Data modeling

Data modeling

Data modeling is the process of creating model (e.g. visual representation) of either a whole information system or parts of it to communicate data structures and connections/relationships between data.

Data models

Data Model - a collection of "tools" for describing

- Data
- Data relationships
- Data semantics
- Data constraints

Data models c.d.

Various data models

- Entity-Relationship data model (mainly for database design)
- Relational model
- Object-based data models (Object-oriented and Object-relational)
- Other older models:
  - document based, semi-structured data model (XML, JSON)
  - graph

ER model

- Entities
- Attributes
- Relationships
  - `1 – 1` one to one
  - `1 – N` one to many
  - `N – M` many to many

ER model c.d.

Entities and attributes - shop (e.g. northwind database)

- `products`
  - `productname`
  - `quantityperunit`
  - `unitprice`
  - ...
- `categoris`
  - `categoryname`
  - `description`
- `suppliers`
  - `companyname`
  - `address`
  - `city`
  - country
  - ...

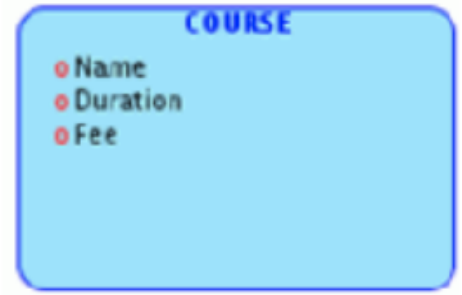
ER model c.d.

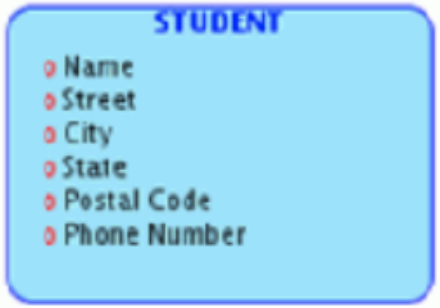
Entities and attributes - university

- `course`
  - `name`
  - `duration`
  - `fee`
- `student`
  - `name`
  - `street`
  - `city`
  - ...
- `instructor`
  - `name`
  - `street`
  - `city`
  - `salary`
  - ...

ER model c.d.

Entitie and attributes





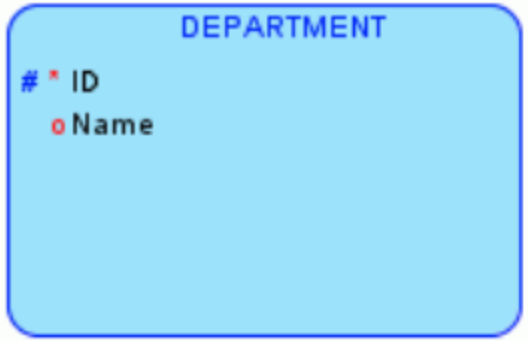
ER model c.d.

Entities and attributes - company - HR

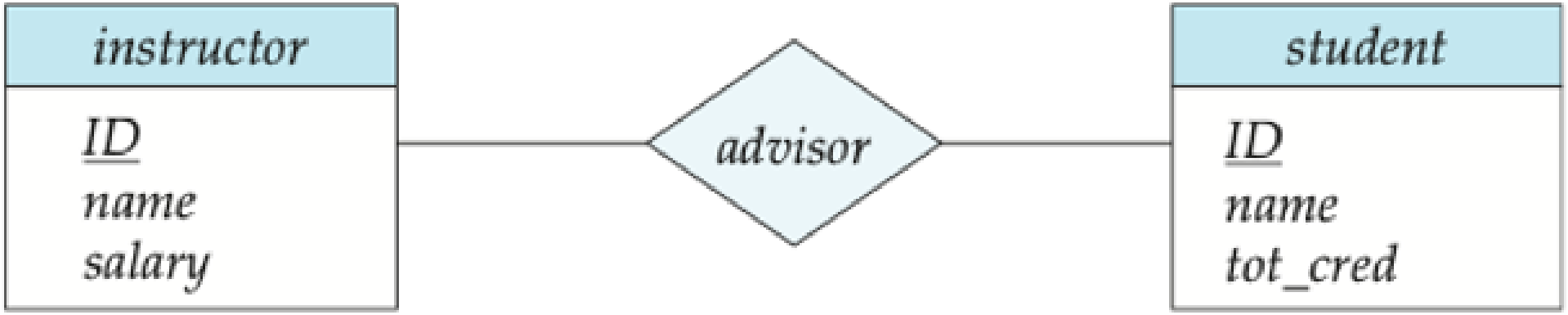
- `department`
  - `name`
- employee
  - `first name`
  - `last name`
  - `email`
  - `phone number`
  - `hire date`
  - `salary`

ER model c.d.

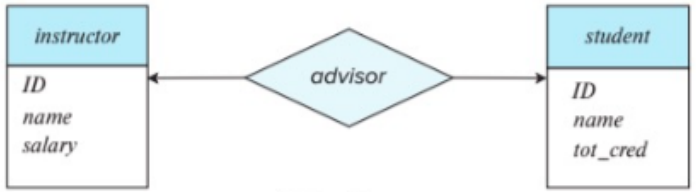
Identifiers



ER diagram



ER diagram c.d.



