Q1:

```
CREATE TABLE IF NOT EXISTS city (
        id int NOT NULL,
        city varchar(255),
        state varchar(255),
        country varchar(255),
        PRIMARY KEY (id)
);
CREATE TABLE IF NOT EXISTS planes (
        plane_number int NOT NULL,
        model varchar(255),
        capacity int,
        create_year int,
        PRIMARY KEY (plane_number)
);
CREATE TABLE IF NOT EXISTS pilot (
        ssn int NOT NULL,
        home_city int,
        fullname varchar(255),
        day_of_birth int,
        month_of_birth int,
        year_of_birth int,
        salary int,
        PRIMARY KEY (ssn),
        FOREIGN KEY (home_city) REFERENCES city(id)
);
CREATE TABLE IF NOT EXISTS flight (
        time_takeoff int,
        time_landing int,
        flight_number int NOT NULL,
        captain_ssn int,
        plane_number int,
        takeoff_city int,
        landing_city int,
        PRIMARY KEY (flight_number),
        FOREIGN KEY (captain_ssn) REFERENCES pilot(ssn),
        FOREIGN KEY (plane_number) REFERENCES planes(plane_number),
        FOREIGN KEY (takeoff_city) REFERENCES city(id),
        FOREIGN KEY (landing_city) REFERENCES city(id)
);
```

Q2:

$$\pi_{city}(\sigma_{city.country='Germany'}(city))$$

Select city
From city
Where country = 'Germany'

Q3:

$$\pi_{pilot.fullname}((city) \bowtie_{city.id=pilot.home\_city \wedge city.country='Germany'} (pilot))$$

Select P.fullname
From city C, pilot P
Where C.id = P.home_city
And C.country = 'Germany'

Q4:

$$\pi_{pilot.fullname}(pilot) \bowtie_{pilot.ssn= flight.captain\_ssn}$$
$$(\sigma_{flight.takeoff_{city}=city.id \vee flight.landing_{city}=city.id}(flight \bowtie_{city.country='Germany'} city))$$

Select P.fullname

From pilot P

Where P.ssn in (Select F.captain_ssn

From flight F

Where

F.take_off_city in (Select C.id

From City C

Where C.country = 'Germany')

Or F.landing_city in (Select C.id

From City C

C.country = 'Germany'))

Q5:

$$\pi_{planes.model}(planes) \bowtie_{plane\_number = flight.plane\_number} (\sigma_{flight.takeoff_{city}=city.id \vee flight.landing_{city}=city.id}(flight \bowtie_{city.country='US'} city))$$

Select Distinct P.model

From planes P

Where P.plane_number in (Select F.plane_number

From flight F

Where

F.take_off_city in (Select C.id

From City C

Where C.country = 'US')

Or F.landing_city in (Select C.id

From City C

C.country = 'US'))

Q6:

$$\pi_{planes.model}(planes) \bowtie_{plane\_number = flight.plane\_number} ((\sigma_{flight.takeoff_{city}=city.id \vee flight.landing_{city}<>city.id}(flight \bowtie_{city.country='US'} city)) \cup (\sigma_{flight.takeoff_{city}<>=city.id \vee flight.landing_{city}==city.id}(flight \bowtie_{city.country='US'} city)))$$

Select Distinct P.model

From planes P

Where P.plane_number in (Select F.plane_number

From flight F

Where (F.take_off_city in (select C.id

From city C

Where C.country = 'US')

And F.landing_city NOT in (select C.id

From city C

Where C.country = 'US'))

Or

(F.take_off_city NOT in (select C.id

From city C

Where C.country = 'US')

And F.landing_city in (select C.id

From city C

Where C.country = 'US'))

Q7:

A,b,c,d

a,b,c

Q8:

1) 13
2) Find planes' information which take off from Pittsburgh
3)

$$\sigma_{plane.plane\_number=flight.plane\_number}(planes \times flight \bowtie_{\sigma_{city='pittsburgh' \wedge flight.takeoff-city=city.id}} city)$$

First do natural join, so we won' t join all the tuples, so it' s more effient.

4)

```
Select P.*

From city C, planes P

Where P.plane_number in (Select F.plane_number

                         From city C, flight F

                         Where C.city =  'Pittsburgh'

                         And C.id = F.take_off_city )
```