10: Database Creation

SQL

(S)TRUCTURED (Q)UERY (L)ANGUAGE

SQL-92

DDL

data definition language
(ch.10/ch.12)

DCL

data control language
(ch.10)

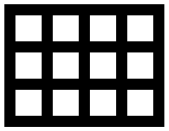
DML

data manipulation language
(ch.10/ch.11)

define
control
manipulate

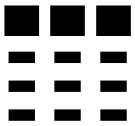
database creation

principal tasks involve...



create and modify database tables

enforce integrity constraints



populate database tables

security and authorizations

defining data in SQL

create

alter

drop

can be applied to:

table

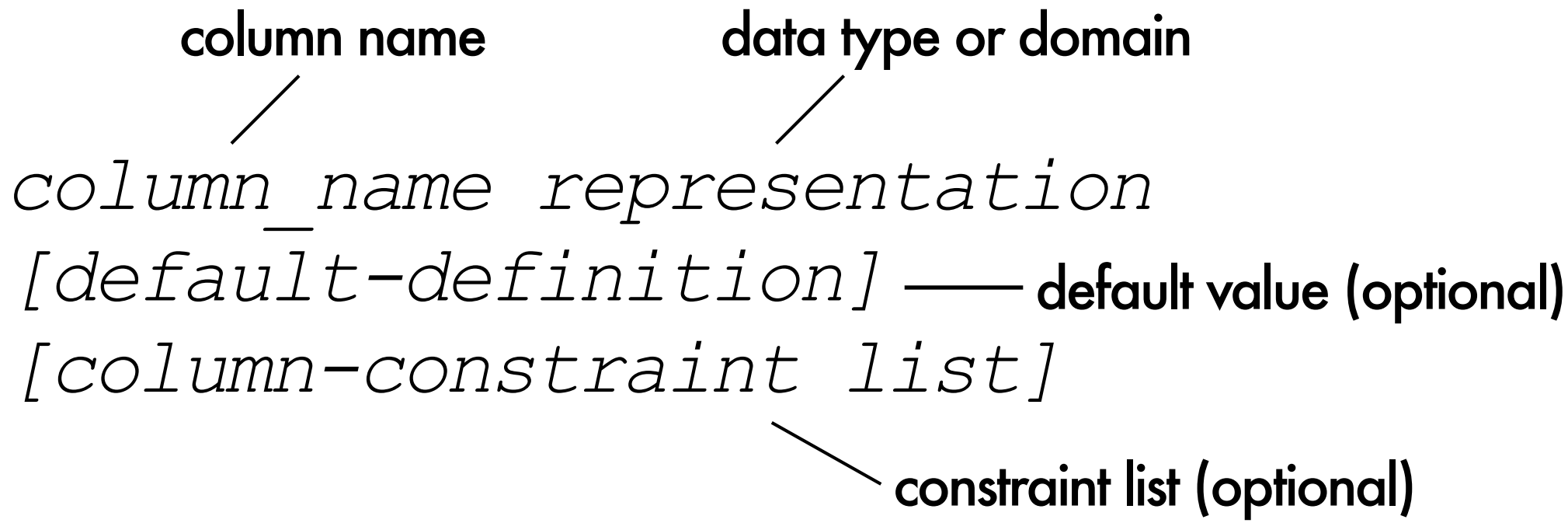
domain

schema

view

defining data in SQL

```
create table TABLE_NAME (comma
delimited list of table-
elements)
```



data types

a sampling of data types (check your book for others)

decimal (p,s) — total number of digits, number after the decimal
the number 123.45 is precision 5, scale 2

