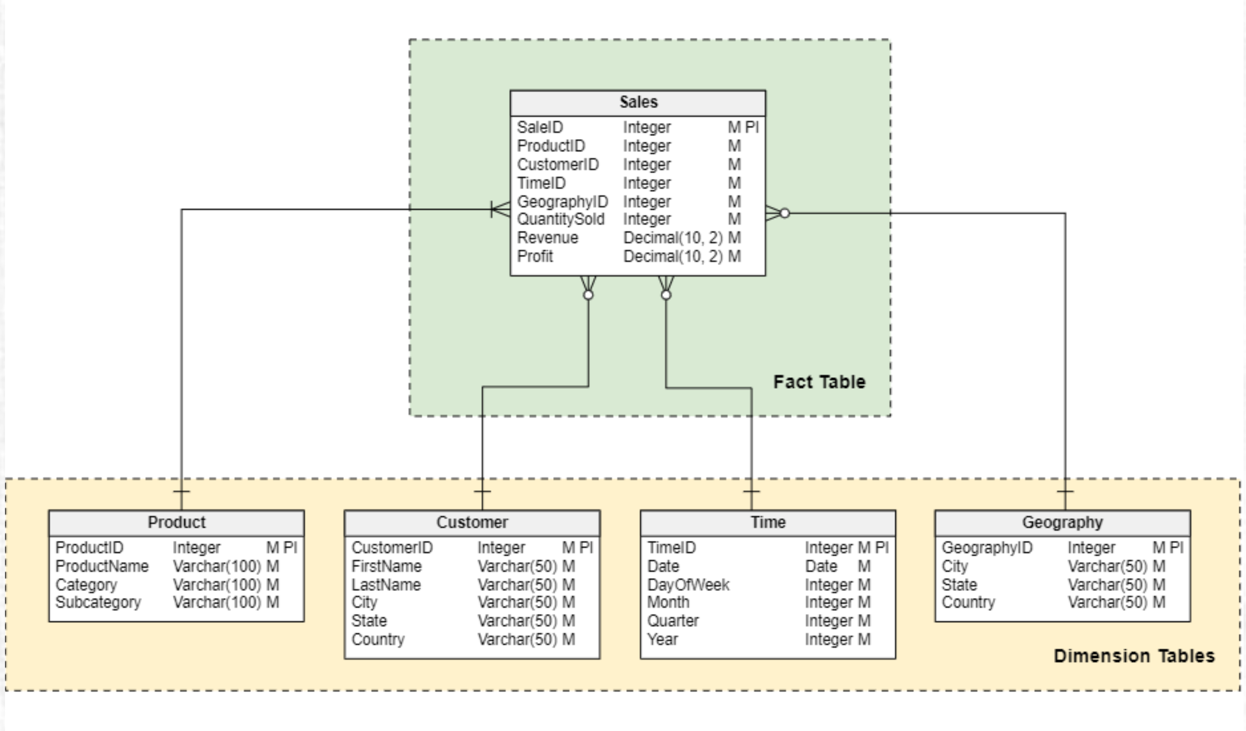
Student ID	First name	Last name
52-743965	Charles	Peters
48-209689	Anthony	Sondrup
14-204968	Rebecca	Phillips

ProviderID	Provider name
156-983	UnitedHealth
146-823	Blue Shield
447-784	Carefirst Inc.

Student ID	ProviderID	Type of plan	Start date
52-743965	156-983	HSA	04/01/2016
48-209689	146-823	HMO	12/01/2015
14-204968	447-784	HSA	03/14/2016

Fig: Example of relational model

01 Database model



Fact Table

Sales

SaleID	Integer	M PI
ProductID	Integer	M
CustomerID	Integer	M
TimeID	Integer	M
GeographyID	Integer	M
QuantitySold	Integer	M
Revenue	Decimal(10, 2)	M
Profit	Decimal(10, 2)	M

Dimension Tables

Product

ProductID	Integer	M PI
ProductName	Varchar(100)	M
Category	Varchar(100)	M
Subcategory	Varchar(100)	M