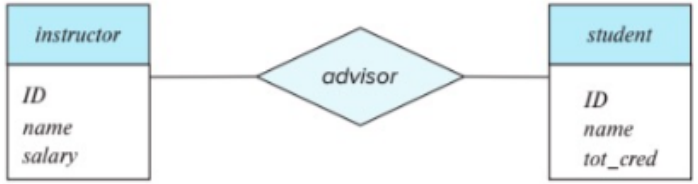
(a) One-to-one

(b) One-to-many

instructor may advise many students,
student may have at most one advisor

(c) Many-to-one

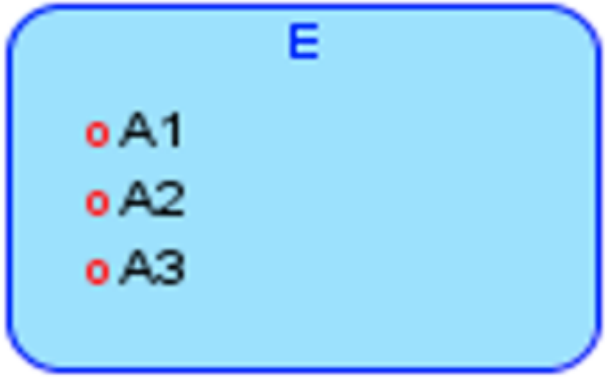
instructor may advise at most one student,
student may have many advisors