smallint

integer

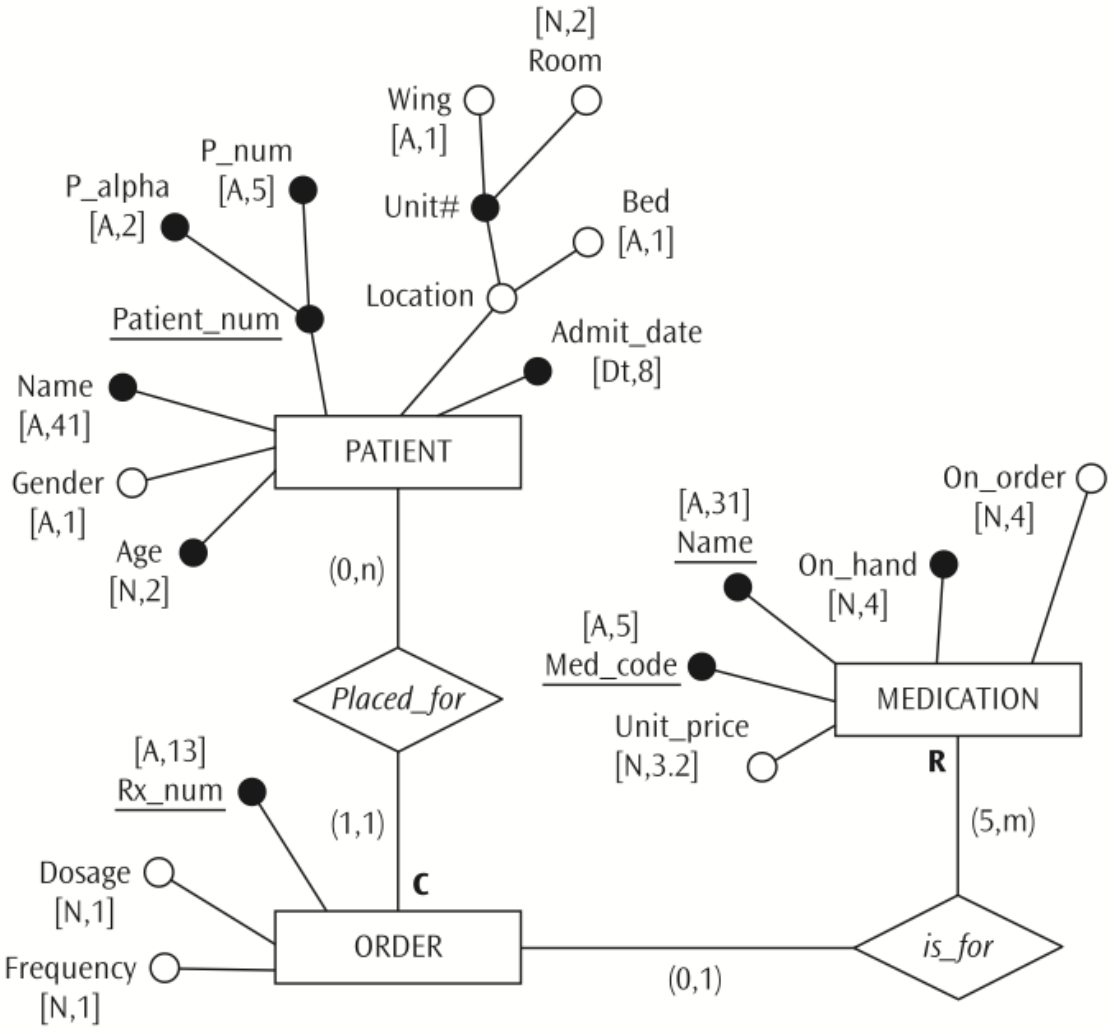
char

varchar

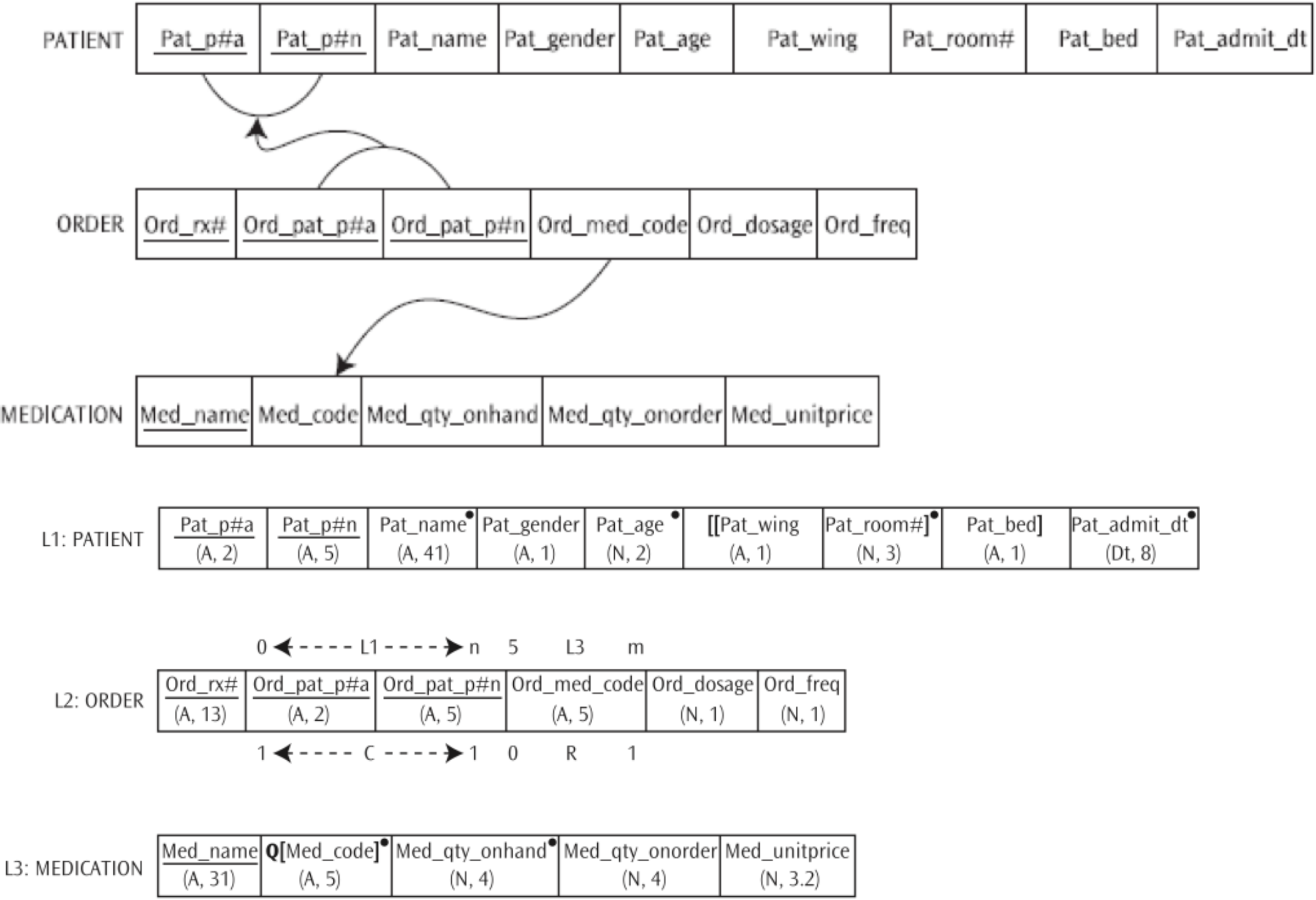
date

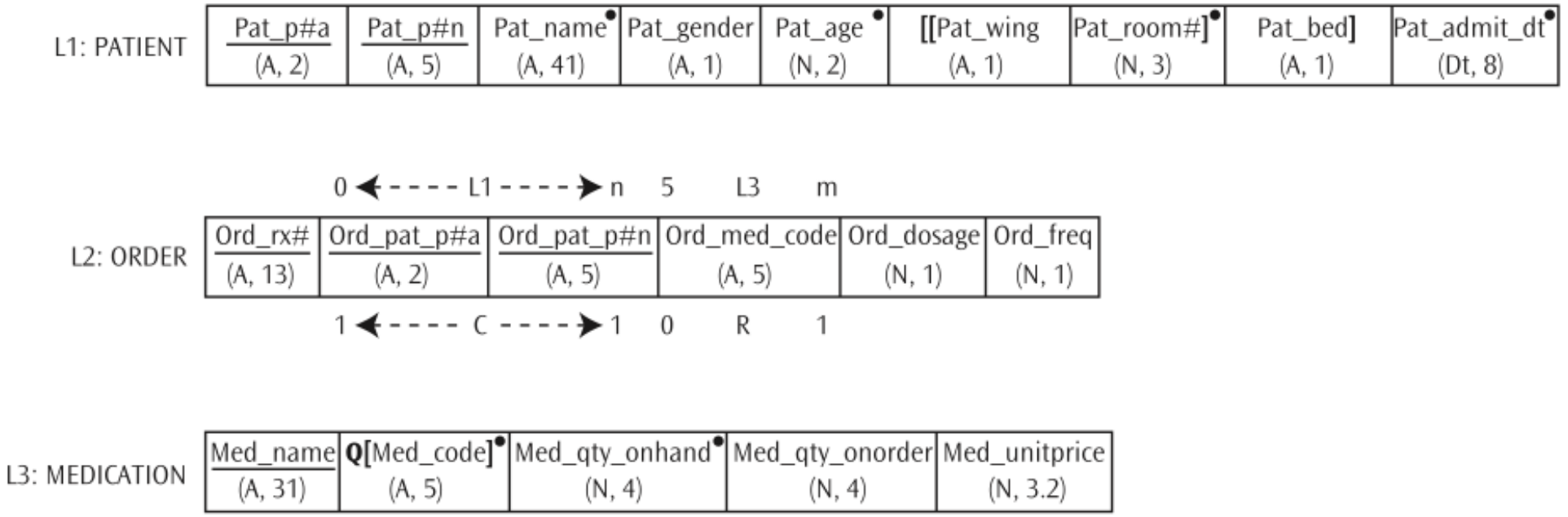
an example

page 445

