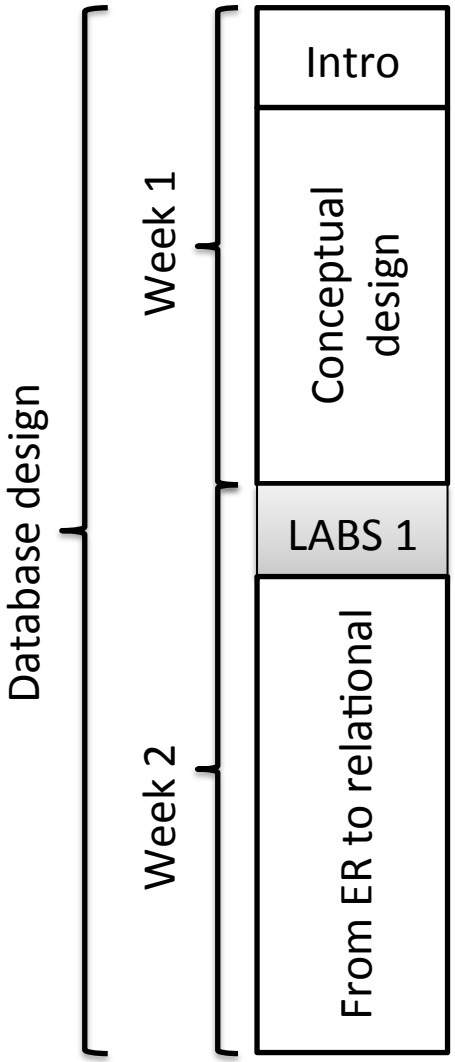
UPPSALA UNIVERSITET

The Relational Model

Lecturer: Neena Thota
neena.thota@it.uu.se

UPPSALA UNIVERSITET

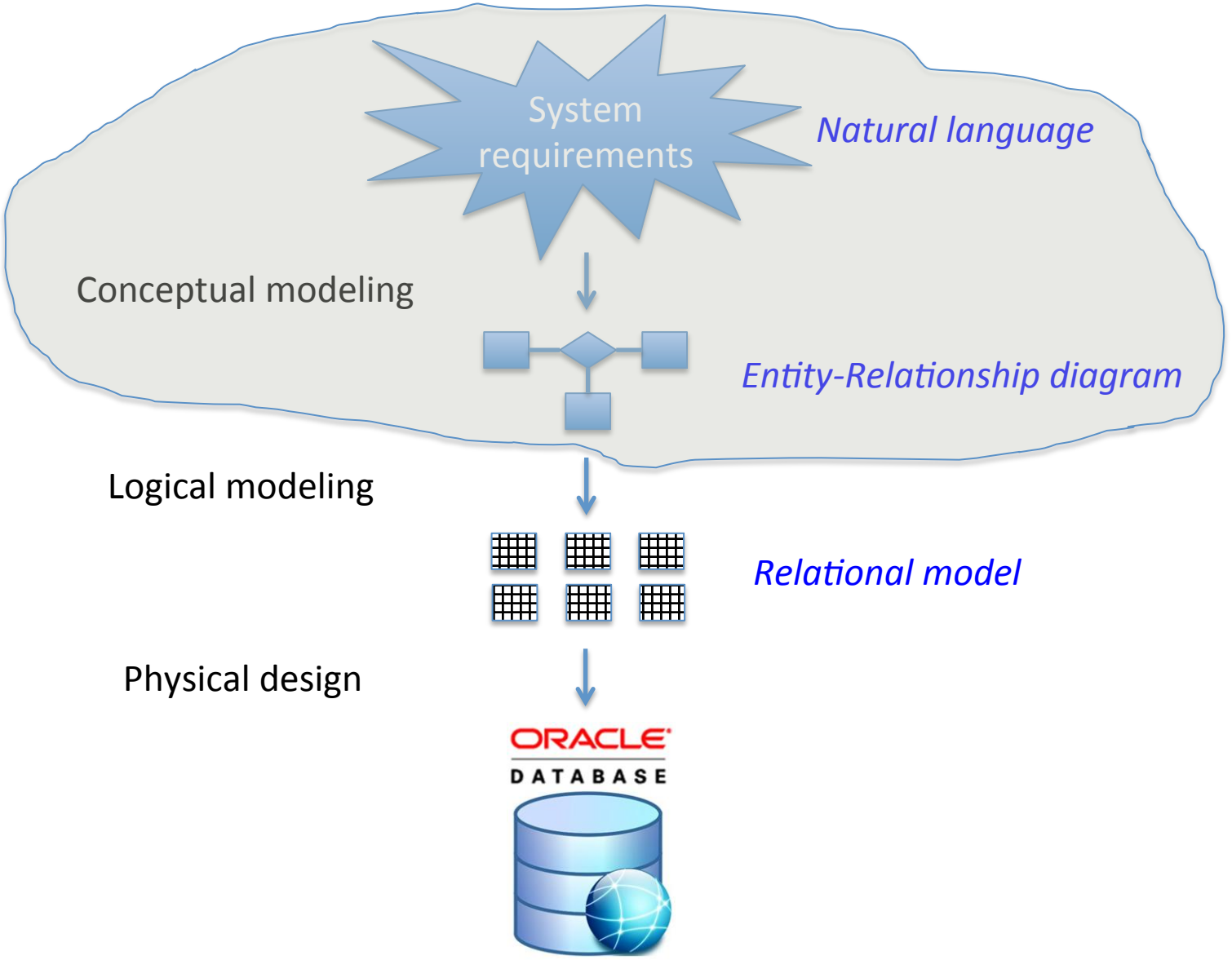
Where are we?