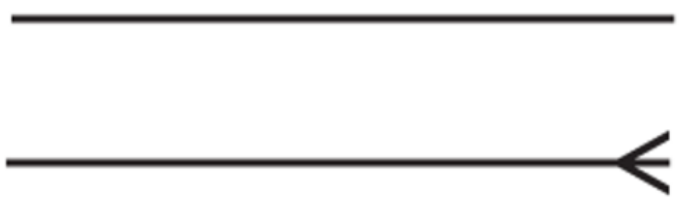
(d) Many-to-many

ER diagram - Barker notation

Entity/Entity set



Relationship



Relationships

one to one

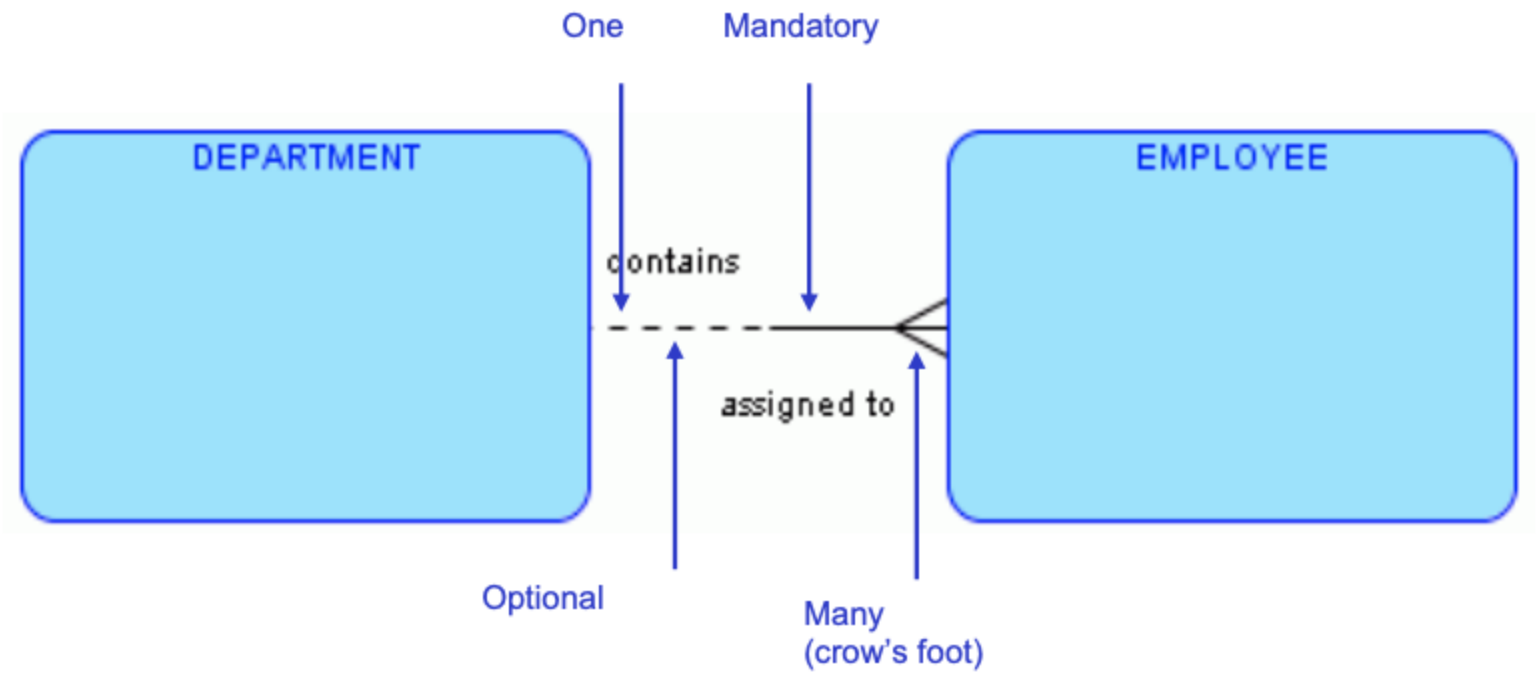
one to many

many to many

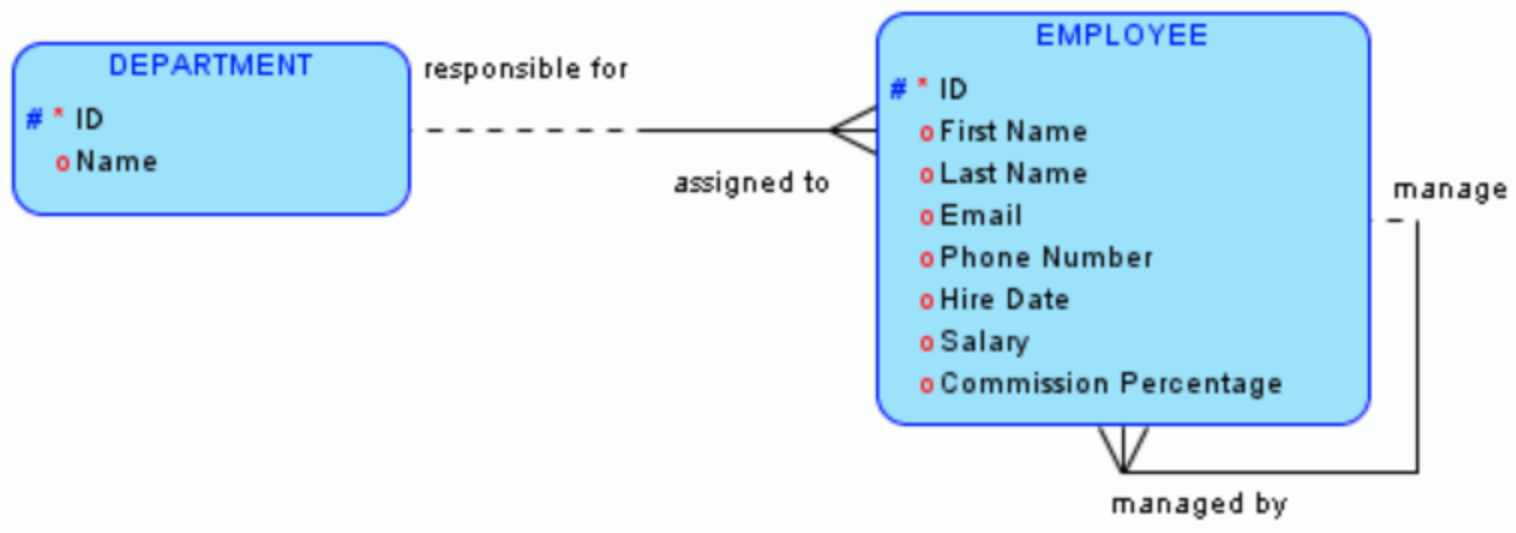
mandatory

optional

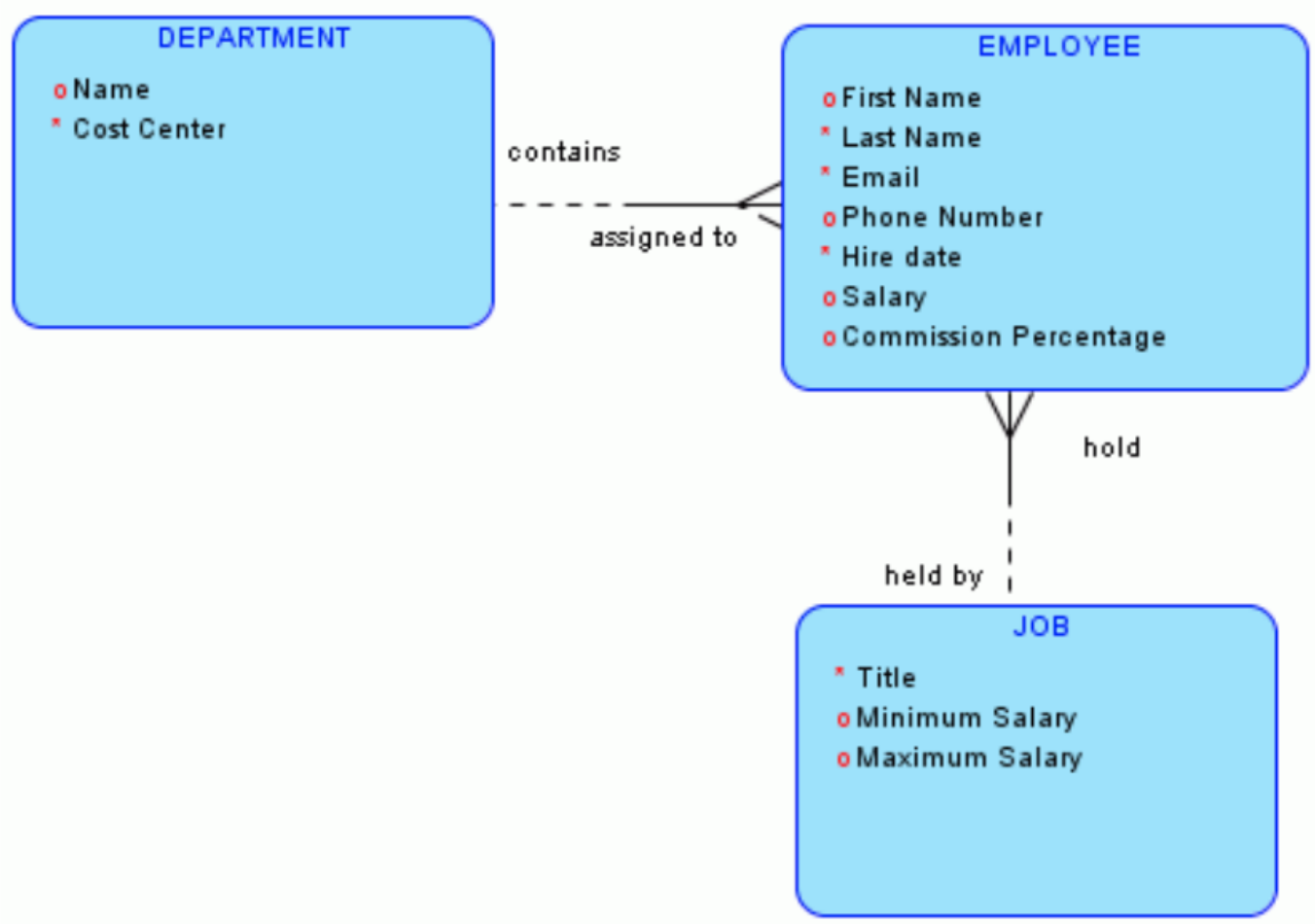
Relationships c.d.



