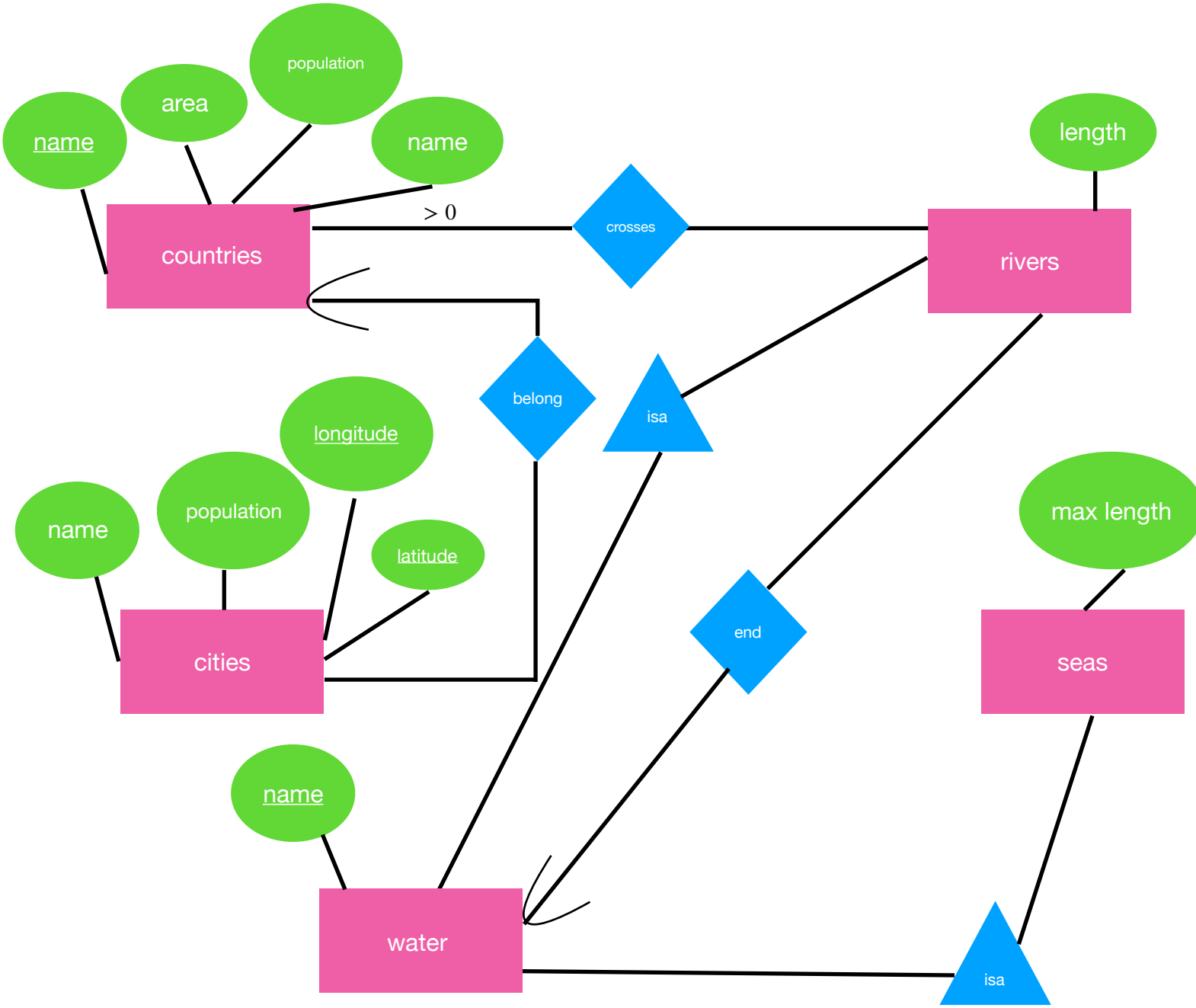
hw7
Part 1 Theory
1



2

```
CREATE TABLE InsuranceCo (
   name varchar(20) Primary key,
   phone INT,
);

CREATE TABLE Person (
   ssn INT Primary key,
   name varchar(20),
);


CREATE TABLE Driver (
   ssn INT Primary key references Person(ssn),
   driverID INT,
);

CREATE TABLE NonProfessionalDriver (
   ssn INT Primary key references Driver(ssn),
);

CREATE TABLE ProfessionalDriver (
   ssn INT Primary key references Driver(ssn),
   medicalHistory varchar(200),
);

CREATE TABLE Vehicle (
   licensePlate varchar(20) Primary key,
   year INT,
   name varchar(20) FOREIGN KEY references InsuranceCo(name),
   ssn INT FOREIGN KEY references Person(ssn),
   maxLiability REAL,
);

CREATE TABLE Car (
   licensePlate varchar(20) Primary key references Vehicle(licensePlate),
   make varchar(10),
);

CREATE TABLE Truck (
   licensePlate varchar(20) Primary key references Vehicle(licensePlate),
   capacity varchar(10),
   ssn INT FOREIGN KEY REFERENCES ProfessionalDriver(ssn),
);

CREATE TABLE Drives (
   licensePlate varchar(20) references Car(licensePlate),
   ssn INT references NonProfessionalDriver(ssn),
   PRIMARY KEY(ssn, licensePlate)
);
```

2.2 Which relation in your relational schema represents the relationship "insures" in the E/R diagram and why is that your representation?

Answer: my table vehicle represents the relationship “insures”, because insures is a N to 1 relationship, so I can use a foreign key in relation vehicle to save one table.

2.3 Compare the representation of the relationships "drives" and "operates" in your schema, and explain why they are different.

