# CS 5200 Hands-on Exercise

## **[50pt] Database Implementation**

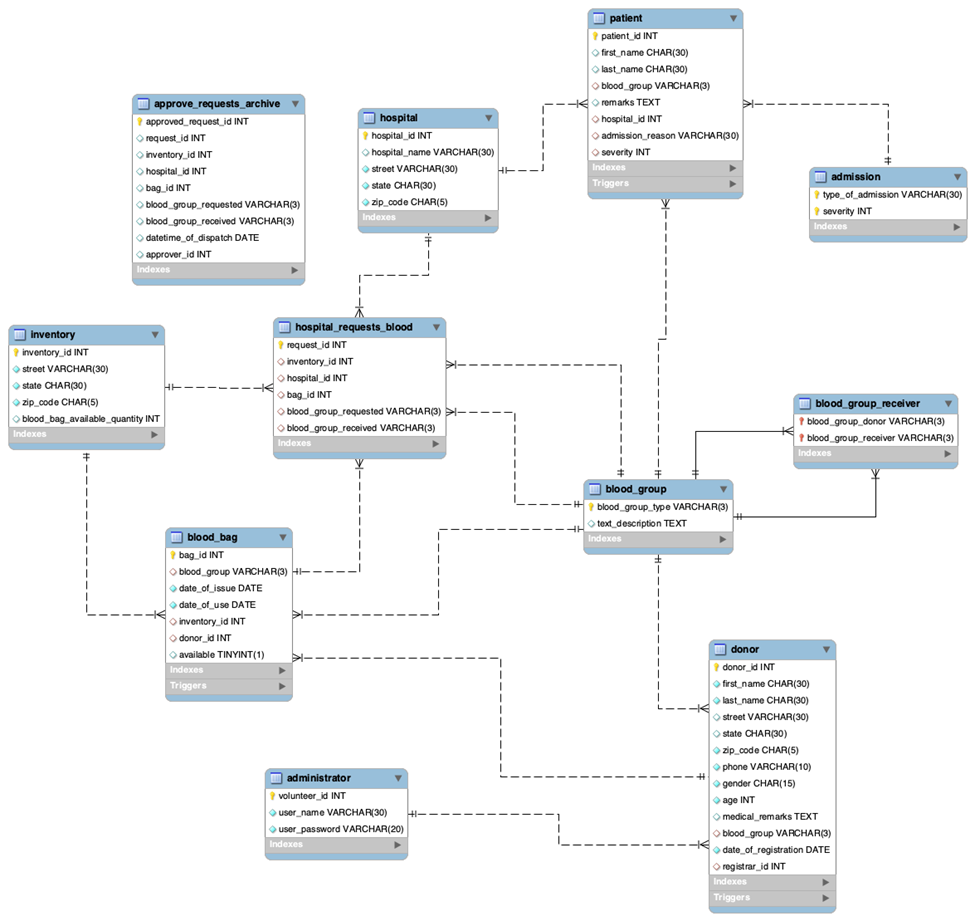


image-20231122194343977

**1) [25pt] SQL queries**

- For each table,

a) **[15pt]** Must present **screenshots of your SQL query** that creates a table for each table

* First, you need to define and present your own decision for enforcing the minimum cardinality for each relationship
* You must define and implement at least three data constraints.
* Your queries must consider all required data constraints and table constraints that are defined in the design.

b) **[10pt]** Must present **screenshots of your SQL query** that inserts data entries for each table

(Weak/Strong) Max: Min - Min: Max

**2) [25pt] Result of Implementation**

Please submit an **exported database**.

\* The minimum number of rows is **10**.



Table creating part:

-- SHOW ENGINE INNODB STATUS;
  
CREATE TABLE administrator (
  
 VolunteerID Int NOT NULL AUTO\_INCREMENT,
  
 UserName VARCHAR(30) NOT NULL,
  
 UserPassword VARCHAR(20) NOT NULL,
  
 CONSTRAINT PK\_ADMN PRIMARY KEY(VolunteerID),
  
 CONSTRAINT AK\_ADMN UNIQUE(UserName)
  
 )ENGINE=InnoDB AUTO\_INCREMENT = 1;
  
  
CREATE TABLE blood\_group (
  
 BloodGroupType VARCHAR(3) NOT NULL,
  
 TextDescription TEXT DEFAULT NULL,
  
 CONSTRAINT PK\_BG PRIMARY KEY(BloodGroupType),
  
 CONSTRAINT BG\_Domain CHECK (BloodGroupType IN ('A+', 'A-', 'B+', 'B-', 'O+', 'O-', 'AB+', 'AB-') )
  