Customer

CustomerID	Integer	M PI
FirstName	Varchar(50)	M
LastName	Varchar(50)	M
City	Varchar(50)	M
State	Varchar(50)	M
Country	Varchar(50)	M

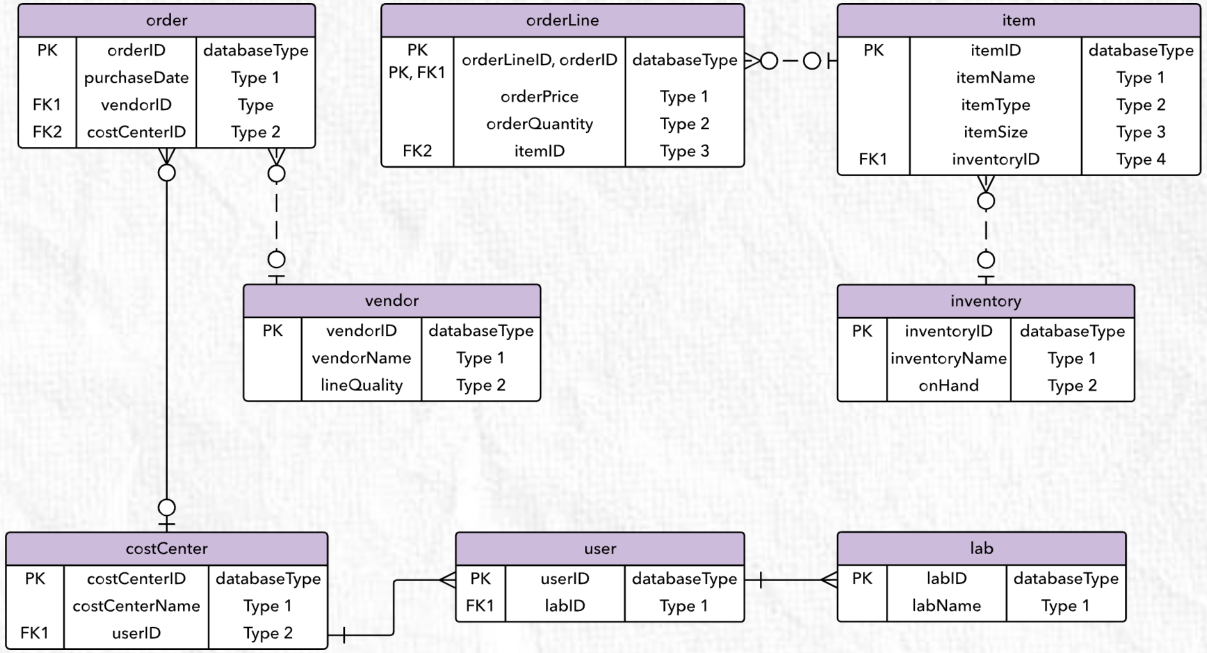
Time

TimeID	Integer	M PI
Date	Date	M
DayOfWeek	Integer	M
Month	Integer	M
Quarter	Integer	M
Year	Integer	M

Geography

GeographyID	Integer	M PI
City	Varchar(50)	M
State	Varchar(50)	M
Country	Varchar(50)	M

Fig: Example of STAR schema



order

PK	orderID	databaseType
	purchaseDate	Type 1
FK1	vendorID	Type
FK2	costCenterID	Type 2

orderLine

PK PK, FK1	orderLineID, orderID	databaseType
	orderPrice	Type 1
	orderQuantity	Type 2
FK2	itemID	Type 3

item

PK	itemID	databaseType
	itemName	Type 1
	itemType	Type 2
	itemSize	Type 3
FK1	inventoryID	Type 4