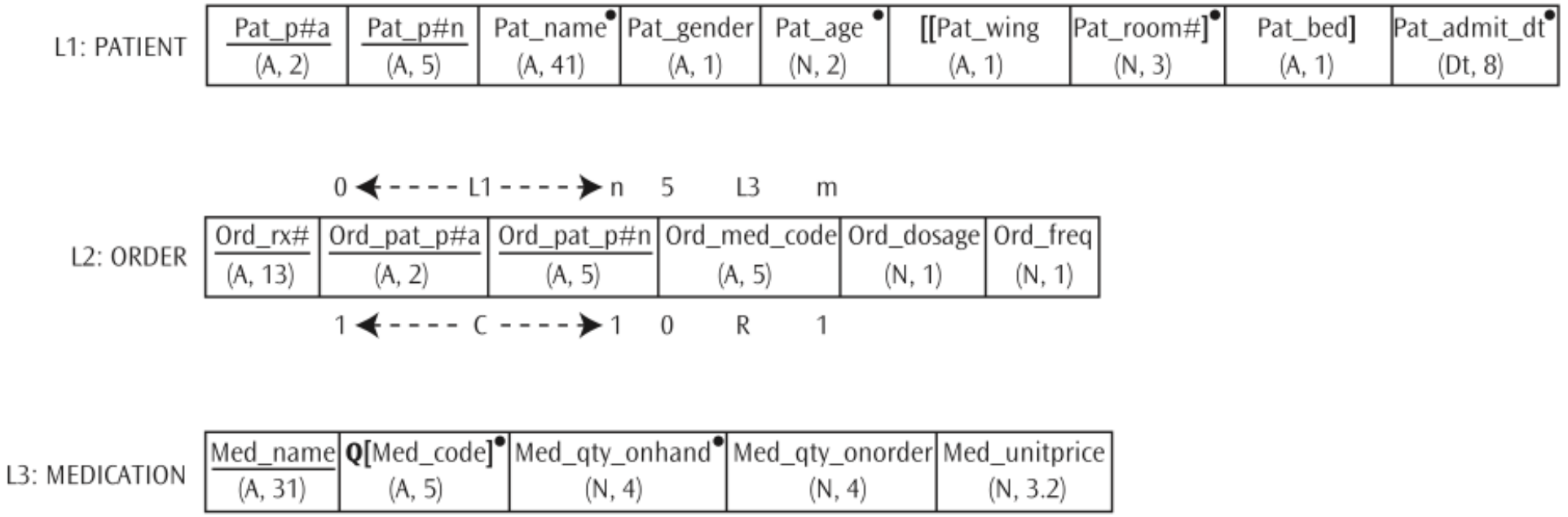


> Constraint	Gender	IN ('M', 'F')
> Constraint	Age	IN (1 through 90)
> Constraint	Bed	IN ('A', 'B')
> Constraint	Unit_price	< 4.50
> Constraint	(Qty_onhand + Qty_onorder)	IN (1000 through 3000)
> Constraint	Dosage	DEFAULT 2
> Constraint	Dosage	IN (1 through 3)
> Constraint	Frequency	DEFAULT 1
> Constraint	Frequency	IN (1 through 3)

