Lớp: L01 (ĐHBKTPHCM) **Báo cáo bài tập lớn số 2 - 2019**

Danh sách thành viên: **Đề tài: HỆ THỐNG QUẢN LÍ**

**THÔNG TIN TICKLAB**

1. Huỳnh Hoàng Kha, 1511449
2. Châu Thành Đạt, 1710934
3. Nguyễn Ngọc Duy Phong, 1712617
4. Phần chung
5. Các câu lệnh tạo bảng và ràng buộc:

|  |
| --- |
| -- CREATE DATABASE  CREATE DATABASE TickLabInfoSystem;  USE TickLabInfoSystem;  -- CREATE TABLE  -- Project Fund Contributor  CREATE TABLE ProjectFundContributor (  projectFundContributor INT NOT NULL,  PRIMARY KEY(projectFundContributor)  );  -- Person  CREATE TABLE Person (  profileNumber INT NOT NULL,  username TEXT,  passwd TEXT,  gender BIT,  permanentAddress TEXT,  dateOfBirth DATE,  firstName TEXT,  middleName TEXT,  lastName TEXT,  nationality TEXT,  nationalIDNumber VARCHAR(9),  nationalIDIssueDate DATE,  passportNumber TEXT,  passportPlaceOfIssue TEXT,  passportDateOfIssue DATE,  passportDateOfExpiry DATE,  profilePhotoURL TEXT,  projectFundContributor INT,  PRIMARY KEY(profileNumber),  FOREIGN KEY(projectFundContributor) REFERENCES ProjectFundContributor(projectFundContributor)  );  -- TickLab ID CARD  CREATE TABLE TickLabIDCard (  profileNumber INT NOT NULL,  dateOfIssue DATE NOT NULL,  rfidNumber VARCHAR(9) NOT NULL,  PRIMARY KEY(profileNumber, dateOfIssue, rfidNumber),  FOREIGN KEY(profileNumber) REFERENCES Person(profileNumber)  );  -- Contact address  CREATE TABLE PersonContactAddress (  profileNumber INT NOT NULL,  contactAddress VARCHAR(256) NOT NULL,  PRIMARY KEY(profileNumber, contactAddress),  FOREIGN KEY(profileNumber) REFERENCES Person(profileNumber)  );  -- Email address  CREATE TABLE PersonEmailAddress (  profileNumber INT NOT NULL,  emailAddress VARCHAR(256) NOT NULL,  PRIMARY KEY(profileNumber, emailAddress),  FOREIGN KEY(profileNumber) REFERENCES Person(profileNumber)  );  -- System roles  CREATE TABLE PersonSystemRole (  profileNumber INT NOT NULL,  systemRole VARCHAR(256) NOT NULL,  PRIMARY KEY(profileNumber, systemRole),  FOREIGN KEY(profileNumber) REFERENCES Person(profileNumber)  );  -- Person related documents  CREATE TABLE PersonRelatedDoc (  profileNumber INT NOT NULL,  personDocumentURL VARCHAR(256) NOT NULL,  PRIMARY KEY(profileNumber, personDocumentURL),  FOREIGN KEY(profileNumber) REFERENCES Person(profileNumber)  );  -- Person phone number  CREATE TABLE PersonPhoneNumber (  profileNumber INT NOT NULL,  phoneNumber VARCHAR(11) NOT NULL,  PRIMARY KEY(profileNumber, phoneNumber),  FOREIGN KEY(profileNumber) REFERENCES Person(profileNumber)  );  -- Department  CREATE TABLE Department (  departmentID INT NOT NULL,  departmentName TEXT,  departmentDescription TEXT,  PRIMARY KEY(departmentID)  );  -- Project  CREATE TABLE Project (  projectID INT NOT NULL,  projectName TEXT,  projectStartTime DATE,  projectEndTime DATE,  projectStatus TEXT,  expectedFinishTime DATE,  projectDescription TEXT,  TickLabBudget MONEY,  projectManager INT,  ofDepartment INT,  PRIMARY KEY(projectID),  FOREIGN KEY(projectManager) REFERENCES Person(profileNumber),  