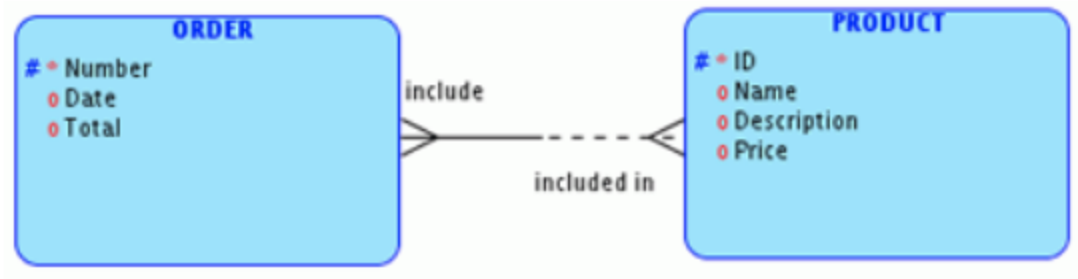
Relationships c.d.

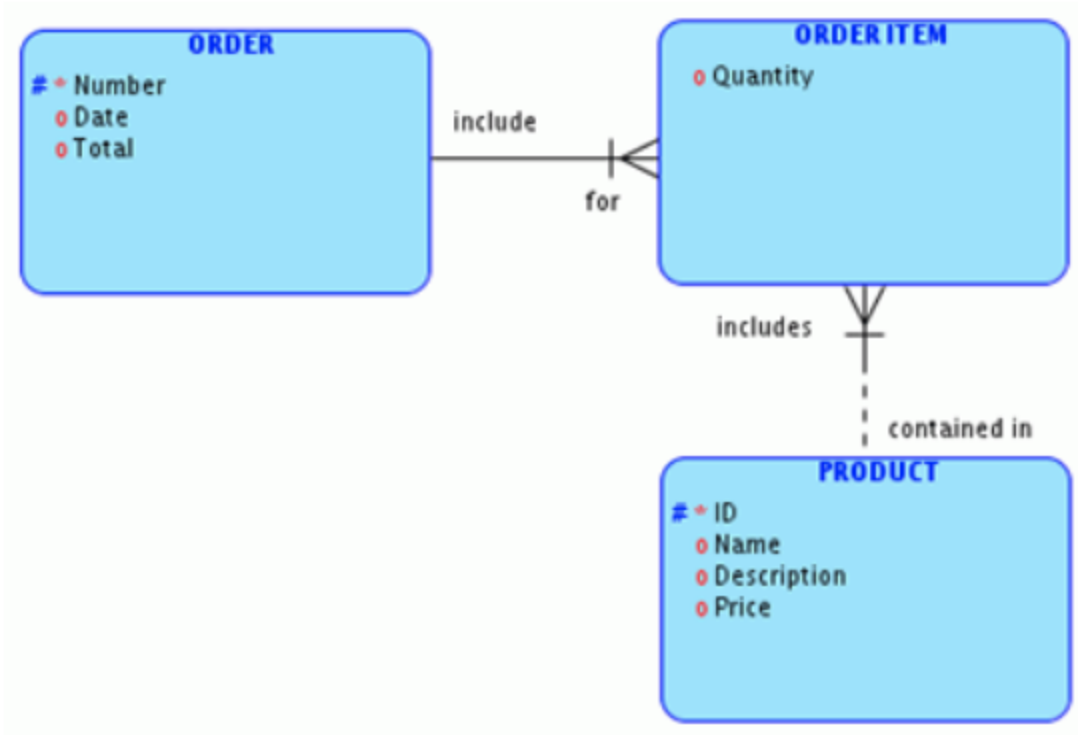


ER model c.d.


DEPARTMENT
o Name
* Cost Center
contains
assigned to
EMPLOYEE
o First Name
* Last Name
* Email
o Phone Number
* Hire date
o Salary
o Commission Percentage
hold
held by
JOB
* Title
o Minimum Salary
o Maximum Salary

Many to many relationship





Relational model

- `A1, A2, …, An` - attributes
- `R = (A1, A2, …, An )` is a relation schema
    - Example:
        - `instructor  = (ID,  name, dept_name, salary)`
- Formally, given sets `D1, D2, …. Dn` a relation `r` is a subset of `D1 x  D2  x … x Dn`. Thus, a relation is a set of n-tuples `(a1, a2, …, an)` where each `ai` belongs to `Di`

