I am building a database for federal elections in Australia. Here are my tables:

CREATE TABLE TESTelectionMaster (

    electionSerialNo INTEGER PRIMARY KEY,

    electionDate date,

    electionType VARCHAR(50),

    totalNumDivisions INTEGER,

    totalRegVoters INTEGER,

    lastDateToVoterRegister DATE

    lastDateCandidateNominate DATE,

    lastDateToDeclareResult DATE

);

CREATE TABLE TESTelectoralDivision (

    divisionName VARCHAR(50) PRIMARY KEY,

    totalRegVoters INTEGER,

    currMember VARCHAR(50),

);

CREATE TABLE TESTelectoralDivisionHistory (

    divisionName VARCHAR(50),

    electionSerialNo Integer,

    historicRegVoters INTEGER,

    PRIMARY KEY (divisionName, electionSerialNo)

    );

CREATE TABLE TESTelectionEvent (

    electionEventID VARCHAR,

    totalVoters INTEGER,

    votesCast INTEGER,

    votesReject INTEGER,

    votesValid INTEGER,

    electionSerialNo INTEGER,

    divisionName VARCHAR(50),

    prefWinnerCandidateID VARCHAR(20),

    winnerTally Integer,

    prefLoserCandidateID VARCHAR(20),

    loserTally INTEGER,

    PRIMARY KEY (electionEventID)

);

CREATE TABLE TESTcandidateList (

    candidateID VARCHAR(20),

    candidateName VARCHAR (50),

    contactAddress VARCHAR (50),

    contactPhone INTEGER,

    contactMobile INTEGER,

    contactEmail VARCHAR(20),

    partyCode VARCHAR(60),

    PRIMARY KEY (candidateID)

);

CREATE TABLE TESTcontests (

    electionEventID VARCHAR,

    candidateID VARCHAR (20),

    PRIMARY KEY (electionEventID, candidateID)

);

CREATE TABLE TESTpoliticalParty (

    partyCode VARCHAR(60) PRIMARY KEY,

    partyName VARCHAR(100),

    partyLogo TEXT,

    postalAddress VARCHAR(50),

    partySecretary VARCHAR(50),

    contactPersonName VARCHAR(50),

    contactPersonPhone VARCHAR(20),

    contactPersonMobile VARCHAR(20),

    contactPersonEmail VARCHAR(50)

);

CREATE TABLE TESTvoterRegistry (

    title VARCHAR(5),

    voterID INTEGER,

    firstName VARCHAR(60),

    middleName VARCHAR(60),

    lastName VARCHAR(60),

    gender VARCHAR(60),

    dateOfBirth DATE,

    residentUnitNumber VARCHAR(20),

    residentStreetNumber INTEGER,

    residentStreetName VARCHAR(60),

    residentsuburb VARCHAR(30),

    residentPostcode INTEGER,

    residentState VARCHAR(30),

    postalUnitNumber INTEGER,

    postalStreetNumber INTEGER,

    postalStreetName VARCHAR(60),

    postalSuburb VARCHAR(60),

    postalPostcode INTEGER,

    postalState VARCHAR(60),

    daytimePhone INTEGER,

    mobile VARCHAR(20),

    emailAddress VARCHAR(40),

    divisionName VARCHAR(50),

    PRIMARY KEY (voterID)

);

CREATE TABLE TESTballot (

    ballotID INTEGER,

    electionEventID VARCHAR,

    PRIMARY KEY (ballotID)

);

CREATE TABLE TESTballotPreferences (

    ballotID INTEGER,

    candidateID VARCHAR (20),

    preference INTEGER,

    PRIMARY KEY (ballotID, candidateID)

);

CREATE TABLE TESTissuanceRecord (

    voterID INTEGER,

    electionEventID VARCHAR,

    issueDate DATE,

    ballotIssue Timestamp,

    pollingStation VARCHAR(50),

    PRIMARY KEY (voterID, electionEventID)

);

CREATE TABLE TESTprefCountRecord (

    electionEventID VARCHAR,

    roundNo INTEGER,

    eliminatedCandidateID VARCHAR (20),

    countStatus VARCHAR, --Done, In-progress, complete

    preferenceAggregate INTEGER,

    PRIMARY KEY (electionEventID, roundNo)

);

CREATE TABLE TESTpreferenceTallyPerRoundPerCandidate (

    electionEventID VARCHAR,

    roundNo integer,

    candidateID VARCHAR(20),

    preferenceTally INTEGER, -- Tally in a round.

    PRIMARY KEY (electionEventID, roundNo, candidateID)

);

Here are there constraints:

alter table TESTelectoralDivisionHistory add constraint TESTelectoralDivisionHistoryKeys

    FOREIGN KEY (divisionName)

        REFERENCES TESTelectoralDivision (divisionName),

    FOREIGN KEY (electionSerialNo)

        REFERENCES TESTelectionMaster (electionSerialNo);

alter table TESTcandidateList add constraint TESTcandidateListKeys

    FOREIGN KEY (partyCode)

        REFERENCES TESTpoliticalParty (partyCode);

alter table TESTvoterRegistry add constraint TESTvoterRegistryKeys

    FOREIGN KEY (divisionName)

        REFERENCES TESTelectoralDivision (divisionName);

alter table TESTcontests add constraint TESTcontestsKeys

    FOREIGN KEY (electionEventID)

        REFERENCES TESTelectionEvent (electionEventID),

    FOREIGN KEY (candidateID)

        REFERENCES TESTcandidateList (candidateID);

alter table TESTballot add constraint TESTballotKeys

    FOREIGN KEY (electionEventID)

        REFERENCES TESTelectionEvent (electionEventID);

alter table TESTballotPreferences add constraint TESTballotPreferencesKeys

    FOREIGN KEY (ballotID)

        REFERENCES TESTballot(ballotID),

    FOREIGN KEY (candidateID)

        REFERENCES TESTcandidateList (candidateID);

alter table TESTissuanceRecord add constraint TESTissuanceRecordKeys

    FOREIGN KEY (voterID)

        REFERENCES TESTvoterRegistry (voterID),

    FOREIGN KEY (electionEventID)

        REFERENCES TESTelectionEvent (electionEventID);

alter table TESTprefCountRecord add constraint TESTprefCountRecordKeys

    FOREIGN KEY (electionEventID)

        REFERENCES TESTelectionEvent (electionEventID),

    FOREIGN KEY (eliminatedCandidateID)

        REFERENCES TESTcandidateList (candidateID);

alter table TESTpreferenceTallyPerRoundPerCandidate add constraint TESTpreferenceTallyPerRoundPerCandidateKeys

    FOREIGN KEY (electionEventID)

        REFERENCES TESTelectionEvent (electionEventID),

    FOREIGN KEY (electionEventID, roundNo)

        REFERENCES TESTprefCountRecord (electionEventID, roundNo),

    FOREIGN KEY (candidateID)

        REFERENCES TESTcandidateList (candidateID);

alter table TESTelectionEvent add constraint TESTelectionEventKeys

    FOREIGN KEY (electionSerialNo)

        REFERENCES TESTelectionMaster (electionSerialNo),

    FOREIGN KEY (divisionName)

        REFERENCES TESTelectoralDivision (divisionName),

    FOREIGN KEY (prefWinnerCandidateID)

        REFERENCES TESTcandidateList (candidateID),

    FOREIGN KEY (prefLoserCandidateID)

        REFERENCES TESTcandidateList (candidateID);

Write an SQL query for the next task:

Make a stored procedure

* Make a NOT null value to check if the row is empty.
* At the end have a TRY catch statement to check for errors.