Jaron Jon Javier

Professor Ahmad Hadaegh

CS443 - Database Management Systems

22 October, 2023

Assignment 1

1) Consider the following data. Arrows show the functional dependency. The arrows in this question indicated the determination of two attributes. For example, the arrow that goes ProductID to ProductDescription indicates that ProductID determines the ProductDescription. This in turn means that ProductId can be considered as the primary key for ProductDescription.

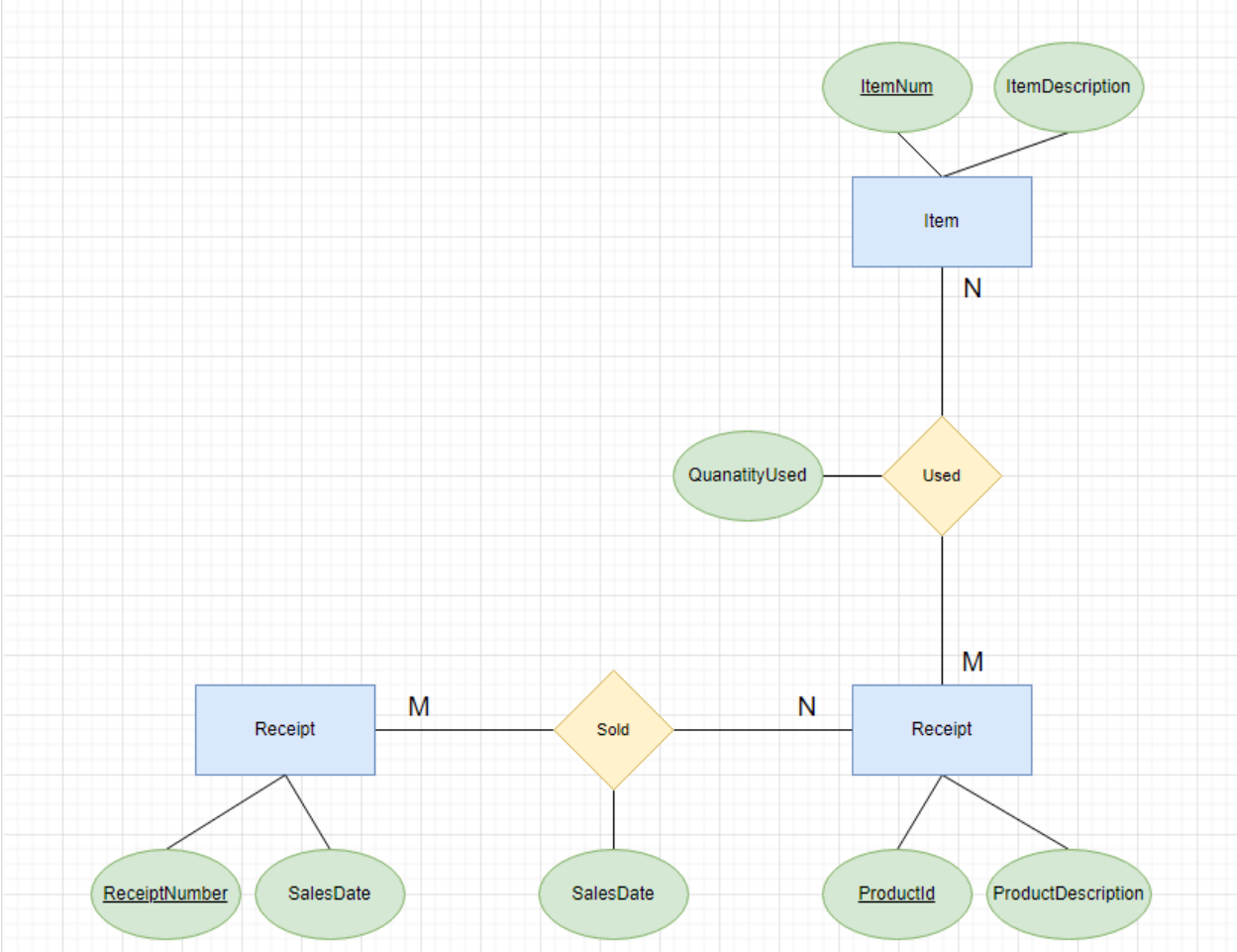
1a) Write the tables

Product(ProductId, ProductDescription)
Item(ItemNum, ItemDescription)
Receipt(ReceiptNumber, SalesDate)
Sold(ProductId*, ReceiptNumber*, QuantitySold)
Used(ItemNum*, ProductId*, QuantityUsed)

1b) Place the tables in 3rd normal form (if necessary)

The tables are already in 3rd normal form since there are no transitive or derived dependency attributes.