vendor

PK	vendorID	databaseType
	vendorName	Type 1
	lineQuality	Type 2

inventory

PK	inventoryID	databaseType
	inventoryName	Type 1
	onHand	Type 2

costCenter

PK	costCenterID	databaseType
	costCenterName	Type 1
FK1	userID	Type 2

user

PK	userID	databaseType
FK1	labID	Type 1

lab

PK	labID	databaseType
	labName	Type 1

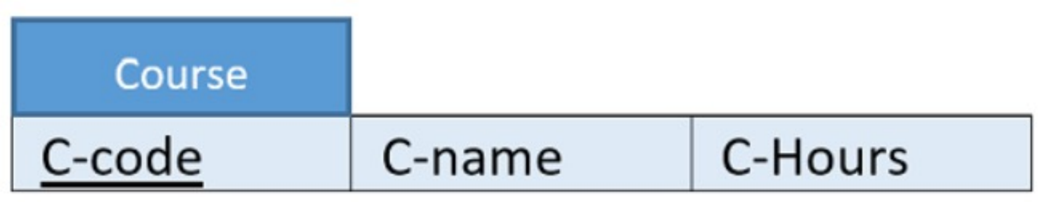
Fig: Example of Entity-relationship model

01 Relational model

The relational model was developed in 1970 by EF Codd as a way to model data in a table format, rather than as a diagram. Instead of focusing on the relationships and instances between entities, tables in the relational model show relevant data, how each table is related.

In other words, each ‘table’ stands for a relation schema in the relational model.

Example:
“*Each course* has a unique **ID**, **Name** and **Accumulated Hours**”.
relation attributes