- Motivation & terminology
- ER data modeling 1
- ER data modeling 2
- Relational model
- ER -> relational

UPPSALA UNIVERSITET

Database design: full picture



UPPSALA UNIVERSITET

Intended Learning Outcomes

- Describe the main concepts of the relational model.
  - Relations, attributes, tuples.
  - First Normal Form.
  - Null values.
  - Superkeys, candidate keys, primary key.
  - Foreign keys.
- Create the corresponding structures using a DBMS.

UPPSALA UNIVERSITET

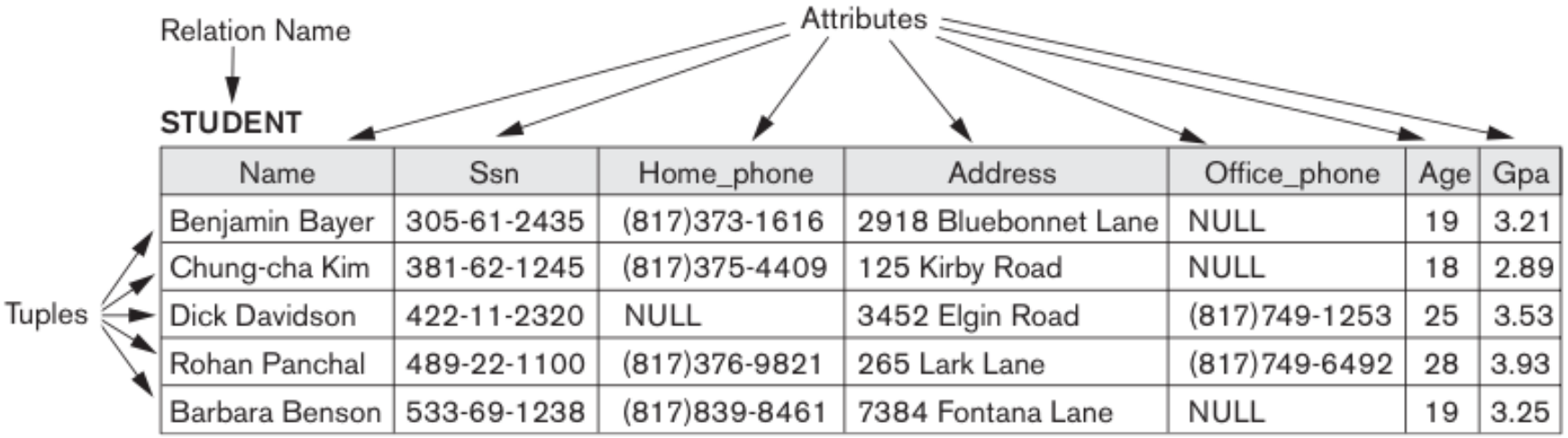
Some history

- The relational model was introduced by Dr. Edgar (Ted) F. Codd (1924-2003) in 1970.
  - A mathematician from Oxford (UK), working as a researcher at the IBM San Jose Research Laboratory (USA).
- Many DBMSs are based on the relational data model.