ER model - Relational model

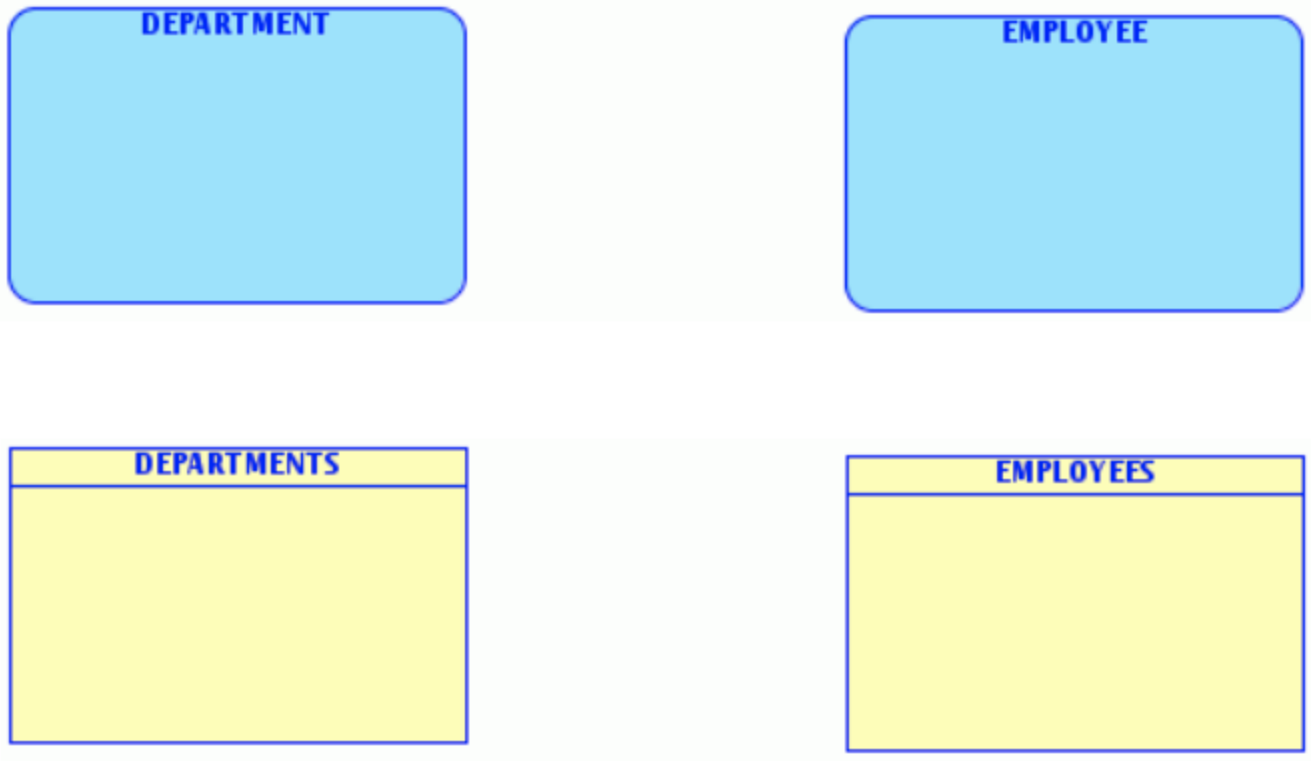
ER- model

- Entities
  - Attributes
  - Identifiers
- Relationship

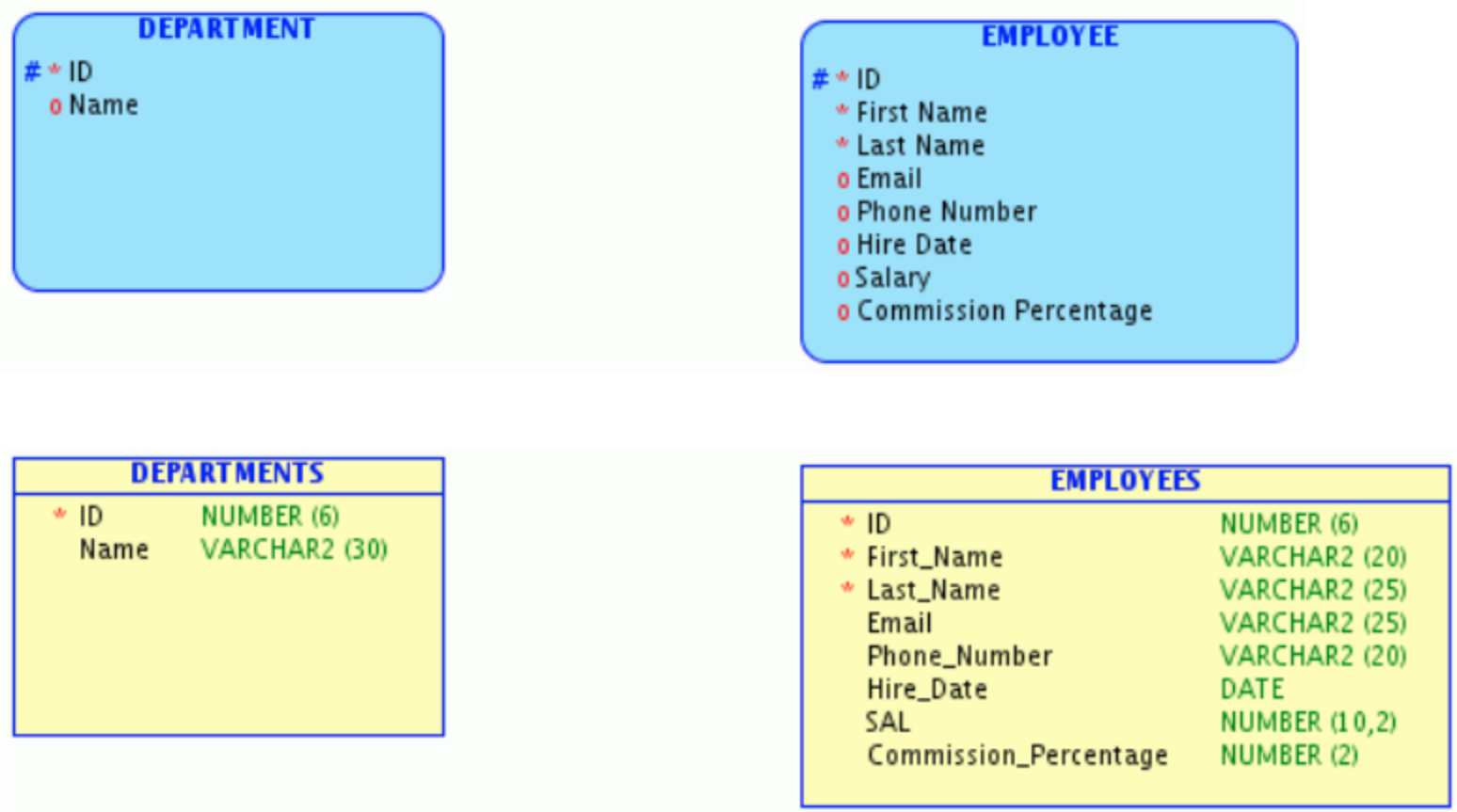
Relational model

- Relations - Tables
  - Attributes
  - Keys
- Relationships

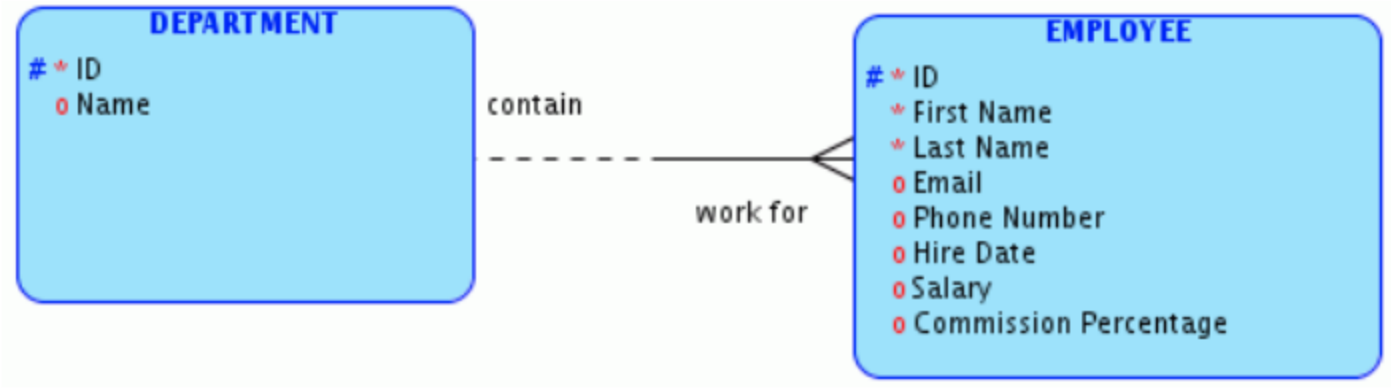
ER model - Relational model c.d.