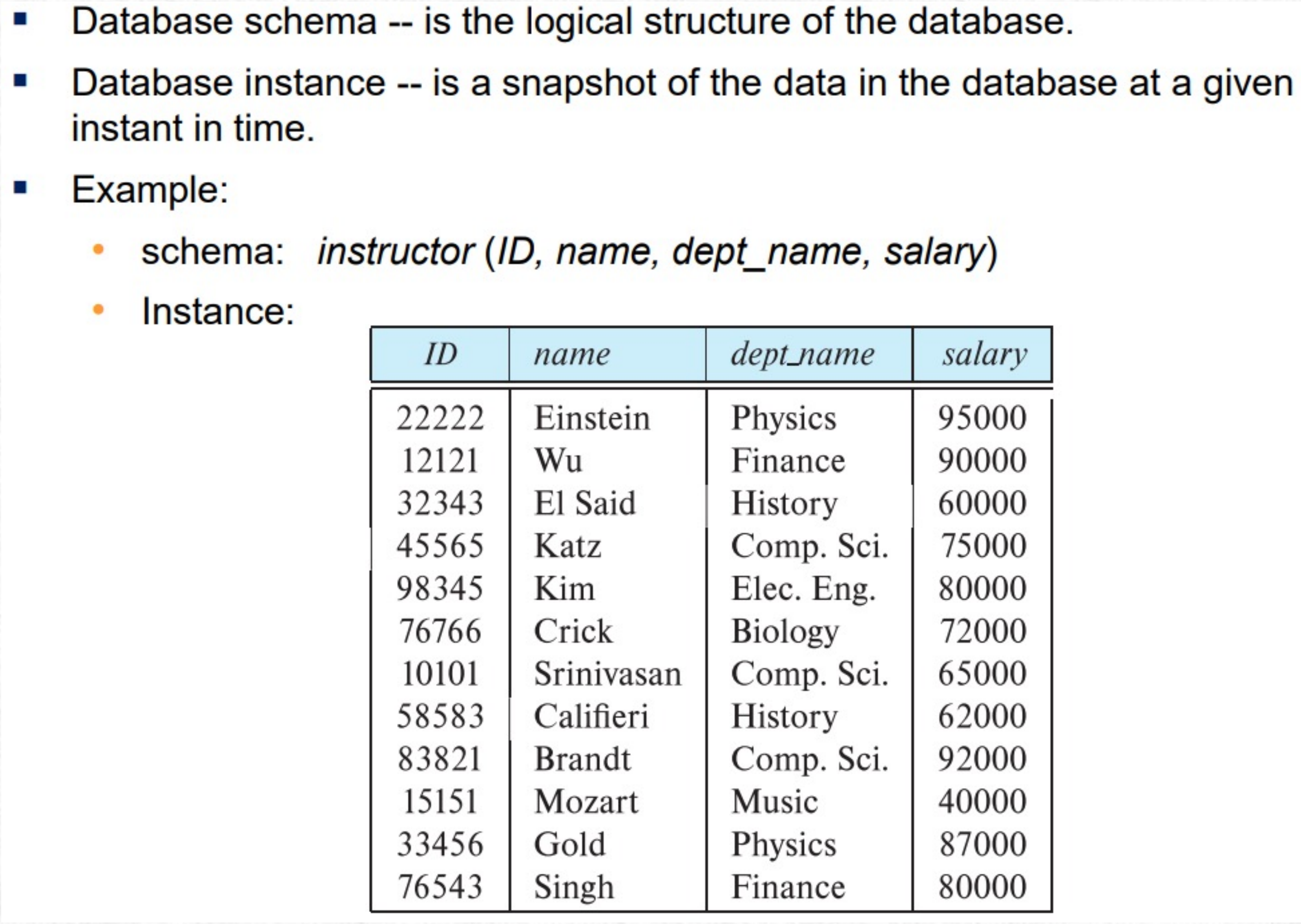
Course		
C-code	C-name	C-Hours

schema: Course (ID, name, hours)

Schema & Instance

~class

~instance

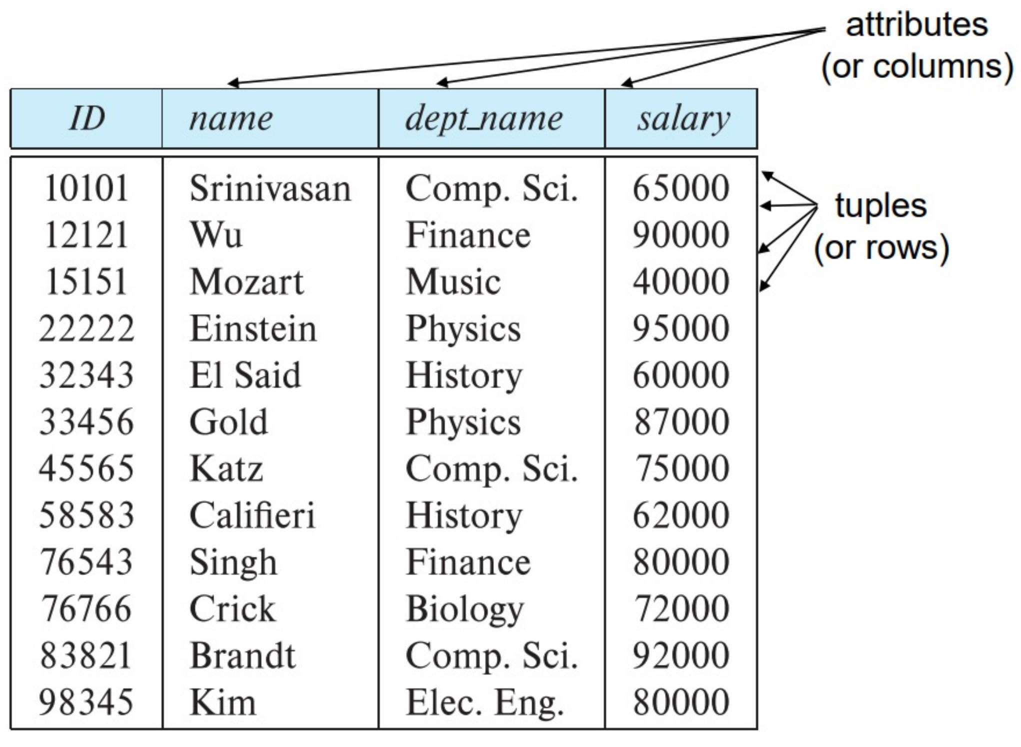


- Database schema -- is the logical structure of the database.
- Database instance -- is a snapshot of the data in the database at a given instant in time.
- Example:
  - schema: *instructor* (*ID, name, dept_name, salary*)
  - Instance:

ID	*name*	*dept_name*	*salary*
22222	Einstein	Physics	95000
12121	Wu	Finance	90000
32343	El Said	History	60000
45565	Katz	Comp. Sci.	75000
98345	Kim	Elec. Eng.	80000
76766	Crick	Biology	72000
10101	Srinivasan	Comp. Sci.	65000
58583	Califieri	History	62000
83821	Brandt	Comp. Sci.	92000
15151	Mozart	Music	40000
33456	Gold	Physics	87000
76543	Singh	Finance	80000

01 Important terminologies

- **Attributes**: properties that define an entity (col).
- **Domain:** The set of allowable values for each attribute.
- **Tuple:** a single record in the database (row).