 );
  
  
CREATE TABLE blood\_group\_receiver (
  
 BloodGroupDonor VARCHAR(3) NOT NULL,
  
 BloodGroupReceiver VARCHAR(3) NOT NULL,
  
 CONSTRAINT PK\_BGR PRIMARY KEY (BloodGroupDonor, BloodGroupReceiver),
  
 CONSTRAINT BG\_FK1 FOREIGN KEY (BloodGroupDonor) REFERENCES blood\_group(BloodGroupType)
  
 ON UPDATE NO ACTION
  
 ON DELETE NO ACTION,
  
 CONSTRAINT BG\_FK2 FOREIGN KEY (BloodGroupReceiver) REFERENCES blood\_group(BloodGroupType)
  
 ON UPDATE NO ACTION
  
 ON DELETE NO ACTION
  
 );
  
  
CREATE TABLE donor ( -- add trigger
  
 DonorID INT NOT NULL AUTO\_INCREMENT,
  
 FirstName CHAR(30) NOT NULL,
  
 LastName CHAR(30) NOT NULL,
  
 Street VARCHAR(30) DEFAULT NULL,
  
 State CHAR(30) DEFAULT NULL,
  
 ZipCode CHAR(5) NOT NULL,
  
 Phone VARCHAR(10) NOT NULL,
  
 Gender CHAR(15) NOT NULL,
  
 Age Int NOT NULL,
  
 MedicalRemarks TEXT DEFAULT NULL,
  
 BLoodGroup VARCHAR(3) NOT NULL,
  
 DateOfRegistration DATE NOT NULL,
  
 RegistrarID Int NOT NULL,
  
 CONSTRAINT PK\_DNR PRIMARY KEY (DonorID),
  
 CONSTRAINT AK\_DNR UNIQUE(Phone),
  
 CONSTRAINT Valid\_Age CHECK (Age > 0 AND Age < 100),
  
 CONSTRAINT Valid\_ZipCode CHECK (
  
 ZipCode LIKE '[0-9][0-9][0-9][0-9][0-9]'),
  
 CONSTRAINT Valid\_Phone CHECK (
  
 Phone LIKE '[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]'),
  
 CONSTRAINT BG\_FK FOREIGN KEY (BLoodGroup) REFERENCES blood\_group(BloodGroupType)
  
 ON UPDATE NO ACTION
  
 ON DELETE NO ACTION,
  
 CONSTRAINT ADMN\_FK FOREIGN KEY (RegistrarID) REFERENCES administrator(VolunteerID)
  
 ON UPDATE NO ACTION
  
 ON DELETE NO ACTION
  
)ENGINE=InnoDB AUTO\_INCREMENT = 11;
  
  
CREATE TABLE inventory (
  
 InventoryID INT NOT NULL AUTO\_INCREMENT,
  
 Street VARCHAR(30) NOT NULL,
  
 State CHAR(30) NOT NULL,
  
 ZipCode CHAR(5) NOT NULL,
  
 BloodBagQtn Int NULL,
  
 CONSTRAINT PK\_INVNTRY PRIMARY KEY(InventoryID),
  
 CONSTRAINT Valid\_ZipCode CHECK (ZipCode LIKE '[0-9][0-9][0-9][0-9][0-9]'),
  
 CONSTRAINT Valid\_BloodBagQtn CHECK (
  
 BloodBagQtn > -1)
  
)ENGINE=InnoDB AUTO\_INCREMENT = 31;
  
  
CREATE TABLE blood\_bag ( -- add trigger
  
 BagID INT NOT NULL AUTO\_INCREMENT,
  
 BLoodGroup VARCHAR(3) NOT NULL,
  
 DateOfIssue DATE NOT NULL,
  
 DateOfuse DATE NOT NULL,
  
 InventoryID INT NOT NULL,
  
 DonorID INT NOT NULL,
  
 Available TINYINT(1) NOT NULL DEFAULT 1,
  
 CONSTRAINT PK\_BB PRIMARY KEY(BagID),
  
 CONSTRAINT FK1\_BB FOREIGN KEY (BLoodGroup) REFERENCES blood\_group(BloodGroupType)
  
 ON UPDATE NO ACTION
  
 ON DELETE NO ACTION,
  
 CONSTRAINT FK2\_BB FOREIGN KEY (InventoryID) REFERENCES inventory(InventoryID)
  
 ON UPDATE NO ACTION
  
 ON DELETE NO ACTION,
  
 CONSTRAINT FK3\_BB FOREIGN KEY (DonorID) REFERENCES donor(DonorID)
  