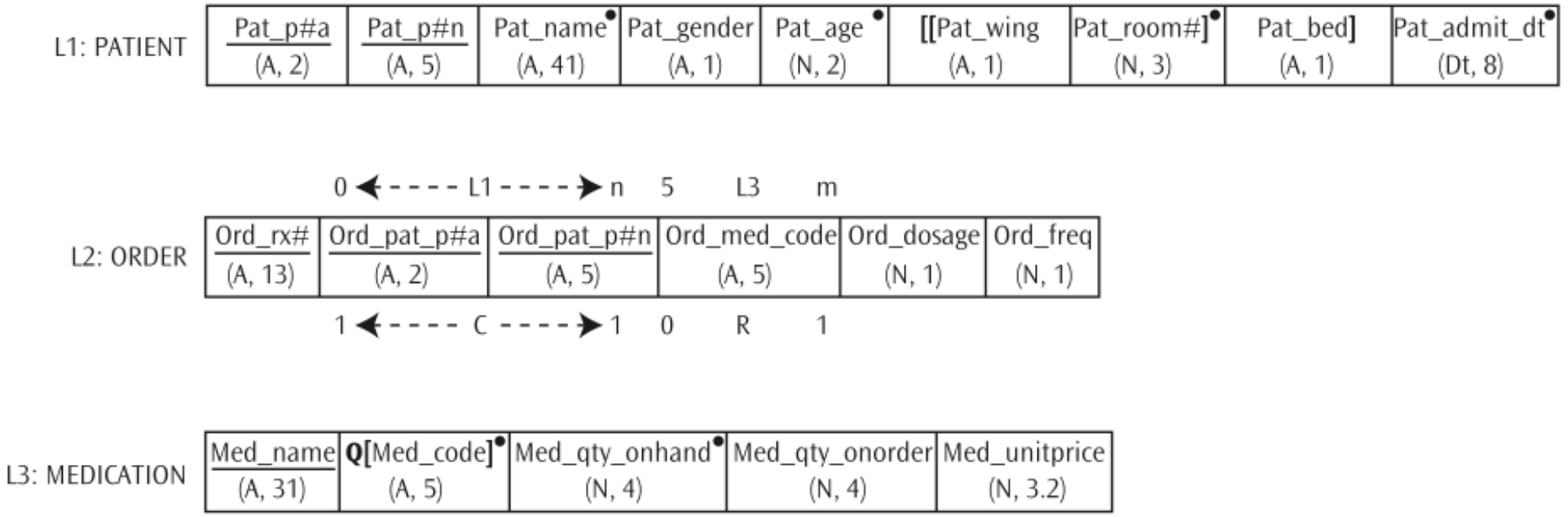

PATIENT
Pat_p#a
Pat_p#n
Pat_name
Pat_gender
Pat_age
Pat_wing
Pat_room#
Pat_bed
Pat_admit_dt
ORDER
Ord_rx#
Ord_pat_p#a
Ord_pat_p#n
Ord_med_code
Ord_dosage
Ord_freq
MEDICATION
Med_name
Med_code
Med_qty_onhand
Med_qty_onorder
Med_unitprice
L1: PATIENT
Pat_p#a (A, 2)
Pat_p#n (A, 5)
Pat_name• (A, 41)
Pat_gender (A, 1)
Pat_age • (N, 2)
[[Pat_wing (A, 1)
Pat_room#]• (N, 3)
Pat_bed] (A, 1)
Pat_admit_dt• (Dt, 8)
0 ◄ - - - - L1 - - - - ► n 5 L3 m
L2: ORDER
Ord_rx# (A, 13)
Ord_pat_p#a (A, 2)
Ord_pat_p#n (A, 5)
Ord_med_code (A, 5)
Ord_dosage (N, 1)
Ord_freq (N, 1)
1 ◄ - - - - C - - - - ► 1 0 R 1
L3: MEDICATION
Med_name (A, 31)
Q[Med_code]• (A, 5)
Med_qty_onhand• (N, 4)
Med_qty_onorder (N, 4)
Med_unitprice (N, 3.2)




```
create table patient
(Pat_p#a char(2), Pat_p#n char(5),
Pat_name varchar(41), Pat_gender char(1),
Pat_age smallint, Pat_admit_dt date,
Pat_wing char(1), Pat_room# integer
Pat_bed char(1));
```




```
create table medication
 (Med_code char(5), Med_name varchar(31),
Med_qty_onhand integer, Med_qty_onorder
integer, Med_unitprice decimal(3,2));
```



order is a reserved word

```
create table orders
(Ord_rx char(13), Ord_pat#a char(2),
Ord_pat#n char(5), Ord_med_code char(5),
Ord_dosage smallint, Ord_freq smallint);
```

constraints and default values

> Constraint	Gender	IN ('M', 'F')
> Constraint	Age	IN (1 through 90)
> Constraint	Bed	IN ('A', 'B')
> Constraint	Unit_price	< 4.50
> Constraint	(Qty_onhand + Qty_onorder)	IN (1000 through 3000)
> Constraint	Dosage	DEFAULT 2
> Constraint	Dosage	IN (1 through 3)
> Constraint	Frequency	DEFAULT 1
> Constraint	Frequency	IN (1 through 3)