UPPSALA UNIVERSITET

Relational Model Concepts

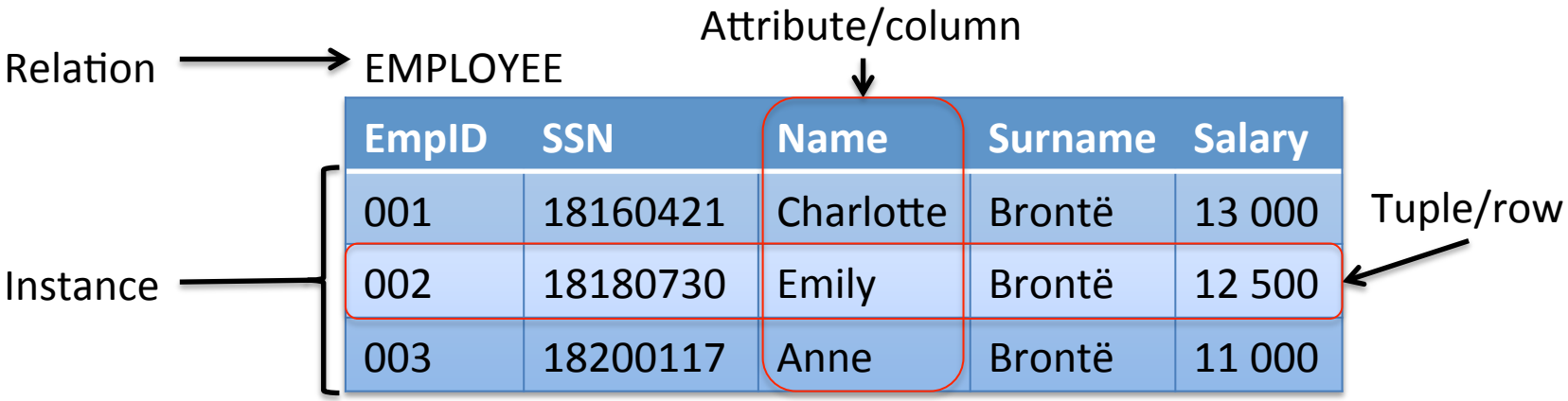
- Represents the database as a collection of relations.



Name	Ssn	Home_phone	Address	Office_phone	Age	Gpa
Benjamin Bayer	305-61-2435	(817)373-1616	2918 Bluebonnet Lane	NULL	19	3.21
Chung-cha Kim	381-62-1245	(817)375-4409	125 Kirby Road	NULL	18	2.89
Dick Davidson	422-11-2320	NULL	3452 Elgin Road	(817)749-1253	25	3.53
Rohan Panchal	489-22-1100	(817)376-9821	265 Lark Lane	(817)749-6492	28	3.93
Barbara Benson	533-69-1238	(817)839-8461	7384 Fontana Lane	NULL	19	3.25

Basic concepts: from relations to tables (DB)

- $A_1, A_2, \ldots, A_n$ are attributes/columns.
- $R(A_1, A_2, \ldots, A_n)$ is a relation schema on these attributes.
  - Employee(EmpID, SSN, Name, Surname, Salary)
- r is a relation on the relation schema R
  - That is, the set of tuples/rows.
- The current values (*relation instance*) of a relation are specified by a table.

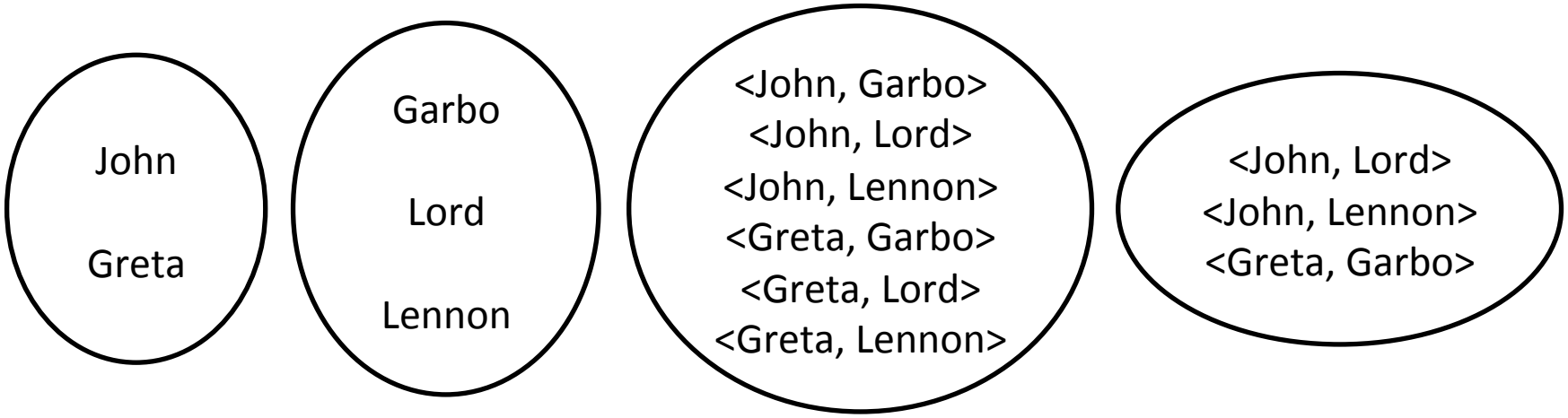


EmpID	SSN	Name	Surname	Salary
001	18160421	Charlotte	Brontë	13 000
002	18180730	Emily	Brontë	12 500
003	18200117	Anne	Brontë	11 000

UPPSALA UNIVERSITET

Basic concepts: Relations (math)

- In set theory, a relation is defined as a **subset** of the **product set** (*Cartesian* or *cross product*) of a number of **domains** (value sets).
- Members of a relation are called **tuples**.
- **Degree** of a relation is number of attributes.
- If the relation is of **degree** n, the tuples are called *n-tuples*.