FOREIGN KEY(ofDepartment) REFERENCES Department(departmentID)  );  -- Project's paricipant  CREATE TABLE ProjectParticipant (  participantID INT NOT NULL,  projectID INT,  profileNumber INT,  formDate DATE,  toDate DATE,  PRIMARY KEY(participantID),  FOREIGN KEY(projectID) REFERENCES Project(projectID),  FOREIGN KEY(profileNumber) REFERENCES Person(profileNumber)  );  -- Project related documents  CREATE TABLE ProjectRelatedDoc (  projectID INT NOT NULL,  projectDocumentURL VARCHAR(256) NOT NULL,  PRIMARY KEY(projectID, projectDocumentURL),  FOREIGN KEY(projectID) REFERENCES Project(projectID)  );  -- Contributing  CREATE TABLE Contributing (  projectFundContributor INT NOT NULL,  projectID INT NOT NULL,  contributingDate DATETIME,  amountOfMoney MONEY,  PRIMARY KEY(projectFundContributor, projectID),  FOREIGN KEY(projectFundContributor) REFERENCES ProjectFundContributor(projectFundContributor),  FOREIGN KEY(projectID) REFERENCES Project(projectID)  );  -- Internal project  CREATE TABLE InternalProject (  internalProjectID INT NOT NULL,  PRIMARY KEY(internalProjectID),  FOREIGN KEY(internalProjectID) REFERENCES Project(projectID)  );  -- Community activity  CREATE TABLE CommunityActivity (  communityID INT NOT NULL,  place TEXT,  PRIMARY KEY(communityID)  );  -- Seminar workshop  CREATE TABLE SeminarWorkshop (  projectID INT NOT NULL,  topic TEXT,  communityID INT,  FOREIGN KEY(projectID) REFERENCES Project(projectID),  FOREIGN KEY(communityID) REFERENCES CommunityActivity(communityID)  );  -- Interview  CREATE TABLE Interview (  projectID INT,  requirement TEXT,  communityID INT,  FOREIGN KEY(projectID) REFERENCES Project(projectID),  FOREIGN KEY(communityID) REFERENCES CommunityActivity(communityID)  );  -- Application form  CREATE TABLE ApplicationForm (  communityIDForm INT NOT NULL,  formID INT NOT NULL,  formFirstName TEXT,  formMiddleName TEXT,  formLastName TEXT,  formGender BIT,  formDateOfBirth DATE,  formPhoneNumber VARCHAR(11),  formContactAddress TEXT,  formSocialAccount TEXT,  formEmail TEXT,  formCVURL TEXT,  PRIMARY KEY(communityIDForm, formID),  FOREIGN KEY(communityIDForm) REFERENCES CommunityActivity(communityID)  );  -- Company  CREATE TABLE Company (  taxIDNumber VARCHAR(9) NOT NULL,  conpanyName TEXT,  companyDescription TEXT,  projectFundContributor INT,  PRIMARY KEY(taxIDNumber),  FOREIGN KEY(projectFundContributor) REFERENCES ProjectFundContributor(projectFundContributor)  );  -- Company related documents  CREATE TABLE CompanyRelatedDoc (  taxIDNumber VARCHAR(9) NOT NULL,  companyDocumentURL VARCHAR(256) NOT NULL,  PRIMARY KEY(taxIDNumber, companyDocumentURL),  FOREIGN KEY(taxIDNumber) REFERENCES Company(taxIDNumber)  );  -- Position  CREATE TABLE WorkPosition (  posID INT NOT NULL,  posName TEXT,  posDescription TEXT,  posInDepartment INT,  PRIMARY KEY(posID),  FOREIGN KEY(posInDepartment) REFERENCES Department(departmentID)  );  -- Person take position  CREATE TABLE Taking (  profileNumberTake INT NOT NULL,  posIDTake INT NOT NULL,  