ID	*name*	*dept_name*	*salary*
10101	Srinivasan	Comp. Sci.	65000
12121	Wu	Finance	90000
15151	Mozart	Music	40000
22222	Einstein	Physics	95000
32343	El Said	History	60000
33456	Gold	Physics	87000
45565	Katz	Comp. Sci.	75000
58583	Califieri	History	62000
76543	Singh	Finance	80000
76766	Crick	Biology	72000
83821	Brandt	Comp. Sci.	92000
98345	Kim	Elec. Eng.	80000

attributes
(or columns)
tuples
(or rows)

Keys

02 Introduction to keys

Q: What are 'keys'?
A: A key is an attribute or a group of attributes in a table.

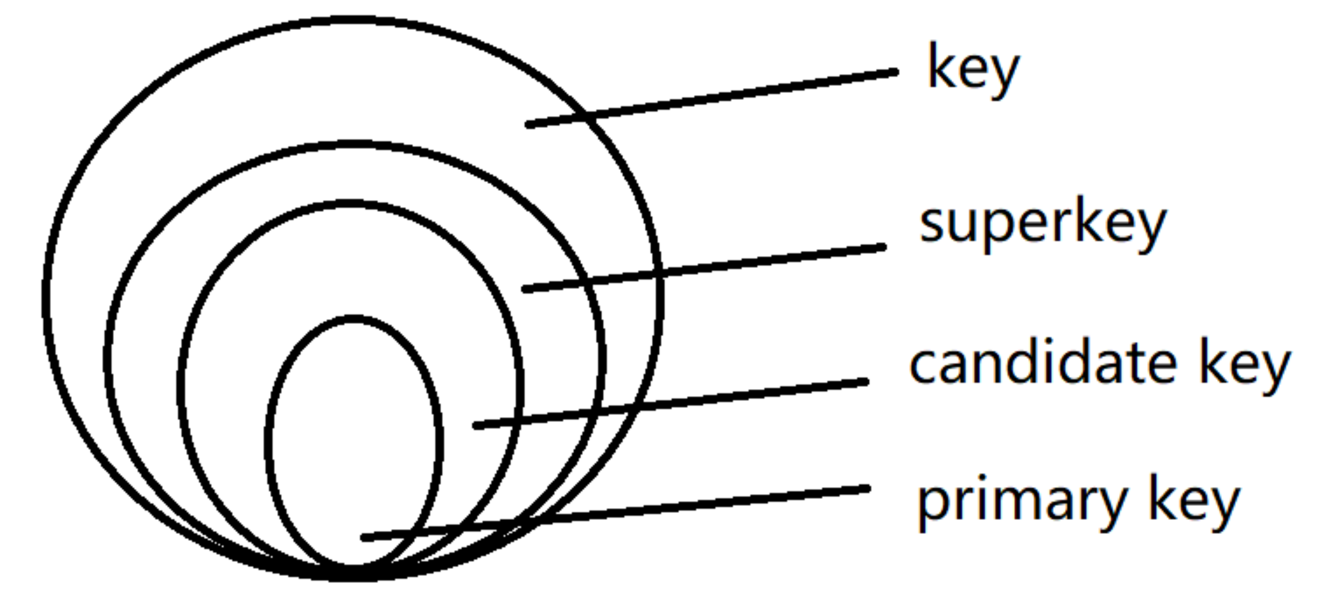
Q: Why do we implement 'keys'?
A1: Uniqueness. We have mentioned that each record is unique in the schema, which formalizes the notion of super key, primary key, and candidate key.
A2: Association. A simple way to associate with different schema is to copy one's primary key into the other's attribute, which is an image of the linked relation's primary key.