 ON UPDATE NO ACTION
  
 ON DELETE NO ACTION,
  
 CONSTRAINT Date\_Check CHECK (DateOfuse > DateOfIssue),
  
 CONSTRAINT Valid\_AVL CHECK (Available = 1 OR Available = 0)
  
)ENGINE=InnoDB AUTO\_INCREMENT = 51;
  
  
CREATE TABLE hospital (
  
 HospitalID INT NOT NULL AUTO\_INCREMENT,
  
 HospitalName VARCHAR(30) NULL,
  
 Street VARCHAR(30) NOT NULL,
  
 State CHAR(30) NOT NULL,
  
 ZipCode CHAR(5) NOT NULL,
  
 CONSTRAINT PK\_HSPTL PRIMARY KEY(HospitalID),
  
 CONSTRAINT AK\_HSPTL UNIQUE(HospitalName),
  
 CONSTRAINT Valid\_ZipCode CHECK (ZipCode LIKE '[0-9][0-9][0-9][0-9][0-9]')
  
)ENGINE=InnoDB AUTO\_INCREMENT = 71;
  
  
CREATE TABLE hospital\_requests\_blood (
  
 RequestID INT NOT NULL AUTO\_INCREMENT,
  
 InventoryID INT NOT NULL,
  
 HospitalID INT NOT NULL,
  
 BagID INT NOT NULL,
  
 BLoodGroupReq VARCHAR(3) NOT NULL,
  
 BLoodGroupRec VARCHAR(3) NOT NULL,
  
 CONSTRAINT PK\_HRB PRIMARY KEY(RequestID),
  
 CONSTRAINT Req\_BG\_FK1 FOREIGN KEY (BLoodGroupReq) REFERENCES blood\_group(BloodGroupType)
  
 ON UPDATE NO ACTION
  
 ON DELETE NO ACTION,
  
 CONSTRAINT Req\_BG\_FK2 FOREIGN KEY (BLoodGroupRec) REFERENCES blood\_group(BloodGroupType)
  
 ON UPDATE NO ACTION
  
 ON DELETE NO ACTION,
  
 CONSTRAINT INVNTRY\_FK1 FOREIGN KEY (InventoryID) REFERENCES inventory(InventoryID)
  
 ON UPDATE NO ACTION
  
 ON DELETE NO ACTION,
  
 CONSTRAINT HSPTL\_FK1 FOREIGN KEY (HospitalID) REFERENCES hospital(HospitalID)
  
 ON UPDATE NO ACTION
  
 ON DELETE NO ACTION,
  
 CONSTRAINT BB\_FK1 FOREIGN KEY (BagID) REFERENCES blood\_bag(BagID)
  
 ON UPDATE NO ACTION
  
 ON DELETE NO ACTION
  
)ENGINE=InnoDB AUTO\_INCREMENT = 121;
  
  
CREATE TABLE admission (
  
 TypeOfAdmission VARCHAR(30) NOT NULL,
  
 Serverity INT NOT NULL,
  
 CONSTRAINT PK\_ADM PRIMARY KEY (TypeOfAdmission, Serverity),
  
 CONSTRAINT Valid\_Serverity CHECK (Serverity >= 0 AND Serverity < 7) -- AIS
  
)ENGINE=InnoDB;
  
  
CREATE TABLE patient ( -- add trigger
  
 PatientID INT NOT NULL AUTO\_INCREMENT,
  
 FirstName CHAR(30) NULL,
  
 LastName CHAR(30) NULL,
  
 BLoodGroup VARCHAR(3) NOT NULL,
  
 Remarks TEXT DEFAULT NULL,
  
 HospitalID Int NOT NULL,
  
 AdmissionReason VARCHAR(30) NOT NULL,
  
 Serverity Int NOT NULL,
  
 CONSTRAINT PK\_PTNT PRIMARY KEY(PatientID),
  
 CONSTRAINT Ptnt\_BG\_FK1 FOREIGN KEY (BLoodGroup) REFERENCES blood\_group(BloodGroupType)
  
 ON UPDATE NO ACTION
  
 ON DELETE NO ACTION,
  
 CONSTRAINT ADM\_FK1 FOREIGN KEY (AdmissionReason) REFERENCES admission(TypeOfAdmission)
  
 ON UPDATE CASCADE
  
 ON DELETE CASCADE,
  
 CONSTRAINT Ptnt\_HSPTL\_FK1 FOREIGN KEY (HospitalID) REFERENCES hospital(HospitalID)
  
