

Feb 10, 17 15:51

all.sql

Page 1/11

```
===== Contacts.sql =====
-- Emily Van Laarhoven and Ainsley St.Clair

use evanlaar_db;

drop table if exists addresses;
drop table if exists phone_numbers;
drop table if exists contacts;

create table contacts (
  id integer primary key,
  nm varchar(25)
)
ENGINE = InnoDB;

create table addresses (
  id integer,
  kind varchar(25),
  city varchar(25),
  st char(2),
  INDEX(id),
  foreign key (id) references contacts(id) on delete restrict
)
ENGINE = InnoDB;

create table phone_numbers (
  id integer,
  kind varchar(25),
  phnum char(10),
  INDEX(id),
  foreign key (id) references contacts(id) on delete restrict
)
ENGINE = InnoDB;
```

Feb 10, 17 15:51

all.sql

Page 2/11

```
===== Q1-All-Addresses.sql =====  
-- Emily Van Laarhoven and Ainsley St.Clair  
  
-- returns a list of all addresses with contact names  
  
use evanlaar_db;  
  
select nm,kind,city,st  
from contacts inner join addresses using (id);
```

Feb 10, 17 15:51

all.sql

Page 3/11

```
===== Q2-All-Phones.sql =====  
-- Emily Van Laarhoven and Ainsley St.Clair  
  
-- Returns list of all phone numbers with names  
  
use evanlaar_db;  
  
select nm,kind,phnum  
from contacts inner join phone_numbers using (id);
```

Feb 10, 17 15:51

all.sql

Page 4/11

```
===== Q3-All-Home-Contacts.sql =====
-- Emily Van Laarhoven and Ainsley St.Clair

-- Return list of all people with home address and home phone

use evanlaar_db;

select nm,city,st,phnum
from phone_numbers, (select id,nm,kind,city,st
from contacts inner join addresses using (id)
where kind="home") as t1
where t1.id=phone_numbers.id and phone_numbers.kind="home";
```

Feb 10, 17 15:51

all.sql

Page 5/11

```
===== Q4-Phone-Count.sql =====  
-- Emily Van Laarhoven and Ainsley St.Clair  
  
-- return count of how many phones each person has  
  
use evanlaar_db;  
  
select contacts.nm, count(phone_numbers.id) as num_of_phones  
from phone_numbers inner join contacts using (id)  
group by id;
```

Feb 10, 17 15:51

all.sql

Page 6/11

```
===== Q5-New-Englanders.sql =====
-- Emily Van Laarhoven and Ainsley St.Clair

-- returns list of people who live in NE

use evanlaar_db;

select nm as NewEnglanders
from contacts inner join addresses using (id)
where addresses.kind="home" and
      (addresses.st="MA" or addresses.st="CT" or addresses.st="VT"
       or addresses.st="NH" or addresses.st="ME" or addresses.st="RI");
```

Feb 10, 17 15:51

all.sql

Page 7/11

```
===== Q6-Hub-People.sql =====  
-- Emily Van Laarhoven and Ainsley St.Clair  
  
-- returns list of people with area code 617  
  
use evanlaar_db;  
  
select nm  
from contacts inner join phone_numbers using (id)  
where phnum like "617%";
```

Feb 10, 17 15:51

all.sql

Page 8/11

```
===== Q7-BiCoastal-People.sql =====
-- Emily Van Laarhoven and Ainsley St.Clair

-- List of people who have address in MA and in CA

use evanlaar_db;

select nm
from contacts inner join
(select t1.id
  from (select id,city,st from addresses where st="MA") as t1,
       (select id,city,st from addresses where st="CA") as t2
 where t1.id=t2.id) as BiCoastalIds
using (id);
```


Feb 10, 17 15:51

all.sql

Page 9/11

```
===== Q8-Tokens.sql =====
-- Emily Van Laarhoven and Ainsley St.Clair

-- returns list of people who are only person living in their state

use evanlaar_db;

select nm
from (select nm, st, count(st) as num_inhabitants
from contacts inner join addresses using (id)
where kind="home"
group by st) as total_inhabitants
where total_inhabitants.num_inhabitants=1;
```

Feb 10, 17 15:51

all.sql

Page 10/11

```
===== Q-my-entries.sql =====  
-- Emily Van Laarhoven  
-- hwl, add 5 entries to the wmdb  
  
use wmdb;  
  
select person.nm,person.name,person.birthdate,person.addedby  
from person where addedby=1261;  
  
select movie.tt,movie.title,movie.release,movie.director,movie.addedby  
from movie where addedby=1261;
```

Feb 10, 17 15:51

all.sql

Page 11/11

```
===== Q-filmographies.sql =====
-- Emily Van Laarhoven

-- returns list of all actors with title and release date
-- of all movies they appear in

use wmdb;

select t1.name,movie.title,movie.release
from movie inner join
(select credit.tt,credit.nm,person.name
from credit inner join person using (nm)) as t1
on t1.tt=movie.tt
order by t1.name,movie.release,movie.title;
```