

CS2102

Tutorial 02

Question 7

- **Domain considerations:**

- building names are usually characters; 100 should be sufficient
- area names are usually characters; 100 should be sufficient
- level and roomNumber are typically integer
- officeID depends on the company, simplistically, can just be an integer

- **Attempt #1**

```
CREATE TABLE Offices (  
    officeID      integer,  
    building      varchar(100),  
    level         integer,  
    roomNumber    integer,  
    area          varchar(100)  
);
```

Question 7

- **Key constraints:**

- officeID is the primary key
- {building, level, roomNumber} is a candidate key
 - unique and not null

- **Attempt #2**

```
CREATE TABLE Offices (  
    officeID      integer      PRIMARY KEY,  
    building      varchar(100) NOT NULL,  
    level         integer      NOT NULL,  
    roomNumber    integer      NOT NULL,  
    area          varchar(100),  
    UNIQUE (building, level, roomNumber)  
);
```

Question 7

- **Domain considerations:**

- name of employees are characters; 100 may be sufficient
- empID, officeID, and managerID depends on the company, simplistically, can just be an integer
 - officeID must be the same type as Offices.officeID

- **Attempt #1**

```
CREATE TABLE Offices (  
    empID        integer,  
    name         varchar(100),  
    officeID     integer,  
    managerID    integer  
);
```

Question 7

- **Key constraints:**

- empID is the primary key and name is non-null
- Exactly one officeID; empID is key, so officeID must be non-null
- At most one managerID; can be 0, so managerID may be null since empID is key

- **Attempt #2**

```
CREATE TABLE Offices (  
    empID          integer          PRIMARY KEY,  
    name           varchar(100) NOT NULL,  
    officeID       integer          NOT NULL,  
    managerID      integer,  
    FOREIGN KEY (officeID) REFERENCES Offices (officeID),  
    FOREIGN KEY (managerID) REFERENCES Employees (empID)  
);
```

Question 8

- **Domain considerations:**

- sid integer
- mentee integer (same as sid)
- grade numeric (may be decimal)

- **Attempt #1**

```
CREATE TABLE Offices (  
    sid      integer,  
    mentee   integer,  
    grade    numeric  
);
```

Question 8

- **Constraints considerations:**

- sid is primary key
- Student.mentee references Student.sid
- Cannot mentor student unless grade ≥ 4.0
 - Either Student.mentee is null or grade ≥ 4.0
- Student.mentee cannot be the same as Student.sid

- **Attempt #2**

```
CREATE TABLE Offices (  
    sid      integer PRIMARY KEY,  
    mentee   integer REFERENCES Offices (sid),  
    grade    numeric,  
    CHECK (sid <> mentee),  
    CHECK (mentee IS NULL OR grade  $\geq 4.0$ )  
);
```

Question 9

- **SQL**

```
CREATE TABLE Cars (  
    plateNum integer PRIMARY KEY,  
    company varchar(100) NOT NULL,  
    type integer NOT NULL,  
    model varchar(5) NOT NULL,  
    capacity integer,  
    UNIQUE (company, type),  
    UNIQUE (type, model, capacity),  
    UNIQUE (plateNum, model)  
);
```


Question 9

- **SQL**

```
CREATE TABLE Cars (  
    plateNum    integer        PRIMARY KEY,  
    company     varchar(100) NOT NULL,  
    type        integer        NOT NULL,  
    model       varchar(5)     NOT NULL,  
    capacity    integer,  
    UNIQUE (company, type),  
    UNIQUE (type, model, capacity),  
    UNIQUE (plateNum, model)  
);
```

- One of the candidate key is selected as primary key
 - Ergo, primary key is a candidate key that was selected
 - plateNum

Question 9

- **SQL**

```
CREATE TABLE Cars (  
    plateNum    integer        PRIMARY KEY,  
    company     varchar(100) NOT NULL,  
    type        integer        NOT NULL,  
    model       varchar(5)     NOT NULL,  
    capacity    integer,  
    UNIQUE (company, type),  
    UNIQUE (type, model, capacity),  
    UNIQUE (plateNum, model)  
);
```

- One of the candidate key is selected as primary key
 - Ergo, primary key is a candidate key that was selected
 - plateNum
- NOT NULL and UNIQUE indicates a candidate key
 - (company, type)

Question 9

- **SQL**

```
CREATE TABLE Cars (  
    plateNum    integer        PRIMARY KEY,  
    company     varchar(100) NOT NULL,  
    type        integer        NOT NULL,  
    model       varchar(5)     NOT NULL,  
    capacity    integer,  
    UNIQUE (company, type),  
    UNIQUE (type, model, capacity),  
    UNIQUE (plateNum, model)  
);
```

- One of the candidate key is selected as primary key
 - Ergo, primary key is a candidate key that was selected
 - plateNum
- NOT NULL and UNIQUE indicates a candidate key
 - (company, type)

Question 9

- **SQL**

```
CREATE TABLE Cars (  
    plateNum    integer        PRIMARY KEY,  
    company     varchar(100) NOT NULL,  
    type        integer        NOT NULL,  
    model       varchar(5)     NOT NULL,  
    capacity    integer,  
    UNIQUE (company, type),  
    UNIQUE (type, model, capacity),  
    UNIQUE (plateNum, model)  
);
```

- One of the candidate key is selected as primary key
 - Ergo, primary key is a candidate key that was selected
 - plateNum
- NOT NULL and UNIQUE indicates a candidate key
 - (company, type)
- Almost, capacity may be null

Question 9

- **SQL**

```
CREATE TABLE Cars (  
    plateNum    integer        PRIMARY KEY,  
    company     varchar(100) NOT NULL,  
    type        integer        NOT NULL,  
    model       varchar(5)     NOT NULL,  
    capacity    integer,  
    UNIQUE (company, type),  
    UNIQUE (type, model, capacity),  
    UNIQUE (plateNum, model)  
);
```

- One of the candidate key is selected as primary key
 - Ergo, primary key is a candidate key that was selected
 - plateNum
- NOT NULL and UNIQUE indicates a candidate key
 - (company, type)

Question 9

- **SQL**

```
CREATE TABLE Cars (  
    plateNum    integer        PRIMARY KEY,  
    company     varchar(100) NOT NULL,  
    type        integer        NOT NULL,  
    model       varchar(5)     NOT NULL,  
    capacity    integer,  
    UNIQUE (company, type),  
    UNIQUE (type, model, capacity),  
    UNIQUE (plateNum, model)  
);
```

- One of the candidate key is selected as primary key
 - Ergo, primary key is a candidate key that was selected
 - plateNum
- NOT NULL and UNIQUE indicates a candidate key
 - (company, type)
- Almost, plateNum already primary key, and (plateNum, model) is its superset

Question 9

- **SQL**

```
CREATE TABLE Cars (  
    plateNum    integer          PRIMARY KEY,  
    company     varchar(100) NOT NULL,  
    type        integer          NOT NULL,  
    model       varchar(5)   NOT NULL,  
    capacity    integer,  
    UNIQUE (company, type),  
    UNIQUE (type, model, capacity),  
    UNIQUE (plateNum, model)  
);
```

- One of the candidate key is selected as primary key
 - Ergo, primary key is a candidate key that was selected
 - **plateNum**
 - NOT NULL and UNIQUE indicates a candidate key
 - **(company, type)**
- ❖ Only these two