1c) Create ERD based on the normalized tables



1d) Write a script to create a database. Your script should create the tables and ensure that all constraints are set properly.

```
CREATE TABLE Product
(
  ProductID                    NUMBER,
  ProductDescription           VARCHAR2(200),
  CONSTRAINT Product_PK        PRIMARY KEY(ProductID)
);

CREATE TABLE Item
(
  ItemNum                      NUMBER,
  ItemDescription              VARCHAR2(200),
  CONSTRAINT Item_PK           PRIMARY KEY(ItemNum)
);
```

```
CREATE TABLE Receipt
(
  ReceiptNumber                 Number,
  SalesDate                     DATE,
  CONSTRAINT Receipt_PK         PRIMARY KEY(ReceiptNumber)
);

CREATE TABLE Used
(
  ProductID                     NUMBER,
  ItemNum                       NUMBER,
  QuantityUsed                  NUMBER,
  CONSTRAINT Used_PK            PRIMARY KEY(ProductID, ItemNum),
  CONSTRAINT Used_FK1           FOREIGN KEY(ProductID) REFERENCES
Product(ProductID),
  CONSTRAINT Used_FK2           FOREIGN KEY(ItemNum) REFERENCES
Item(ItemNum),
  CONSTRAINT QuantityUsed_CK    CHECK(QuantityUsed >= 0)
);

CREATE TABLE Sold
(
  ProductID                     NUMBER,
  ReceiptNumber                 NUMBER,
  QuantitySold                  NUMBER,
  CONSTRAINT Sold_PK            PRIMARY KEY(ProductID, ReceiptNumber),
  CONSTRAINT Sold_FK1           FOREIGN KEY(ProductID) REFERENCES
Product(ProductID),

  CONSTRAINT Sold_FK2           FOREIGN KEY(ReceiptNumber)
REFERENCES Receipt(ReceiptNumber),

  CONSTRAINT QuantitySold_CK    CHECK(QuantitySold >= 0)
);
```

2a) Change the ERD to tables

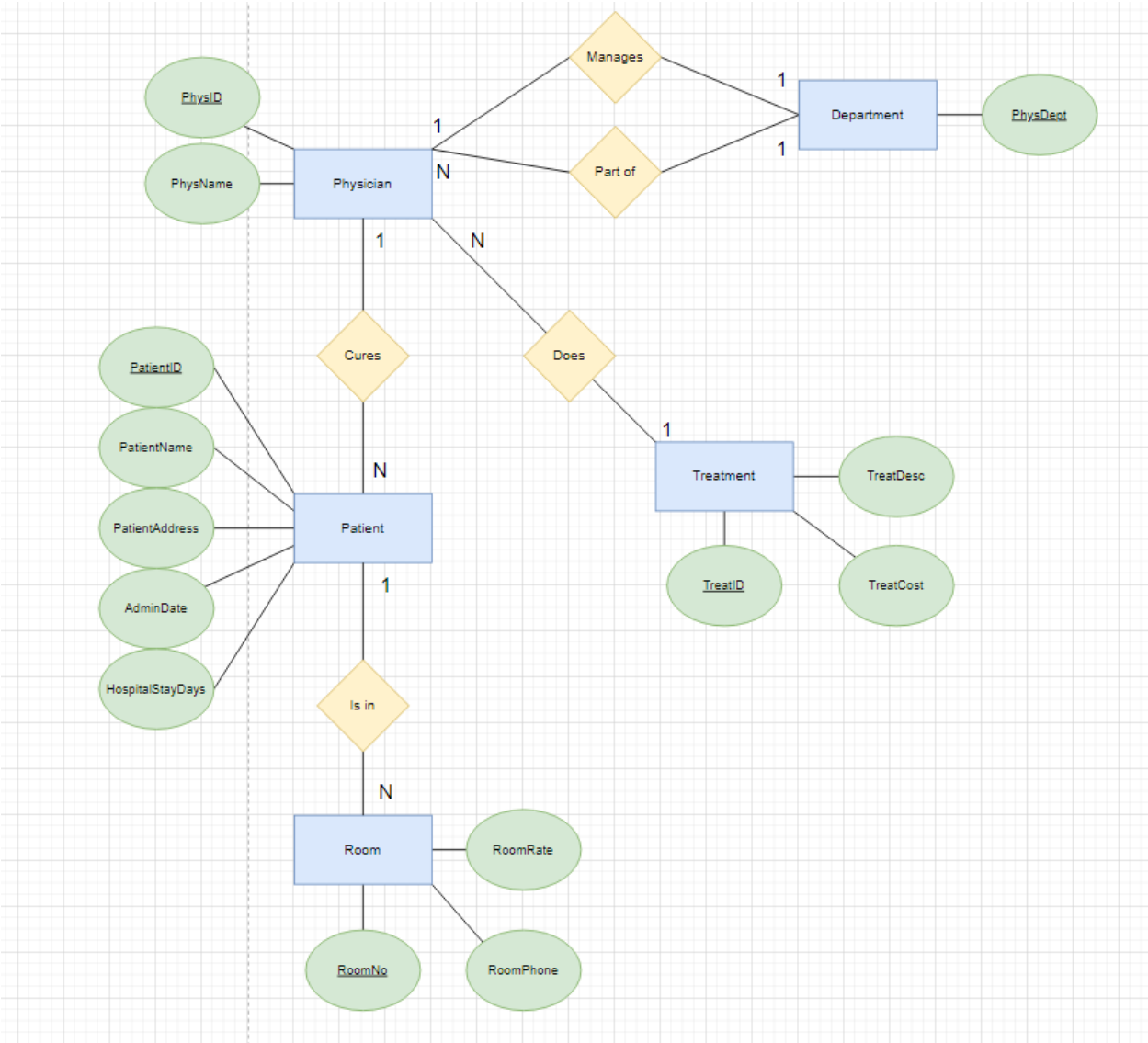
Physician(PhysID, PhyName, PhysDept, DeptSupervisorId, TreatCost, TreatDesc, TreatId)

