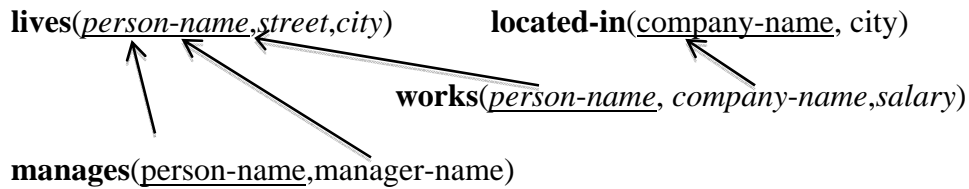


Schema for Relational Algebra and Calculus In-Class Exercises

Schema: the **Corporation** database



The primary key for each relation is denoted by the underlined attribute.

The foreign keys are indicated by the arrow – for example, *person-name* in **works** references *person-name* in **lives**.

When you write the queries, for brevity you can choose to use *p-name*, *c-name*, *m-name* for *person-name*, *company-name*, *manager-name* respectively.

Relational Database Example Schemas.

There are two database schemas we will frequently use in our examples: (1) the Company database and (2) Bank database.

1. COMPANY Database

The COMPANY database keeps track of Employees and Departments, the Projects the Company has, and dependents of the employees (for benefits and tax purposes).

- Employees identified by SSN, Name, Location
- Department specified by Department ID (did), a department Name, Budget
- Each department has a unique manager, who is also an employee of the Company.
 - The Database must keep track of starting date
- No two departments have the same name
- A person can manage at most one department
- A person can manage a department even if they do not work in the department
- Each employee works in a department
 - Database must keep track of starting date
- An employee can only be assigned to one department
- A department can have zero employees
- Every employee is assigned a Supervisor (i.e, the person that directly manages the employee).
- The Company has several Projects
 - The project has a number, name, and a location for the project (For example, Banner is a project located in Foggy Bottom.)
 - Each project is controlled by a department
- Employees can be assigned to different projects. An employee can work on more than one project.
 - For accounting purposes, we need to track the number of hours an employee works on each project they are assigned to.
- Information on Dependents of employees needs to be stored for employee benefits purposes. The information can be social security number, name and relationship to employee.

Schema for COMPANY database

```
CREATE TABLE EMP (  
    SSN int PRIMARY KEY,  
    Name Varchar(50),  
    Birthdate Date,  
    Street Varchar(50),  
    City Varchar(50),  
    DNO int,  
    SuperSSN int,  
    Salary Real,  
    FOREIGN KEY (DNO) REFERENCES DEPT(DNUM),  
    FOREIGN KEY (SuperSSN) REFERENCES EMP(SSN));
```

```
CREATE TABLE DEPT (  
    DNUM int PRIMARY KEY,  
    DNAME Varchar(50),  
    MGRSSN int,  
    UNIQUE DNAME,  
    FOREIGN KEY (MGRSSN) REFERENCES EMP(SSN));
```

```
CREATE TABLE PROJECTS(  
    PNum int PRIMARY KEY,  
    Pname Varchar(50),  
    Plocation Varchar(50),  
    DNO int,  
    FOREIGN KEY (DNO) REFERENCES DEPT(DNUM));
```

```
CREATE TABLE WORKS(  
    ESSN int,  
    PNO int,  
    HOURS int,  
    PRIMARY KEY (ESSN,PNO),  
    FOREIGN KEY (ESSN) REFERENCES EMP(SSN),  
    FOREIGN KEY (PNO) REFERENCES PROJECTS(PNUM));
```

```
CREATE TABLE DEPENDENT(  
    ESSN int NOT NULL,  
    DEPNAME Varchar(50) NOT NULL,  
    SEX char,  
    BDATE Date,  
    RELATIONSHIP Varchar(10),  
    PRIMARY KEY (ESSN,DEPNAME),  
    FOREIGN KEY (ESSN) REFERENCES EMPLOYEE(SSN));
```

2. The BANK database

A simplified bank database keeps track of customers, their accounts, and branch information.

- Store information on each customer – their unique ID, name, address
- Store information about accounts (checking or saving are assumed to be the same in this example) – the customer who owns the account, the account number, the balance in the account and the branch where the account is housed.
- Some questions to ponder:
 - Can a customer have more than one account (or loan)
 - Can we have joint accounts wherein two customers jointly own an account.
- Store similar information about loans given by the bank.
- Information about each branch is stored – the branch location, the total assets in the branch, and the location (city) of the branch.
- Primary key is underlined (for each table)

Schema for Bank database:

- Customer (CustID, Name, street,city,zip)
Customer ID, Name, and Address info: street, city, zip
- Deposit (CustID, Acct-num, balance,Branch-name)
Customer ID, Account number, Balance in account, name of branch where account is held;
CustID is foreign key referencing Customer.
Branch-name is foreign key referencing Branch relation
- Loan (CustID, Loan-num, Amount, Branch-name)
Customer ID, loan number, amount of loan;
CustID is foreign key referencing Customer relation;
Branch-name is foreign key referencing Branch relation.
- Branch (Branch-name, assets, Branch-city)
Name of the branch (unique name), assets in dollars, and the city where the branch is located.