A)

Product: ProductId QunatitySold QuantityUsed ProductDescription

Item: ItemNum QuantityUsed ItemDescription

Receipt: ReceiptNumber QuantitySold SalesDate

B)

Product: ProductId ProductDescription

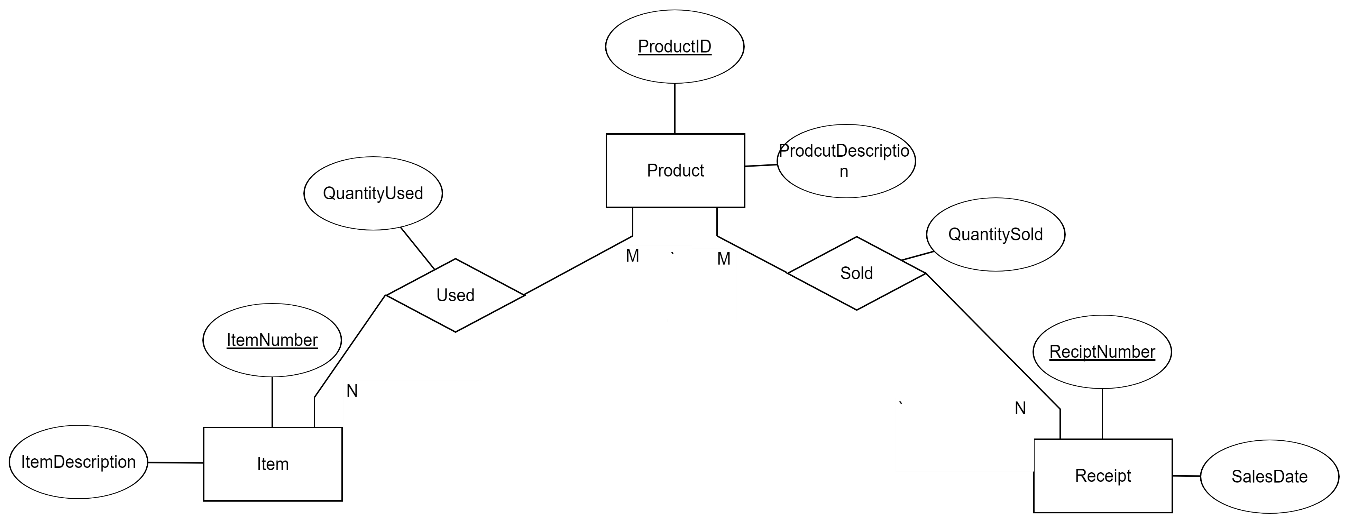
Item: ItemNum ItemDescription

Receipt: ReceiptNumber SalesDate

Used: ProductId ItemNum QuantityUsed

Sold: ReceiptNumber ItemNum QuantitySold

C)



D)

CREATE TABLE Product(

ProductId int

CONSTRAINT PK\_Prod PRIMARY KEY,

ProductDesc VARCHAR(200)

);

CREATE TABLE Item(

ItemNum int NOT NULL,

ItemDesc VARCHAR(200),

CONSTRAINT PK\_Item PRIMARY KEY(ItemNum)

);

CREATE TABLE Receipt(

ReceiptNum int NOT NULL,

SalesDate DATE,

CONSTRAINT PK\_Rec PRIMARY KEY(ReceiptNum)

);

CREATE TABLE Sold(

ReceiptNum int,

ItemNum int,

QuantitySold int,

CONSTRAINT PK\_Sold PRIMARY KEY(ReceiptNum, ItemNum),

CONSTRAINT CH\_Sold CHECK(QuantitySold >= 0)

);

CREATE TABLE Used(

ProductId int,

ItemNum int,

QuantityUsed int,

CONSTRAINT PK\_Used PRIMARY KEY(ProductID, ItemNum),

CONSTRAINT CH\_Used CHECK(QuantityUsed >= 0)

);

2)

A)

Physician: PhysDept PhyName PhyId DeptSupervisorId TreatCost TreatDesc TreatId

Patient: PatientID RoomPhone RoomNo RoomRate AmountOwing AdminDate PatientAddress PatientName HospitalStayDays

B)(amount owing is left out since it can be determined by HospitalStayDays\*RoomRate + TreatCost)