UPPSALA UNIVERSITET

Data types of domains

- **Data types in SQL:**
  - **char(n).** Fixed length character string, with user-specified length n.
  - **varchar(n)**. Variable length character strings, with user-specified maximum length n.
  - **int.** Integer (a finite subset of the integers that is machine-dependent).
  - **smallint.** Small integer (a machine-dependent subset of the integer domain type).
  - **numeric(p,d).** Fixed point number, with user-specified precision of p digits, with d digits to the right of decimal point.
  - **real, double precision.** Floating point and double-precision floating point numbers, with machine-dependent precision.
  - **float(n).** Floating point number, with user-specified precision of at least n digits.
  - **date.** Dates, containing a (4 digit) year, month and date.
  - **time.** Time of day, in hours, minutes and seconds.
  - **etc.**

UPPSALA UNIVERSITET

Data types and domains

- Domain is a set of **atomic** values.
- Data **type or format** is specified for each domain

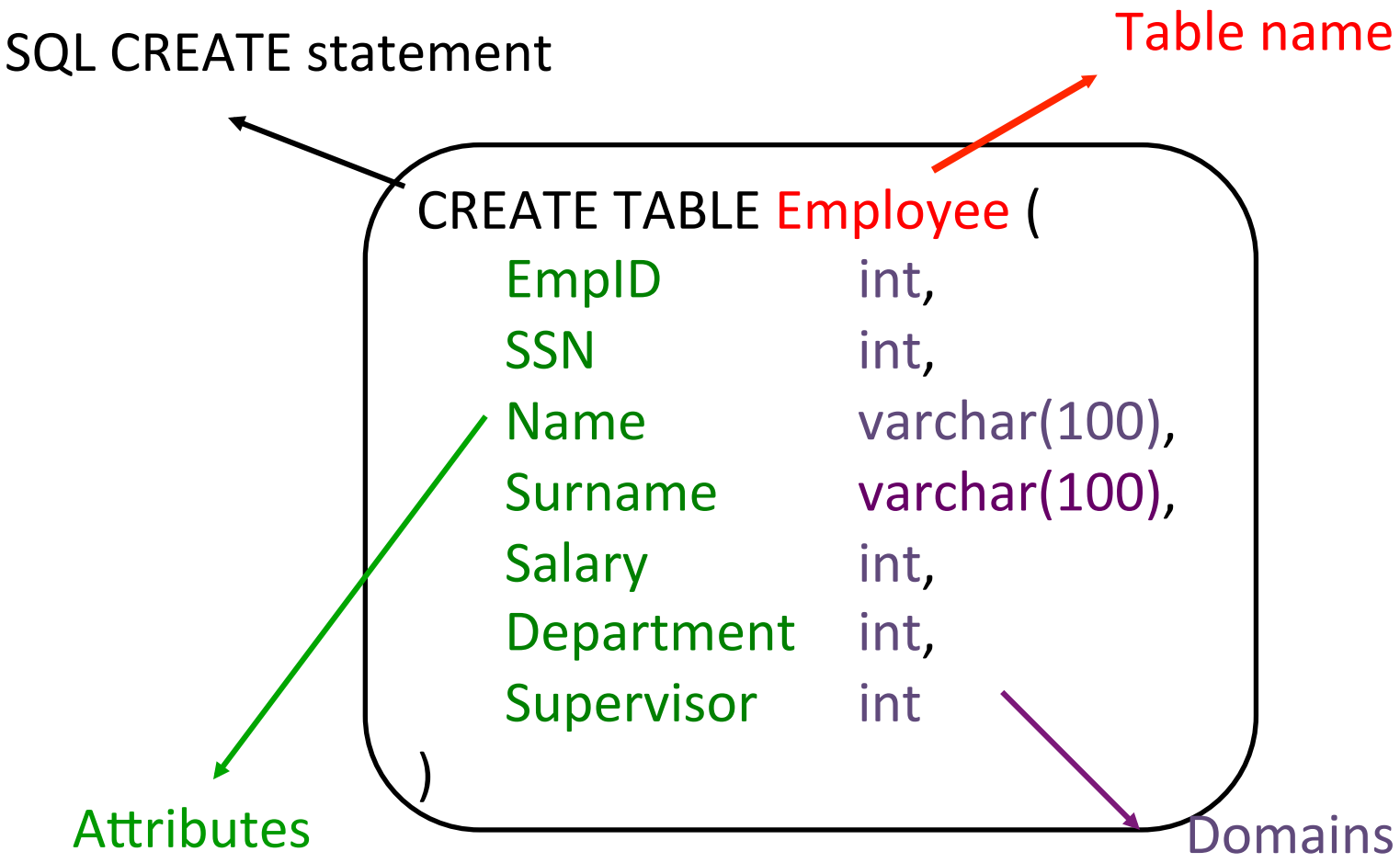
Example:

STUDENT (Name: string, Ssn: string, Home-phone: string, Address: string, Office-phone: string, Age: integer, Gpa: real)

Name	Ssn	Home_phone	Address	Office_phone	Age	Gpa
Benjamin Bayer	305-61-2435	(817)373-1616	2918 Bluebonnet Lane	NULL	19	3.21
Chung-cha Kim	381-62-1245	(817)375-4409	125 Kirby Road	NULL	18	2.89
Dick Davidson	422-11-2320	NULL	3452 Elgin Road	(817)749-1253	25	3.53
Rohan Panchal	489-22-1100	(817)376-9821	265 Lark Lane	(817)749-6492	28	3.93
Barbara Benson	533-69-1238	(817)839-8461	7384 Fontana Lane	NULL	19	3.25

UPPSALA UNIVERSITET

Creation of a new relation (SQL)



UPPSALA UNIVERSITET

Insertion of new data into relation

```
INSERT INTO Employee
VALUES (1,19851,'John','McEnroe',43000,4,13)
```

```
CREATE TABLE Employee (
    EmpID          int,
    SSN            int,
    Name           varchar(100),
    Surname        varchar(100),
    Salary         int,
    Department     int,
    Supervisor     int
)
```

Basic concepts: First Normal Form (1NF)

First Normal Form

- Only **simple/atomic** values are allowed in the relational model.
  - E.g., attributes are not allowed to have composite or multiple values, like an attribute "phone" with values {00461254362, 003933874638 4} for a single employee.

EmpID	SSN	Name	Surname	Salary
001	18160421	Charlotte	Brontë	13 000
002	18180730	Emily	Brontë	12 500
003	18200117	Anne	Brontë	11 000

UPPSALA UNIVERSITET

Feature 1: Null values

- A special value, **null** or ⊥, denote:
  - an unknown value (e.g. home phone no.)
  - a missing value (office phone no.)
  - an attribute that is not applicable (e.g., visa status for foreign status).

Name	Ssn	Home_phone	Address	Office_phone	Age	Gpa
Benjamin Bayer	305-61-2435	(817)373-1616	2918 Bluebonnet Lane	NULL	19	3.21
Chung-cha Kim	381-62-1245	(817)375-4409	125 Kirby Road	NULL	18	2.89
Dick Davidson	422-11-2320	NULL	3452 Elgin Road	(817)749-1253	25	3.53
Rohan Panchal	489-22-1100	(817)376-9821	265 Lark Lane	(817)749-6492	28	3.93
Barbara Benson	533-69-1238	(817)839-8461	7384 Fontana Lane	NULL	19	3.25

UPPSALA UNIVERSITET

Constraints: NULL values

- **Constraints** are restrictions on actual values in a database state.
- Derived from rules in miniworld that database represents.

```
CREATE TABLE Employee (
    EmpID        int,
    SSN          int,
    Name         varchar NOT NULL,
    Surname      varchar NOT NULL,
    Salary       int,
    Department   int,
    Supervisor   int)
```

UPPSALA UNIVERSITET

Insertion of new data

```
CREATE TABLE Employee (
        EmpID           int,
        SSN             int,
        Name            varchar NOT NULL,
        Surname         varchar NOT NULL,
        Salary          int,
        Department      int,
        Supervisor      int)
```

```
INSERT INTO Employee
VALUES (1,1985123,'John','McEnroe',43000,4,13)
```

```
INSERT INTO Employee(EmpID,Name,Surname,SSN)
VALUES (2,'Roger','Federer',198511155482)
```

UPPSALA UNIVERSITET

Feature 2: Tuple identification

- **Superkey:** specifies *uniqueness constraint* that no two distinct tuples in any state *r* of *R* can have same value for set of attributes.
- A set K of the attributes in a relation schema R, i.e. $K \subseteq R$, such that:
  - $t1, t2 \in r(R), t1 \neq t2 \rightarrow t1[K] \neq t2[K]$

Examples of superkeys:

- EmpID, SSN, Name, Surname, Salary
- EmpID, Surname
- SSN
- EmpID, SSN, Name
- EmpID
- …

EmpID	SSN	Name	Surname	Salary
001	18160421	Charlotte	Brontë	13 000
002	18180730	Emily	Brontë	12 500
003	18200117	Anne	Brontë	11 000

UPPSALA UNIVERSITET

Question

Given a relation with 4 attributes, how many superkeys will be present *for sure*?

A: 0

B: 1

C: 4

D: 16

UPPSALA UNIVERSITET

Basic concepts: Candidate keys

- A superkey k is minimal if we cannot remove any attributes and still have the uniqueness constraint i.e. no other superkey K' such that $K' \subset K$.
- Every minimal superkey is called a **candidate key** for R.