Patient(PatientID, RoomPhone, RoomNo, RoomRate, AmountOwing, AdminDate, PatientAddress, PatientName, HospitalStayDays, PhysID*)

2b) Place the tables in 3rd normal form (if necessary)

Physician(PhysID, PhysName, PhysDept*, TreatID*)
Department(PhysDept, DeptSupervisorId*)
Patient(PatientID, PatientName, PatientAddress, AdminDate, HospitalStayDays, RoomNo*, PhysID*)
Room(RoomNo, RoomPhone, RoomRate)
Treatment(TreatId, TreatDesc, TreatCost, PhysID*)

2c) Revise the given ERD based on the normalized tables (if necessary)



2d) Write a script to create a database. Your script should create the tables and ensure that all constraints are set properly.

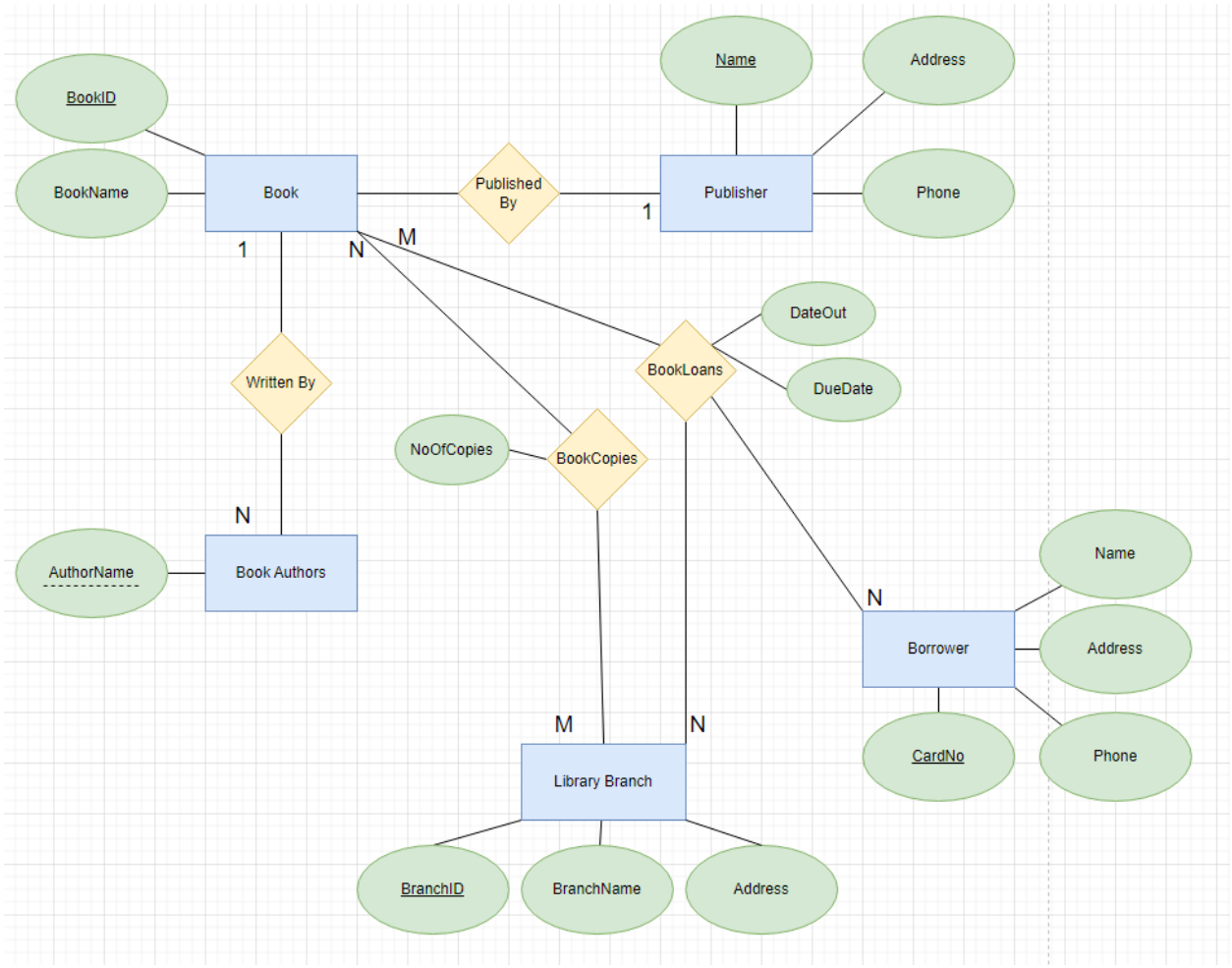
```
CREATE TABLE Department
(
  PhysDept                      NUMBER,
  DeptSupervisorID              NUMBER,
  CONSTRAINT Department_PK      PRIMARY KEY(PhysDept)
);

CREATE TABLE Treatment
(
  TreatID                       NUMBER,
  TreatCost                     NUMBER(10, 2),
  TreatDesc                     VARCHAR2(200),
  CONSTRAINT Treatment_PK       PRIMARY KEY(TreatID),
  CONSTRAINT TreatCost_CK       CHECK(TreatCost >= 50.00)
);

CREATE TABLE Room
(
  RoomNo                        NUMBER,
  RoomPhone                     VARCHAR(8),
  RoomRate                      NUMBER(10, 2),
  CONSTRAINT Room_PK            PRIMARY KEY(RoomNo),
  CONSTRAINT RoomRate_CK        CHECK(RoomRate >= 30.00 AND RoomRate