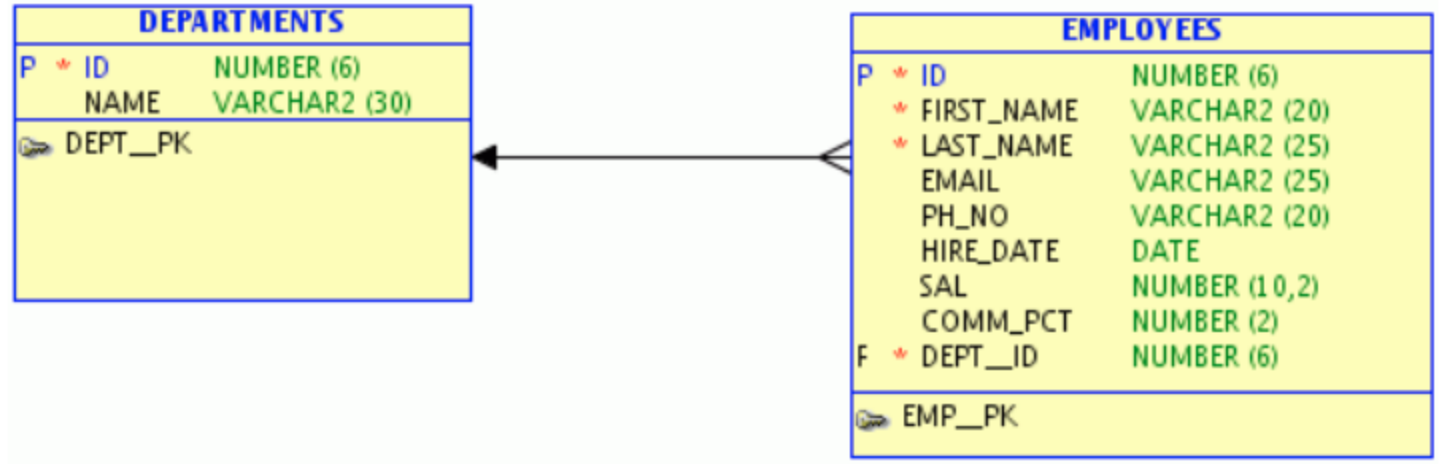


ER model - Relational model c.d.



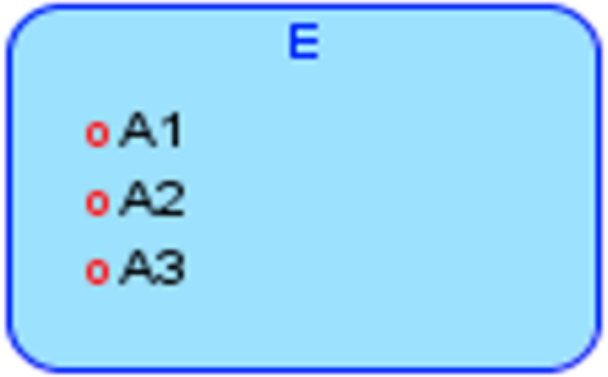
ER model - Relational model c.d.



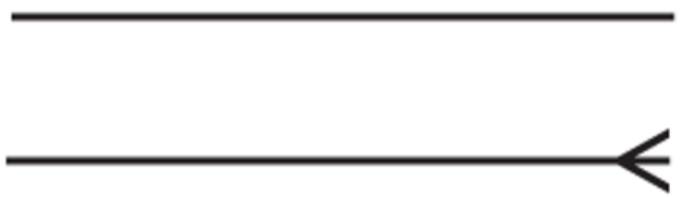


Crow’s Foot notation

Entity/Entity set



Relationship



Crow’s Foot notation c.d.

one to one

one to many

optional

mandatory

Crow’s Foot notation c.d.