Examples of candidate keys:

- EmpID, SSN, Name, Surname, Salary
- EmpID, Surname
- **SSN**
- EmpID, SSN, Name
- **EmpID**
- …

EmpID	SSN	Name	Surname	Salary
001	18160421	Charlotte	Brontë	13 000
002	18180730	Emily	Brontë	12 500
003	18200117	Anne	Brontë	11 000

UPPSALA UNIVERSITET

Basic concepts: Primary key

- Used to uniquely identify a tuple in R.
- Database designer chooses one of R's candidate keys as **primary key** (or just **key**).
- Preferable to choose primary key with single attribute.

Examples of primary key:

- EmpID, SSN, Name, Surname, Salary
- EmpID, Surname
- **SSN**
- EmpID, SSN, Name
- **EmpID**
- …

EmpID	SSN	Name	Surname	Salary
001	18160421	Charlotte	Brontë	13 000
002	18180730	Emily	Brontë	12 500
003	18200117	Anne	Brontë	11 000

UPPSALA UNIVERSITET

Relational keys: Summary

We need to identify specific tuples in a relation.

- **Superkey:** a set of attributes that allows us to do it.
  - A relation can have many superkeys.
- **Candidate key:** a minimal superkey.
  - A relation can have many candidate keys – normally, less then the superkeys.
  - UNIQUE in SQL.
- **Primary key:** the single candidate key we choose to identify records in a table.
  - PRIMARY KEY in SQL.
  - No NULLs allowed.

UPPSALA UNIVERSITET

Constraints: Candidate keys

```
CREATE TABLE Employee (
    EmpID          int UNIQUE,
    SSN            int UNIQUE,
    Name           varchar,
    Surname        varchar,
    Salary         int,
    Department     int,
    Supervisor     int,
)
```

UPPSALA UNIVERSITET

Constraints: Primary key

```
CREATE TABLE Employee (
    EmpID          int PRIMARY KEY,
    SSN            int UNIQUE,
    Name           varchar(100) NOT NULL,
    Surname        varchar(100) NOT NULL,
    Salary         int,
    Department     int,
    Supervisor     int,
    )
```

UPPSALA UNIVERSITET

Exercise: Key constraints

A1	A2	A3	A4
a	1	alpha	0
a	2	beta	0
b	3	gamma	0
b	4	beta	0
b	5	beta	0
c	1	alpha	0
d	2	gamma	0

A) A2 is a candidate key
B) A4 is the primary key
C) A1, A2 is a candidate key
D) A2, A3 is superkey
E) None of the previous answers
F) No answer

UPPSALA UNIVERSITET

Basic concepts: Foreign key constraints

- References between relations are made *by value* using ***foreign keys***.
- "Manager" is a foreign key referencing the EmpID key in the Employee relation.

ProjectID	Manager
PR123	001
PR153	002
Pr12	003

EmpID	SSN	Name	Surname	Salary
001	18160421	Charlotte	Brontë	13 000
002	18180730	Emily	Brontë	12 500
003	18200117	Anne	Brontë	11 000

UPPSALA UNIVERSITET

Constraints: Referential integrity

- Specified between **two relations** to maintain consistency.
- A tuple in one relation that refers to another relation must refer to an **existing** tuple in that relation.

```
CREATE TABLE Employee (
        EmpID               int PRIMARY KEY,
        SSN             int UNIQUE,
        Name                varchar NOT NULL,
        Surname             varchar NOT NULL,
        Salary              int,
        Department  int,
        Supervisor      int references Employee(EmpID),
        )
```