 ON UPDATE NO ACTION
  
 ON DELETE NO ACTION,
  
 CONSTRAINT ADM\_FK2 FOREIGN KEY (Serverity) REFERENCES admission(Serverity)
  
 ON UPDATE CASCADE
  
 ON DELETE CASCADE
  
)ENGINE=InnoDB AUTO\_INCREMENT = 91;
  
  
CREATE TABLE approve\_requests\_archive (
  
 ApprovedRequestID INT NOT NULL AUTO\_INCREMENT,
  
 RequestID INT NULL,
  
 InventoryID INT NULL,
  
 HospitalID INT NULL,
  
 BagID INT NULL,
  
 BLoodGroupReq VARCHAR(3) NULL,
  
 BLoodGroupRec VARCHAR(3) NULL,
  
 DatetimeOfDispatch Date NULL,
  
 ApproverID INT NULL,
  
 CONSTRAINT PK\_ARA PRIMARY KEY(ApprovedRequestID)
  
)ENGINE=InnoDB AUTO\_INCREMENT = 151;

Trigger Creating part:

TODO:

* finish the trigger

DELIMITER //  
CREATE TRIGGER donor\_initail\_value  
BEFORE INSERT ON donor  
FOR EACH ROW  
-- I am thinking if a donor come with the bloodtype not in the bloodgroup shall I also create a new entry in bloodgroup, or I shall think all possible bloodgroups already stores in the bloodgroup table?  
BEGIN  
 DECLARE VarVolunteerID INT;  
 DECLARE Flag1 INT DEFAULT 0;  
 DECLARE Flag2 INT DEFAULT 0;  
 DECLARE VarBloodGroupType VARCHAR(3);  
 SET VarVolunteerID = NEW.VolunteerID;  
 SET VarBloodGroupType = NEW.BloodGroupType;  
   
 SELECT COUNT(VolunteerID) INTO FLag1  
 FROM administrator  
 WHERE VolunteerID = VarVolunteerID  
   
 SELECT COUNT(VarBloodGroupType) INTO FLag2  
 FROM blood\_group  
 WHERE BloodGroupType = VarVolunteerID  
   
 IF (FLag1 = 0 OR FLAG2 = 0) THEN  
 SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = 'Admin or blood group invalid';   
 END IF;  
   
  
END  
  
  
//  
DELIMITER ;  
  
DELIMITER ;  
  
DELIMITER //  
CREATE TRIGGER blood\_bag\_initail\_value  
BEFORE INSERT ON blood\_bag  
FOR EACH ROW  
  
BEGIN  
 DECLARE VarDonorID INT;  
 DECLARE VarBloodGroupType VARCHAR(3),  
 DECLARE VarInventoryID INT;  
 DECLARE Flag INT;  
   
 SET VarDonorID = NEW.DonorID;  
 SET VarInventoryID = NEW.InventoryID;  
 SET VarBloodGroupType = NEW.BloodGroupType;  
   
   
 SELECT SUM(COUNT(DonorID), COUNT(BloodGroupType), COUNT(InventoryID)) INTO Flag  
 FROM donor, blood\_group, inventory  
 WHERE DonorID = VarDonorID AND VarBloodGroupType = BloodGroupType AND VarInventoryID = InventoryID;  
   
 IF (FLag < 3) THEN  
 SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = 'donor or blood group or inventory invalid';   
 END IF;  
  
END  
  
  
//  
DELIMITER ;  
  
DELIMITER //  
CREATE TRIGGER patient\_initail\_value  
BEFORE INSERT ON patient  
FOR EACH ROW  
  
BEGIN   
 DECLARE VarHospitalID INT;  
 DECLARE VarBloodGroupType VARCHAR(3),  
 DECLARE VarSeverity INT;  
 DECLARE VarReason VARCHAR(30);  
 DECLARE Flag INT;  
   
 SET VarHospitalID = NEW.HospitalID;  
 SET VarReason = NEW.TypeOfAdmission;  
 SET VarSeverity = NEW.Severity;  
 SET VarBloodGroupType = NEW.BloodGroupType;  
   
   
 SELECT SUM(COUNT(HospitalID), COUNT(BloodGroupType), COUNT(TypeOfAdmission), COUNT(Severity)) INTO Flag  
 FROM hospital, blood\_group, admission  
 WHERE HospitalID = VarHospitalID AND VarBloodGroupType = BloodGroupType AND TypeOfAdmission = VarReason, Severity = VarSeverity;  
   
 IF (FLag < 4) THEN  
 SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = 'donor or blood group or inventory invalid';   
 END IF;  
  
END  
  
  
//  
DELIMITER ;