constraints and default values

default *literal/niladic-function/NULL*

constraint *constraint_name constraint_definition*

examples:

PRIMARY KEY *(comma-delimited column list)*

CONSTRAINT pk_pat PRIMARY KEY (Pat_p#a, Pat_p#n)

> Constraint	Gender	IN ('M', 'F')
> Constraint	Age	IN (1 through 90)
> Constraint	Bed	IN ('A', 'B')
> Constraint	Unit_price	< 4.50
> Constraint	(Qty_onhand + Qty_onorder)	IN (1000 through 3000)
> Constraint	Dosage	DEFAULT 2
> Constraint	Dosage	IN (1 through 3)
> Constraint	Frequency	DEFAULT 1
> Constraint	Frequency	IN (1 through 3)

```
create table patient
(Pat_p#a char(2), Pat_p#n char(5),
Pat_name varchar(41), Pat_gender char(1),
Pat_age smallint, Pat_admit_dt date,
Pat_wing char(1), Pat_room# integer,
Pat_bed char(1),
constraint pk_pat primary key (Pat_p#a, Pat_p#n),
constraint chk_gender check (Pat_gender in ('M','F')),
constraint chk_age check (Pat_age between 1 and 90),
constraint chk_bed check (Pat_bed in ('A','B'));
```

> Constraint	Gender	IN ('M', 'F')
> Constraint	Age	IN (1 through 90)
> Constraint	Bed	IN ('A', 'B')
> Constraint	Unit_price	< 4.50
> Constraint	(Qty_onhand + Qty_onorder)	IN (1000 through 3000)
> Constraint	Dosage	DEFAULT 2
> Constraint	Dosage	IN (1 through 3)
> Constraint	Frequency	DEFAULT 1
> Constraint	Frequency	IN (1 through 3)

```
create table medication
(Med_code char(5),
Med_name varchar(31) constraint pk_med primary key,
Med_qty_onhand integer, Med_qty_onorder integer,
Med_unitprice decimal(3,2)
constraint chk_unitprice check (Med_unitprice < 4.5),
constraint chk_qty check ((Med_qty_onhand +
Med_qty_onorder) between 1000 and 3000));
```