UPPSALA UNIVERSITET

Exercise: Identify the Foreign Keys

EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno

DEPARTMENT

Dname	Dnumber	Mgr_ssn	Mgr_start_date

DEPT_LOCATIONS

Dnumber	Dlocation

PROJECT

Pname	Pnumber	Plocation	Dnum

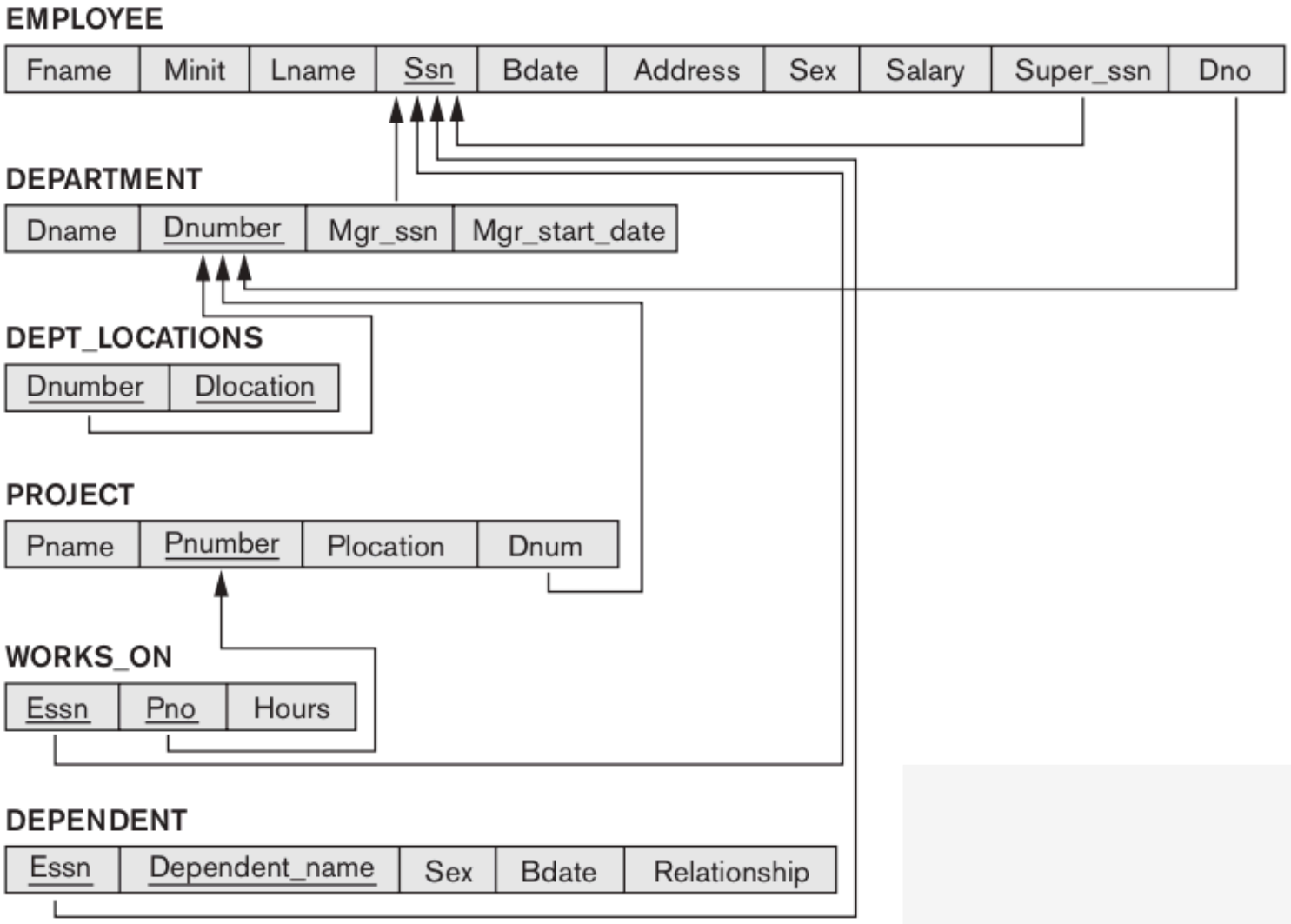
WORKS_ON

Essn	Pno	Hours

DEPENDENT

Essn	Dependent_name	Sex	Bdate	Relationship

Exercise: Identify Referential integrity constraints



UPPSALA UNIVERSITET

A summary

```
CREATE TABLE Employee (
      EmpID          int PRIMARY KEY,
      SSN            int UNIQUE,
      Name           varchar NOT NULL,
      Surname        varchar NOT NULL,
      Salary         int,
      Department     int
        references Department(Number),
      Supervisor     int
        references Employee(EmpID),
)
```

```
CREATE TABLE Department (
      Number int PRIMARY KEY,
      Name   varchar NOT NULL
)
```

```
CREATE TABLE Manager (
      EmpID          int
        references Employee(EmpID),
      Department     int
        references Department(Number),
      StartDate      date,
      PRIMARY KEY(EmpID,Department)
)
```

UPPSALA UNIVERSITET

Insertion, deletion and update

INSERT INTO Employee
VALUES (1, 'John', …

INSERT INTO Manager
<SQL query>

DELETE FROM Employee
WHERE EmpID = 1

UPDATE Employee
SET EmpID = 3
WHERE EmpID = 1

UPPSALA UNIVERSITET

Integrity constraints: Reactions

- Deleting tuples or modifying values in a relation may make references inconsistent.
- We can instruct the database to automatically ensure that referential constraints are kept consistent.
- Possible instructions are:

on delete

< cascade | set null | set default | no action >

on update

< cascade | set null | set default | no action >

Delete reactions

- **cascade**: deletes are propagated.
- **set null**: the referring attribute values are set to null.
- **set default**: the referring attribute values are set to their default.
- **no action**: deletion is not authorized.

UPPSALA UNIVERSITET

Update reactions

- **cascade**: the value is updated also on the referring attributes.
- **set null**: the referring attribute values are set to null.
- **set default**: the referring attribute values are set to their default.
- **no action**: deletion is not authorized.