<= 100.00),
  CONSTRAINT RoomNo_CK          CHECK(RoomNo >= 100 AND RoomNo <=
999)
);
```

```
CREATE TABLE Physician
(
  PhysID                          NUMBER,
  PhysName                        VARCHAR2(50) CONSTRAINT
PhysName_Null NOT NULL,
  PhysDept                        NUMBER,
  TreatID                         NUMBER,
  CONSTRAINT Physician_PK         PRIMARY KEY(PhysID),
  CONSTRAINT Physician_FK1        FOREIGN KEY(PhysDept) REFERENCES
Department(PhysDept),
  CONSTRAINT Physician_FK2        FOREIGN KEY(TreatID) REFERENCES
Treatment(TreatID)
);

CREATE TABLE Patient
(
  PatientID                       NUMBER,
  PatientName                     VARCHAR2(50) CONSTRAINT
PatientName_Null NOT NULL,
  PatientAddress                  VARCHAR2(200)     CONSTRAINT
PatientAddress_Null NOT NULL,
  AdminDate                       DATE,
  HospitalStayDays                NUMBER,
  RoomNo                          NUMBER,
  PhysID                          NUMBER,
  CONSTRAINT Patient_PK           PRIMARY KEY(PatientID),
  CONSTRAINT Patient_FK1          FOREIGN KEY(RoomNo) REFERENCES
Room(RoomNo),
  CONSTRAINT Patient_FK2          FOREIGN KEY(PhysID) REFERENCES
Physician(PhysID),
  CONSTRAINT HospitalStayDays_CK  CHECK(HospitalStayDays >= 0)
);

ALTER TABLE DEPARTMENT
        ADD CONSTRAINT Department_FK1 FOREIGN KEY(DeptSupervisorID)
        REFERENCES Physician(PhysID);
```