> Constraint	Gender	IN ('M', 'F')
> Constraint	Age	IN (1 through 90)
> Constraint	Bed	IN ('A', 'B')
> Constraint	Unit_price	< 4.50
> Constraint	(Qty_onhand + Qty_onorder)	IN (1000 through 3000)
> Constraint	Dosage	DEFAULT 2
> Constraint	Dosage	IN (1 through 3)
> Constraint	Frequency	DEFAULT 1
> Constraint	Frequency	IN (1 through 3)

```
create table orders
(Ord_rx char(13) constraint pk_ord primary key,
Ord_pat_p#a char(2), Ord_pat_p#n char(5),
Ord_med_code char(5), constraint fk_med foreign key
(Ord_med_code) references medication(Med_code),
Ord_dosage smallint default 2 constraint chk_dosage
check (Ord_dosage between 1 and 3),
Ord_freq smallint default 1 constraint chk_freq check
(Ord_freq in (1, 2, 3)), constraint fk_pat foreign key
(Ord_pat_p#a, Ord_pat_p#n) references patient
(Pat_p#a, Pat_p#n));
```

still missing:

mandatory attributes (no null values): Pat_name, Pat_age, Pat_admit_dt, Med_code, Med_qty_onhand

alternate key: Med_code since Med_name has been chosen as the primary key of medication table

participation: Placed_for relationship - ORDER (total), PATIENT (partial)
Is_for relationship - MEDICATION (total), ORDER (partial)

deletion rules: is_for relationship - restrict
placed_for relationship - cascade

composite attributes:[Pat_wing, Pat_room], [Pat_wing, Pat_room, Pat_bed]

```
CREATE TABLE patient
(Pat_p#a char (2),
Pat_p#n char (5),
Pat_name varchar (41) constraint nn_patnm not null,
Pat_gender char (1),
Pat_age smallint constraint nn_patage not null,
Pat_admit_dt date constraint nn_patadmdt not null,
Pat_wing char (1),
Pat_room# integer,
Pat_bed char (1),
constraint pk_pat primary key (Pat_p#a, Pat_p#n),
constraint chk_gender check (Pat_gender IN ('M', 'F')),
constraint chk_age check (Pat_age between 1 and 90)),
constraint chk_bed check (Pat_bed IN (‘A', ‘B’)));
```

```
CREATE TABLE medication
(Med_code char(5) constraint nn_medcd not null
constraint unq_med unique,
Med_name varchar(31) constraint pk_med primary key,
Med_unitprice decimal(3,2) constraint chk_unitprice
check (Med_unitprice < 4.50),
Med_qty_onhand integer constraint nn_medqty not null,
Med_qty_onorder integer,
constraint chk_qty check ((Med_qty_onhand +
Med_qty_onorder) between 1000 and 3000))
```

```
CREATE TABLE orders
(Ord_rx char(13) constraint pk_ord primary key,
Ord_pat_p#a char(2) constraint nn_ord_pat_p#a not null,
Ord_pat_p#n char(5) constraint nn_ord_pat_p#n not null,
Ord_med_code char(5), constraint fk_med foreign key
(Ord_med_code) references medication(Med_code),
Ord_dosage smallint default 2 constraint chk_dosage check
(Ord_dosage between 1 and 3),
Ord_freq smallint default 1 constraint chk_freq check
(Ord_freq in (1, 2, 3)), constraint fk_pat foreign key
(Ord_pat_p#a, Ord_pat_p#n) references patient (Pat_p#a,
Pat_p#n) on delete cascade);
```

best practices

```
Pat_age smallint not null check
(Pat_age between 1 and 90)
```