UPPSALA UNIVERSITET

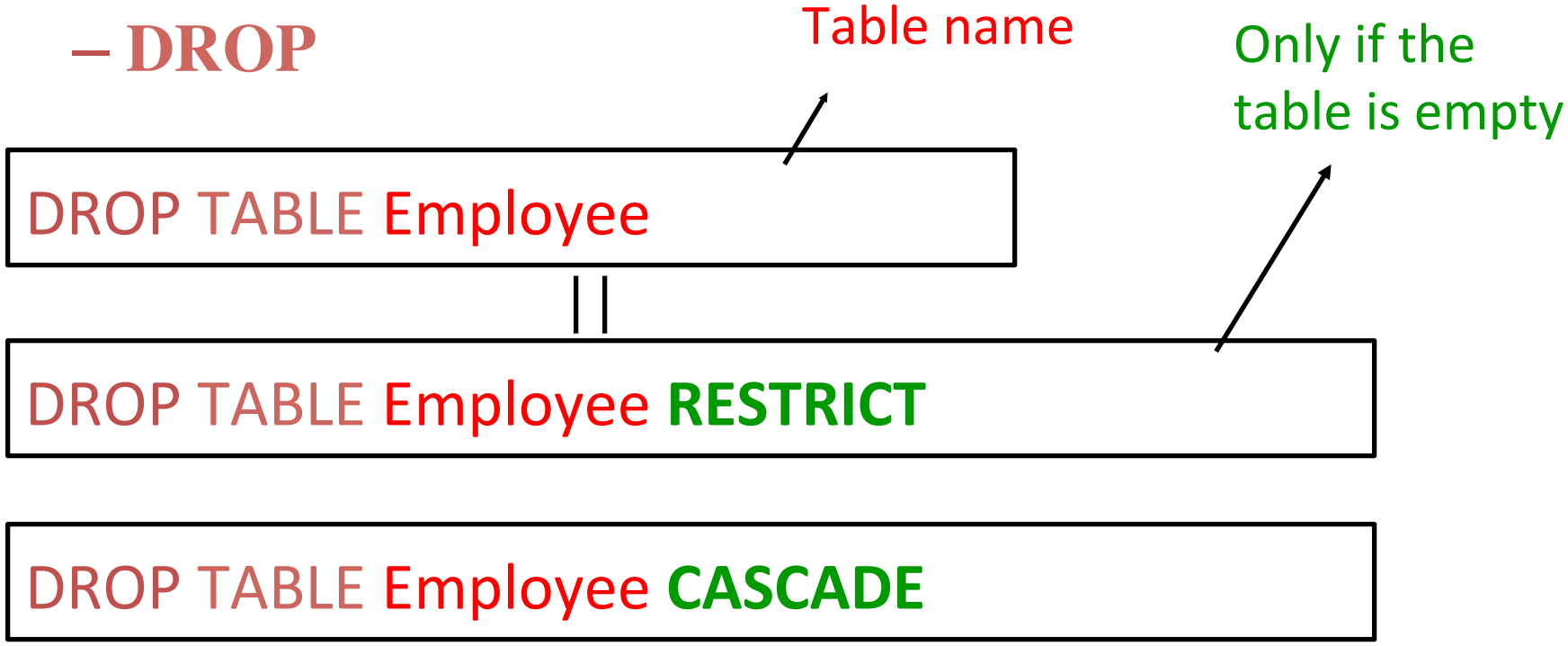
Reactions to deletion / updates

```
CREATE TABLE Employee (
      EmpID          int PRIMARY KEY,
      SSN            int UNIQUE,
      Name           varchar NOT NULL,
      Surname        varchar NOT NULL,
      Salary         int,
      Department     int
        references Department(Number)
        on delete set null,
      Supervisor     int
        references Employee(EmpID)
        on update cascade,
      unique (Name, Surname)
)
```

UPPSALA UNIVERSITET

Schema updates

- It is possible to modify the design of a relational database using two main statements:
  - **ALTER**
  - **DROP**



UPPSALA UNIVERSITET

Details about the syntax

MySQL 5.7 Reference Manual :: 13 SQL Statement Syntax

Chapter 13 SQL Statement Syntax

Table of Contents [+/-]

This chapter describes the syntax for the SQL statements supported by MySQL.

UPPSALA UNIVERSITET

Relational keys: Summary

We need to identify specific tuples in a relation.

- **Superkey:** a set of attributes that allows us to do it.
  - A relation can have many superkeys.
- **Candidate key:** a minimal superkey.
  - A relation can have many candidate keys – normally, less then the superkeys.
  - UNIQUE in SQL.
- **Primary key:** the single candidate key we choose to identify records in a table.
  - PRIMARY KEY in SQL.
  - No NULLs allowed.
- **Foreign key:** used to "connect" relations and reference tuples in other relations.
  - FOREIGN KEY … REFERENCES in SQL.