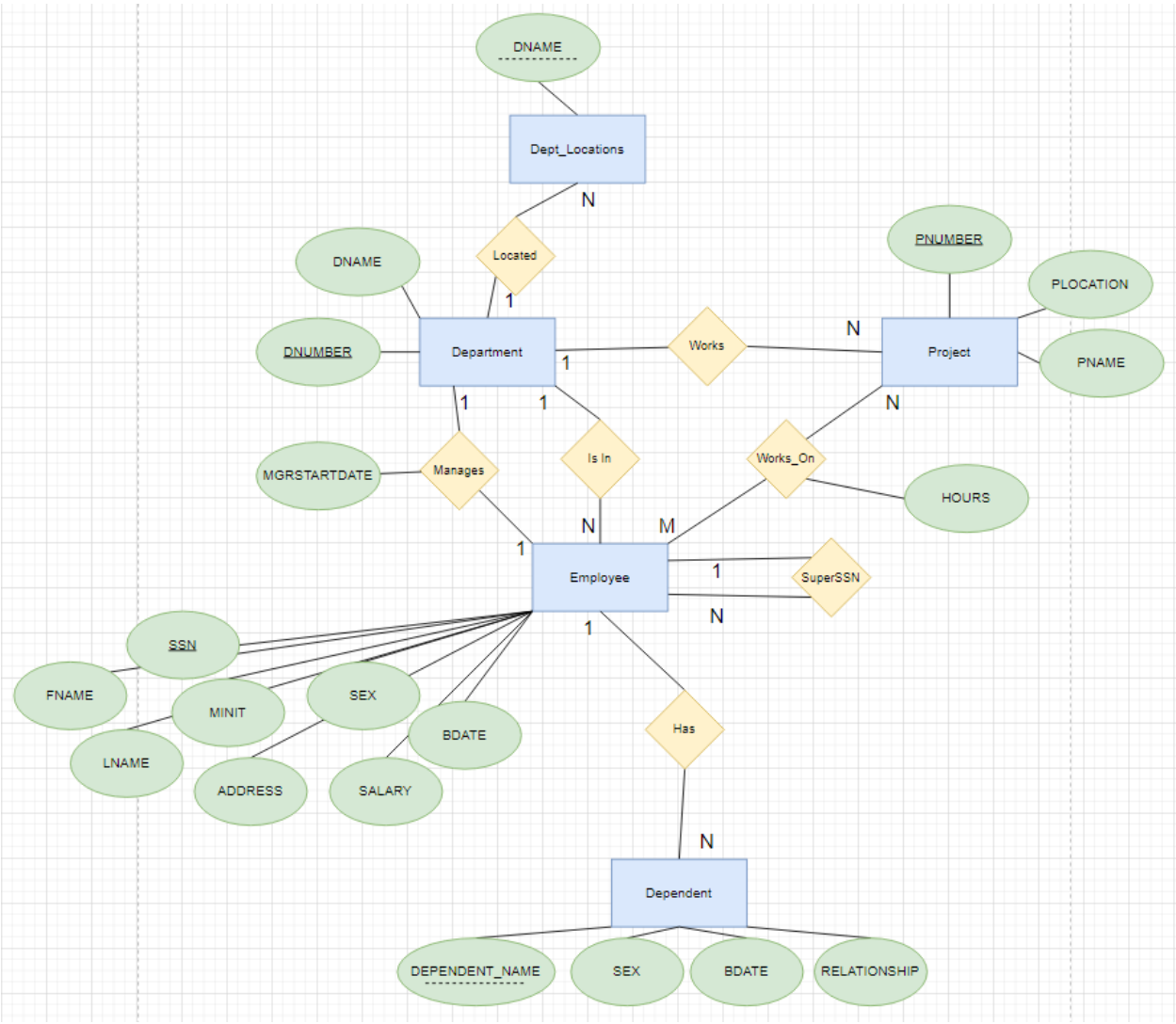
3) Create the tables related to the following ERD. Determine the primary Keys and the foreign keys of each table.

A(A1, A2)
B(B1, B2, A1*, C1*)
C(C1, C2)
D(D1, D5, D2, D3, D4)
E(E1, (D1, D5)*, E2, AttOfR4)
F(F1, F2, (E1, D1, D5)*, F3, F4)
R3(C1*, (D1, D5)*, AttOfR3)

4)



5)


DNAME
Dept_Locations
N
Located
1
DNAME
DNUMBER
Department
1
1
1
Works
N
1
PNUMBER
PLOCATION
Project
PNAME
N
MGRSTARTDATE
Manages
Is In
Works_On
HOURS
N
M
1
Employee
1
SuperSSN
N
1
SSN
FNAME
MINIT
LNAME
SEX
BDATE
ADDRESS
SALARY
Has
N
Dependent
DEPENDENT_NAME
SEX
BDATE
RELATIONSHIP