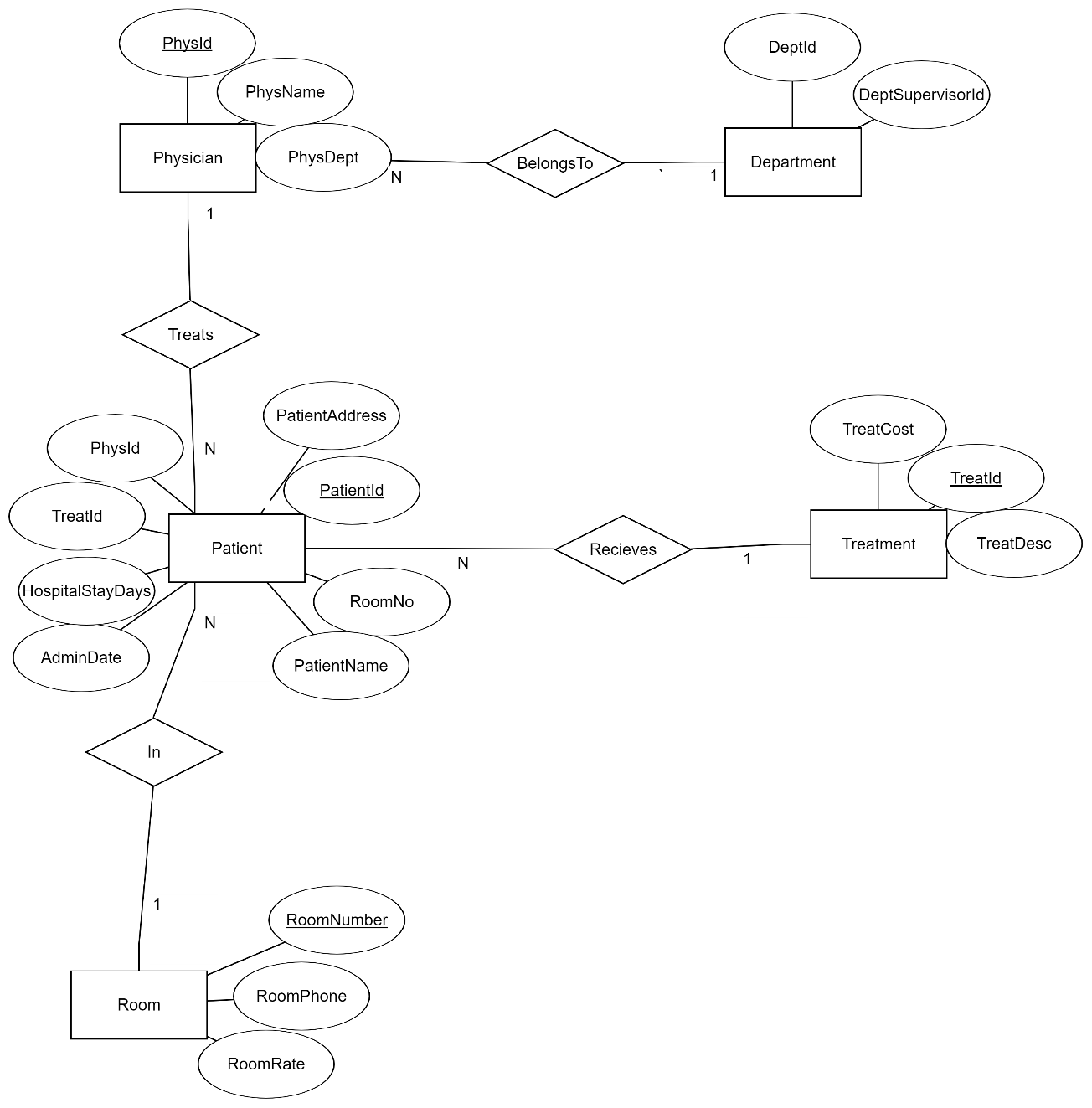
Physician: PhysID PhyName PhysDept\*

Room: RoomNo RoomPhone RoomRate

Treatment: TreatId TreatDesc TreatCost

Department: DeptId DeptSupervisorId\*

Patient: PatientID RoomNo\* PatientName PatientAdress AdminDate HospitalStayDays TreatId\* PhysId\*

C) 

D) CREATE TABLE Physician(

PhysID int,

PhyName VARCHAR(50) NOT NULL,

PhysDept int,

CONSTRAINT PK\_PhyId PRIMARY KEY(PhysID)

);

CREATE TABLE ROOM(

RoomNo int,

RoomPhone VARCHAR(8),

RoomRate DECIMAL(10,2),

CONSTRAINT PK\_RoomID PRIMARY KEY(RoomNo),

CONSTRAINT CH\_Room CHECK(RoomNo >= 100 AND RoomNo <= 999),

CONSTRAINT CH\_Cost CHECK(RoomRate >= 30 AND RoomRate <= 100)

);

CREATE TABLE TREATMENT(

TreatID int,

TreatDesc VARCHAR(200),

TreatCost DECIMAL(10,2),

CONSTRAINT PK\_TreatID PRIMARY KEY(TreatID),

CONSTRAINT CH\_TreatCost CHECK(TreatCost >= 50)

);

CREATE TABLE Department(

DeptID int,

DeptSupervisorID int,

CONSTRAINT PK\_DepID PRIMARY KEY(DeptID)

);

CREATE TABLE Patient(

PatientID int,

RoomNo int,

PatientName VARCHAR(50) NOT NULL,

PatientAddress VARCHAR(200) NOT NULL,

AdminDate DATE,

HospitalStayDays int,

TreatID int,

PhysID int,

CONSTRAINT CH\_StayDays CHECK(HospitalStayDays >= 0)

);

3)R3 becomes its own table since its part of an m-n relationship and E and F have composite keys that are compose of their own unique attributes and the primary key of their identifying entity. Since F is related to a weak entity its primary key is composed of the primary key of E which is weak and D which is strong.