Answer: relationship “drives” is a N to N relationship, which means a non professional driver can drive as many cars as he wants, and a car can be driven by one or many non professional drivers. However, Operates is a N to 1 relationship, which means a truck can only be operated by at most one professional driver, and one professional drive can drive one or many trucks.

3. Consider the following two relational schemas and sets of functional dependencies

3.1 R(A,B,C,D,E) with functional dependencies D → B, CE → A

Answer:
Using BCNF Decomposition Algorithm on R(A,B,C,D,E)
X = D, X^+ = BD, X ≠ X^+ and X^+ ≠ [all attributes]
Y = B, Z = ABCDE - BD = ACE
R1 = BD, R2 = ACDE
decompose R2
Using BCNF Decomposition Algorithm on R(A,C,D,E)
X = CE, X^+ = ACE, X ≠ X^+ and X^+ ≠ [all attributes]
Y = A, Z = ACDE - ACE = D
R1 = ACE, R2 = DCE
So we have R1(B,D), R2(A,C,E), R3(D,C,E)

3.2 S(A,B,C,D,E) with functional dependencies A → E, BC → A, DE → B.

Answer:
Using BCNF Decomposition Algorithm on R(A,B,C,D,E)
X = A, X^+ = AE, X ≠ X^+ and X^+ ≠ [all attributes]
Y = E, Z = ABCDE - AE = BCD
R1 = AE, R2 = ABCD
decompose R2
Using BCNF Decomposition Algorithm on R(A,B,C,D)
X = BC, X^+ = ABC, X ≠ X^+ and X^+ ≠ [all attributes]
Y = A, Z = ABCD - ABC = D
R1 = ABC, R2 = BCD
So we have R1(AE), R2(ABC), R3(BCD)

4. A set of attributes X is called closed (with respect to a given set of functional dependencies) if X+=X. Consider a relation with schema R(A,B,C,D) and an unknown set of functional dependencies. For each closed attribute set below, give a set of functional dependencies that is consistent with it.

4.1 All sets of attributes are closed

A →A
B →B
C →C
D →D
All are trivial FDs

4.2 The only closed sets are {} and {A,B,C,D}.
A → B
B →C
C →D
D →A

4.3 The only closed sets are {}, {A,B}, and {A,B,C,D}.
A→B
B→A
C→ABC
D→BCD