OR

```
Pat_age smallint constraint nn_patage
not null, constraint chk_age
check (Pat_age between 1 and 90)
```

What happens when we want to drop the constraint?

```
ALTER TABLE patient drop constraint nn_patage;
```

altering table structure

`alter table` *`table_name action`*

add columns

add default value for column

drop column

add constraint

drop constraint

altering table structure

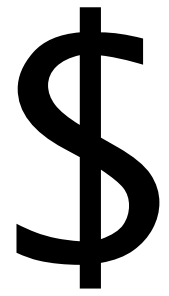
how do we add a patient phone number?

```
ALTER TABLE patient add Pat_phone# char (10);
```

how do we remove a column?

```
ALTER TABLE patient drop column Pat_phone#;
```

altering table structure



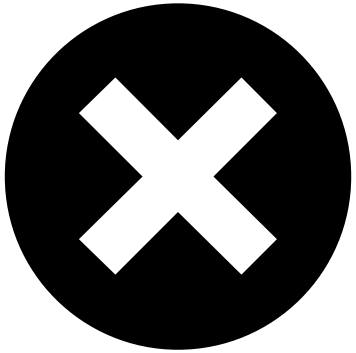
how do we set a default value of $3.00 for medication?

```
ALTER TABLE medication modify Med_unitprice
default 3.00;
```

how do we remove the default value?

```
ALTER TABLE medication modify Med_unitprice
default null;
```

dropping tables



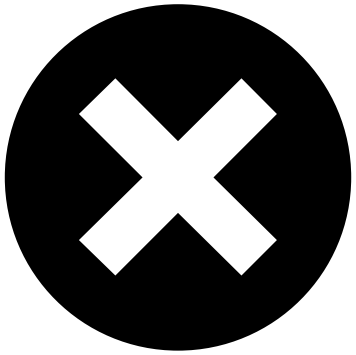
how do you remove a table?

```
DROP TABLE medication;
```

can add `cascade constraints` to statement to remove any foreign key constraints on referencing tables

```
DROP TABLE medication cascade constraints;
```

domains and schemas



read sections 10.1.2 and 10.1.3 in the textbook and understand the concepts, but we will not be covering the syntax in class

populating data in SQL

insert

delete

update

populating data in SQL

```
insert into TABLE_NAME values (comma separated
list of values);
```

```
insert into patient values ('DB77642', 'Davis,
Bill', 'Professor', 'M', 38, 'Blue Cross/Blue
Shield');
```

apostrophes are for literals

populating data in SQL

insert into *TABLE_NAME (comma separated list of columns)* values *(comma separated list of values);*

```
insert into prescription (Pre_p#, Pre_dosage,
    Pre_freq, Pre_pat_pid, Pre_med_code,
Pre_phy_ph#) values ('WAL4345',1,1,'DB77642',
                'VIB', '2314');
```

deleting data in SQL

```
delete from TABLE_NAME [where search-condition];
```

```
delete from patient where pat_name like '%Davis,
                                         Bill%';
```

```
delete from medication where med_code = 'TAG';
```

without the where clause, all rows would be deleted

updating data in SQL

```
update TABLE_NAME set column-name = expression
    [where search-condition];
```

```
update medication set med_unitprice = 5.00
    where med_code = 'TAG';
```

access control

create a user:

```
CREATE USER username IDENTIFIED BY apassword;
```

access control

grant *PRIVILEGES* on *OBJECT* to *USER [with grant option];*

grant

revoke

PRIVILEGES:

SELECT
INSERT
UPDATE
DELETE
REFERENCES
ALTER

access control

```
grant select, insert, delete, update on orders to
                        user_b;

    grant select, insert on patient to user_c;

    grant insert, delete on patient to user_d;

  grant all privileges on medication to public;

grant update (Pat_wing, Pat_room#, Pat_bed) on
                    patient to user_b;
```

refers to everyone

access control

```
revoke PRIVILEGES on OBJECT from USER;
```

```
revoke all privileges on orders from username;
```