A: A1 A2

B: B1 B2

C: C1 C2

D: D1 D5 D2 D3 D4

E: E1 D1 D5 R4 E1

F: F1 F2 E1 D1 D5 R5 F3 F4

R3: C1 D1 D5 R3

4) A-C and L-C become tables since they are m-n relationships and since Bank-branch is a weak relationship it’s primary key because a composite primary key composed of the primary key of its identifying entity and its own unique attribute.

Bank: Code Name Addr

Customer: SSN Name Addr Phone

Loan: LoanNo Amount Type

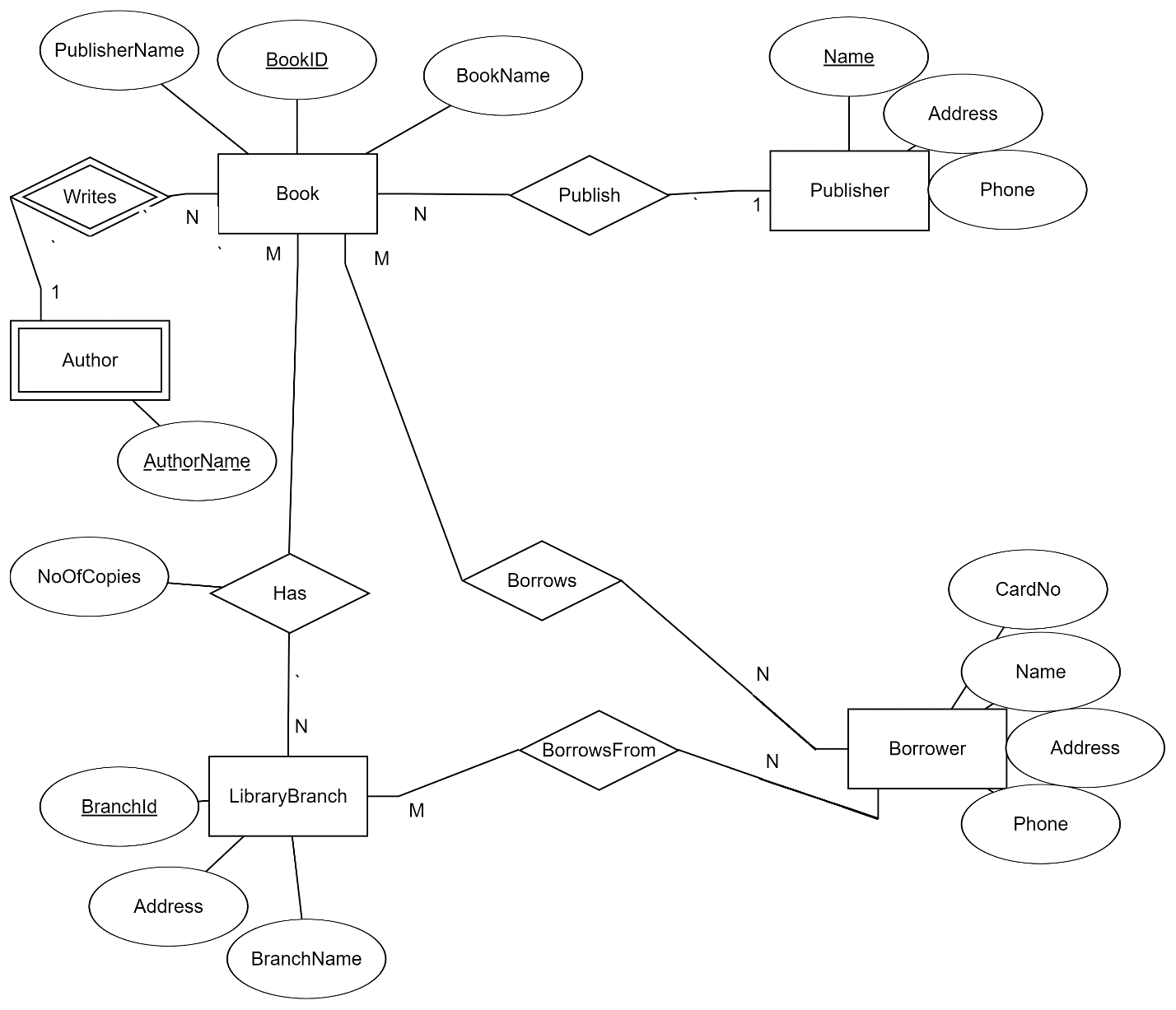
Account: AcctNo Balance Type

BankBranch: Code BranchNo Addr

A-C: AcctNo SSN

L-C: LoanNo SSN

5)



6)