02 Super, candidate and primary keys

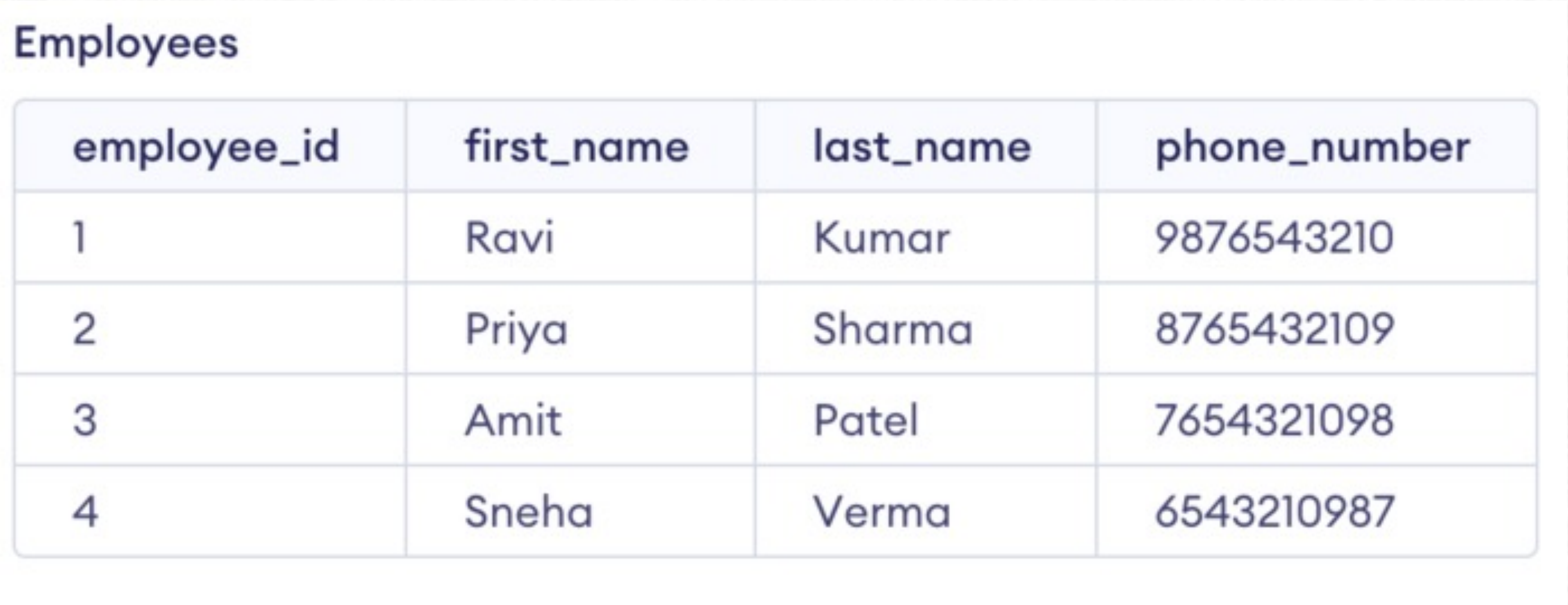
- A super key is any set of attributes whose values, taken together, uniquely identify each row of a table.

- A candidate key is a **minimal** super key.

- A primary key is the **specific** candidate key that we picked to serve as the unique identifier for rows of this table.



02 Practice 1



Employees

employee_id	first_name	last_name	phone_number
1	Ravi	Kumar	9876543210
2	Priya	Sharma	8765432109
3	Amit	Patel	7654321098
4	Sneha	Verma	6543210987