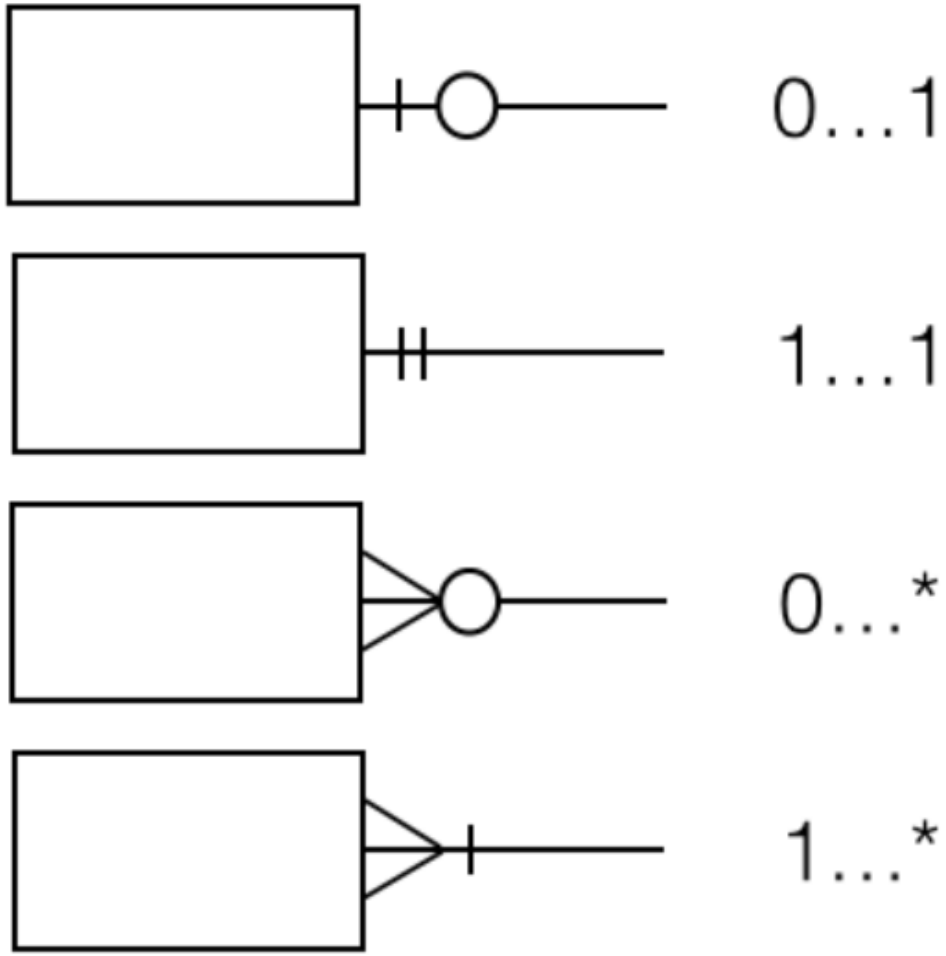
- 1 - (0 lub 1, optional)

- 1 - (1, mandatory)

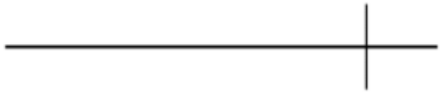
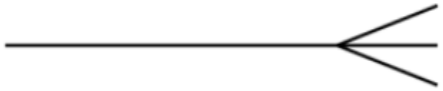
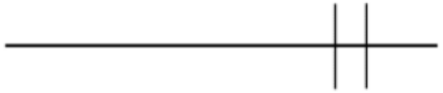
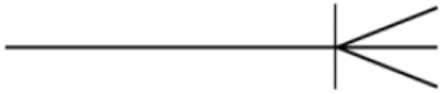
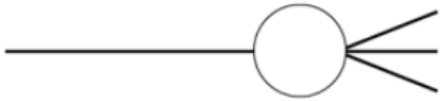
- N – (0 or N, optional)

- N – (1 or N, mandatory)

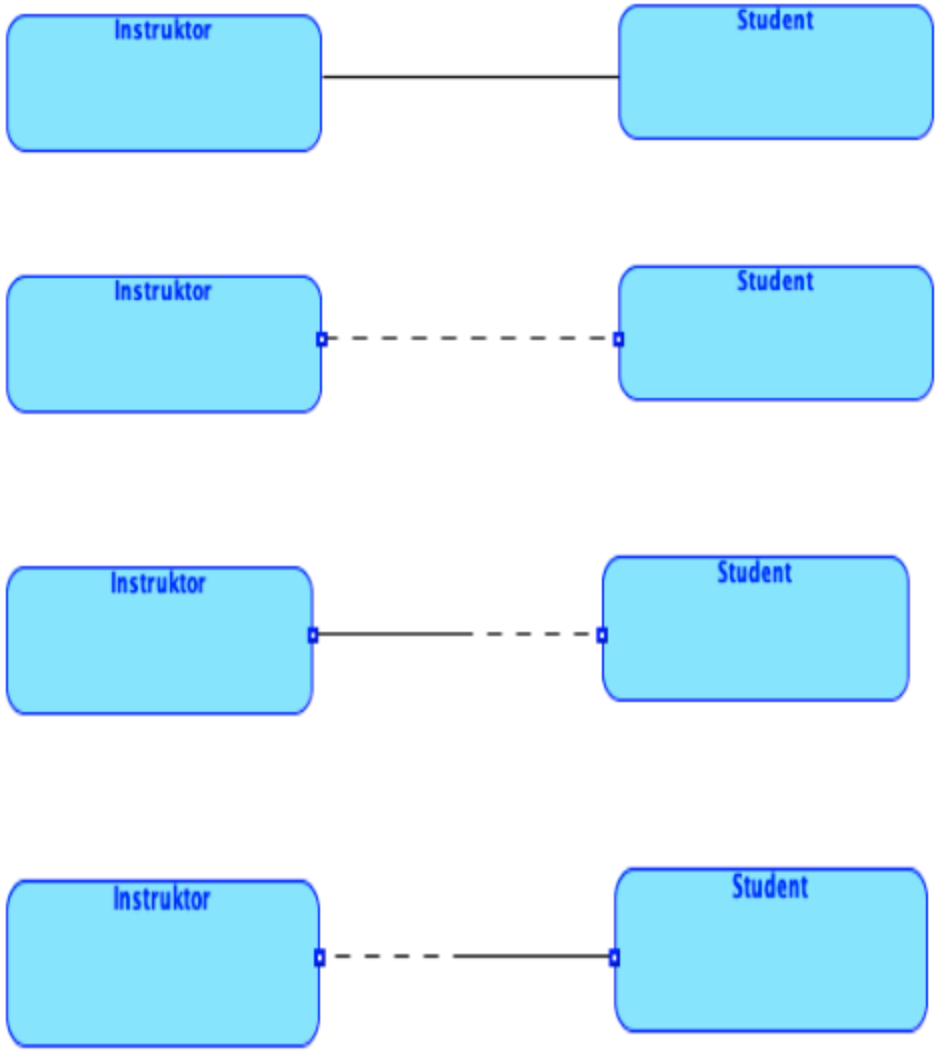
Crow's Foot notation c.d.



Crow's Foot notation c.d.

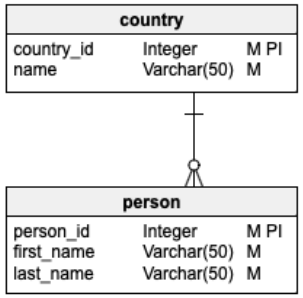
Symbol	Meaning	Number
	One	N/A
	Many	N/A
	Mandatory-One	Exactly one
	Optional-One	Zero or one
	Mandatory-Many	One or More
	Optional-Many	Zero or more

Barker vs Crow's Foot

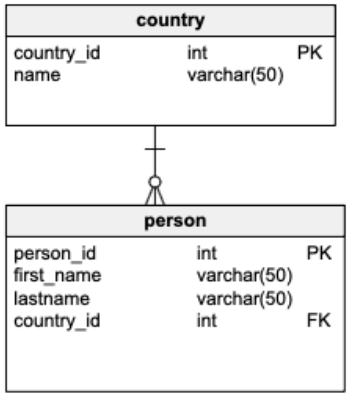


ER model - Relational model

ER model - logical



Relational - phisical

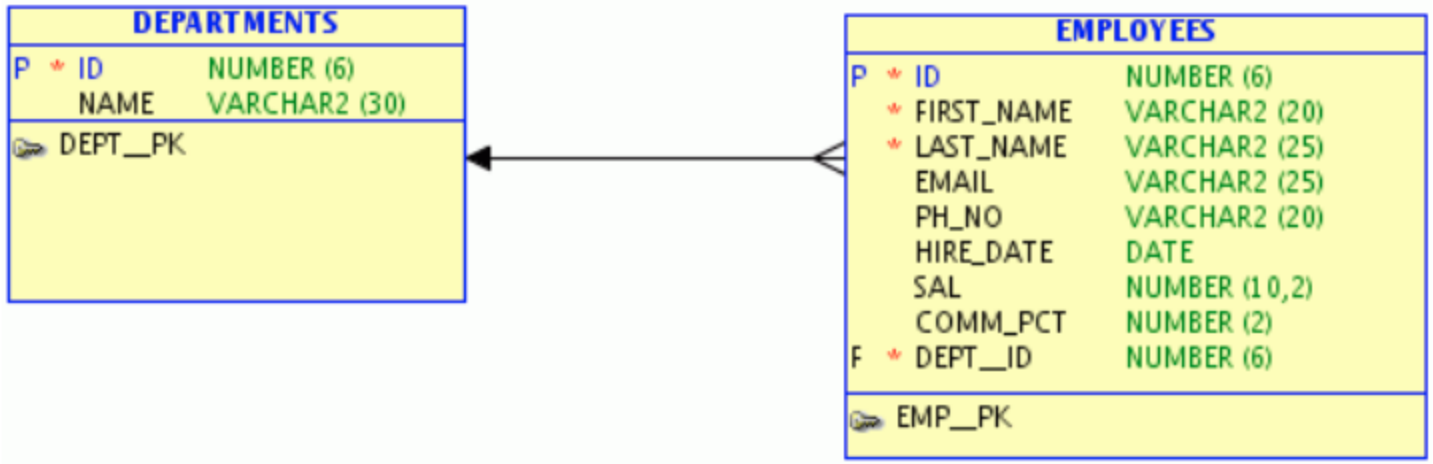


Database definition

SQL

- DDL - Data definition langiage
  - `create`
  - `alter`
  - `drop`

Model


DEPARTMENTS
P * ID NUMBER (6)
NAME VARCHAR2 (30)
DEPT__PK
EMPLOYEES
P * ID NUMBER (6)
* FIRST_NAME VARCHAR2 (20)
* LAST_NAME VARCHAR2 (25)
EMAIL VARCHAR2 (25)
PH_NO VARCHAR2 (20)
HIRE_DATE DATE
SAL NUMBER (10,2)
COMM_PCT NUMBER (2)
F * DEPT__ID NUMBER (6)
EMP__PK

SQL DDL

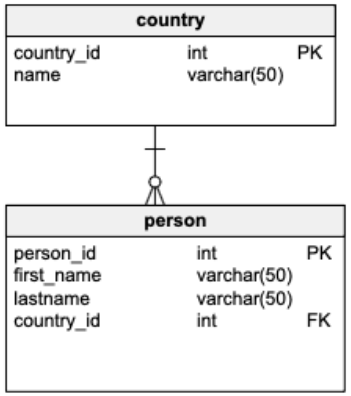
```
create table departments
(
    id number(6) not null,
    name varchar2(50)
);


alter table departments
    add constraint department_pk primary key (id);


create table employees
(
    id number(6) not null,
        first_name varchar2 (50),
    last_name varchar2 (50),
    email varchar2(30),
    phone_number varchar2 (20),
    hire_date date,
    salary number(8, 2),
    commission_percentage number (2,2),
    department_id number(6) not null
);

alter table employees
    add constraint employee_pk primary key (id);
alter table employees
    add constraint relation_1 foreign key (department_id)
        references departments (id);
```

Model


country
country_id int PK
name varchar(50)
person
person_id int PK
first_name varchar(50)
lastname varchar(50)
country_id int FK

SQL DDL

```
create table country (
    country_id int  not null,
    name varchar(50)  not null,
    constraint country_pk primary key (country_id)
);

-- table: person
create table person (
    person_id int  not null,
    first_name varchar(50)  not null,
    lastname varchar(50)  not null,
    country_id int  not null,
    constraint person_pk primary key (person_id)
);

-- foreign keys
-- reference: person_country (table: person)
alter table person add constraint person_country
    foreign key (country_id)
    references country (country_id)
not deferrable
    initially immediate
;
```

Data modeling tools

- Vertabello
- Oracle data modeler
- SSMS
- ...