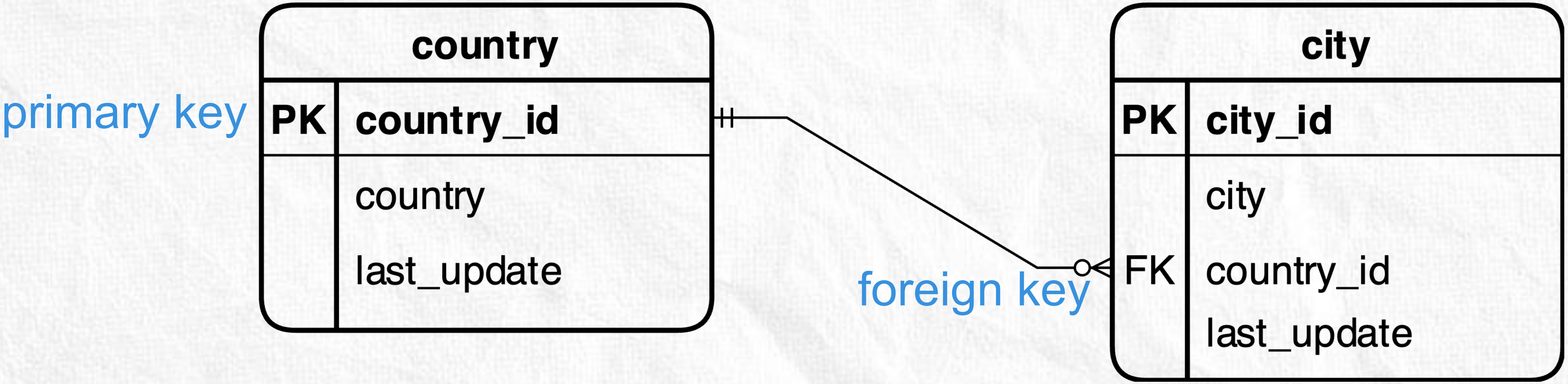
Q1: What are super keys?

Q2: What are candidate keys?

Q3: How many primary keys are possible?

02 Foreign keys

- A foreign key is a column or columns in a table that that are linked to a primary key in a different table.



primary key

	country
PK	**country_id**
	country
	last_update

foreign key

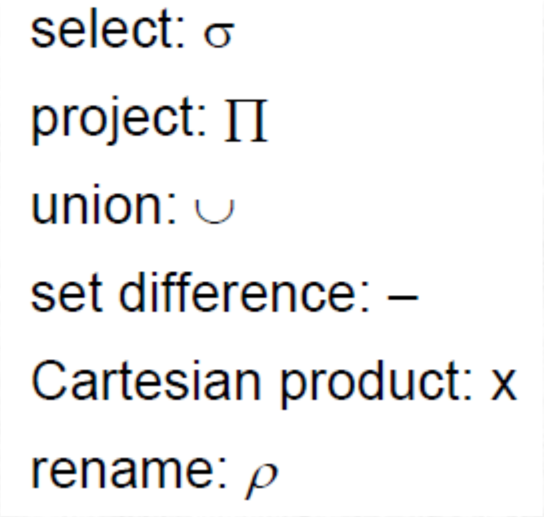
	city
PK	**city_id**
	city
FK	country_id
	last_update

Relational Algebra

03 Relational Algebra

Relational algebra is a procedural query language, which takes instances of relations as input and yields instances of relations as output.

There are 6 basic operators:



select: σ

project: Π

union: $\cup$

set difference: –

Cartesian product: x

rename: ρ

03 Practice 2

Consider a database with the following schema:

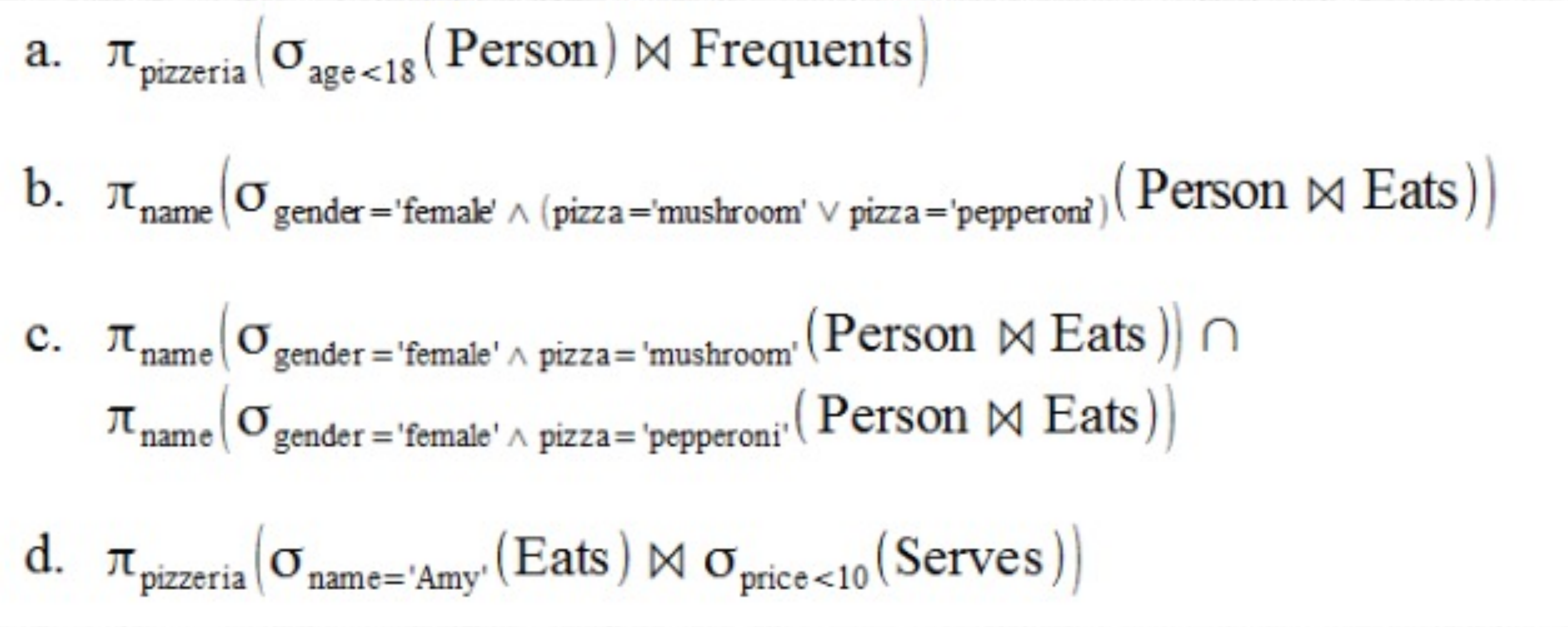
Person (name, age, gender)
Frequents (name, pizzeria) [Note: *visit a lot*]
Eats (name, pizza)
Serves (pizzeria, pizza, price)

Write relational algebra expressions for the following 4 queries:

a) Find all pizzerias frequented by at least one person under the age of 18.
b) Find the names of all females who eat either mushroom or pepperoni pizza (or both).
c) Find the names of all females who eat both mushroom and pepperoni pizza.
d) Find all pizzerias that serve at least one pizza that Amy eats for less than $10.00.

03

Practice 2 keys



a. $\pi_{\text{pizzeria}}\left(\sigma_{\text{age}<18}(\text{Person}) \bowtie \text{Frequents}\right)$

b. $\pi_{\text{name}}\left(\sigma_{\text{gender}=\text{'female'} \wedge (\text{pizza}=\text{'mushroom'} \vee \text{pizza}=\text{'pepperoni'})}(\text{Person} \bowtie \text{Eats})\right)$

c. $\pi_{\text{name}}\left(\sigma_{\text{gender}=\text{'female'} \wedge \text{pizza}=\text{'mushroom'}}(\text{Person} \bowtie \text{Eats})\right) \cap$
$\pi_{\text{name}}\left(\sigma_{\text{gender}=\text{'female'} \wedge \text{pizza}=\text{'pepperoni'}}(\text{Person} \bowtie \text{Eats})\right)$

d. $\pi_{\text{pizzeria}}\left(\sigma_{\text{name}=\text{'Amy'}}(\text{Eats}) \bowtie \sigma_{\text{price}<10}(\text{Serves})\right)$

For more practice questions, you may visit: Relational Algebra Exercises

Thank you!