takeFromDate DATE,  takeToDate DATE,  PRIMARY KEY(profileNumberTake, posIDTake),  FOREIGN KEY(profileNumberTake) REFERENCES Person(profileNumber),  FOREIGN KEY(posIDTake) REFERENCES WorkPosition(posID)  );  -- Task  CREATE TABLE Task (  taskID INT NOT NULL,  taskStartTime DATE,  taskEndTime DATE,  whatToDo TEXT,  taskDescription TEXT,  taskStatus TEXT,  ofProject INT,  PRIMARY KEY(taskID),  FOREIGN KEY(ofProject) REFERENCES Project(projectID)  );  -- Task's participant  CREATE TABLE TaskParticipant (  participantID INT NOT NULL,  taskID INT NOT NULL,  fromDate DATE,  toDate DATE,  PRIMARY KEY(participantID, taskID),  FOREIGN KEY(participantID) REFERENCES Participate(participantID),  FOREIGN KEY(taskID) REFERENCES Task(taskID)  );  -- Task remake  CREATE TABLE TaskRemark (  taskID INT NOT NULL,  taskRemark VARCHAR(256) NOT NULL,  PRIMARY KEY(taskID, taskRemark),  FOREIGN KEY(taskID) REFERENCES Task(taskID)  );  -- Duty  CREATE TABLE Duty (  dutyID INT NOT NULL,  dutyName TEXT,  shift TEXT,  dutyDescription TEXT,  PRIMARY KEY(dutyID)  );  -- Has to do duty  CREATE TABLE HaveToDoDuty (  profileNumberDuty INT NOT NULL,  dutyID INT NOT NULL,  PRIMARY KEY(profileNumberDuty, dutyID),  FOREIGN KEY(profileNumberDuty) REFERENCES Person(profileNumber),  FOREIGN KEY(dutyID) REFERENCES Duty(dutyID)  );  -- Infrastructures  CREATE TABLE Infrastructure (  infraID INT NOT NULL,  infraName TEXT,  totalNumber INT,  numberOfAvailable INT,  infraDescription TEXT,  PRIMARY KEY(infraID)  );  -- Borrow record  CREATE TABLE BorrowRecord (  borrowID INT NOT NULL,  borrowDate DATE,  borrowStatus BIT,  borrower INT,  PRIMARY KEY(borrowID),  FOREIGN KEY(borrower) REFERENCES Person(profileNumber)  );  -- Include  CREATE TABLE Including (  borrowIDInclude INT NOT NULL,  infraIDInclude INT NOT NULL,  numberOfItem INT,  dueDate DATE,  returnDate DATE,  PRIMARY KEY(borrowIDInclude, infraIDInclude),  FOREIGN KEY(borrowIDInclude) REFERENCES BorrowRecord(borrowID),  FOREIGN KEY(infraIDInclude) REFERENCES Infrastructure(infraID)  );  -- Fund  CREATE TABLE Fund (  fundID INT NOT NULL,  currentBudget MONEY,  originalCapital MONEY,  treasurer INT,  PRIMARY KEY(fundID),  FOREIGN KEY(treasurer) REFERENCES Person(profileNumber)  );  -- Revenue category  CREATE TABLE RevenueCategory (  revenueID INT NOT NULL,  revenueName TEXT,  moneyHaveToPay MONEY,  moneyPay MONEY,  moneyRest MONEY,  revenueDeadline DATE,  revenueDescription TEXT,  fundIDRevenue INT,  PRIMARY KEY(revenueID),  FOREIGN KEY(fundIDRevenue) REFERENCES Fund(fundID)  );  -- Expediture category  CREATE TABLE ExpeditureCategory (  expeditureID INT NOT NULL,  expeditureName TEXT,  expeditureDate DATE,  expeditureMoney MONEY,  expeditureDescription TEXT,  fundIDExpediture INT,  PRIMARY KEY(expeditureID),  FOREIGN KEY(fundIDExpediture) REFERENCES Person(profileNumber)  );  -- Have to pay  CREATE TABLE HaveToPay (  profileNumberPay INT NOT NULL,  revenueIDPay INT NOT NULL,  weighting FLOAT,  amount INT,  PRIMARY KEY(profileNumberPay, revenueIDPay),  FOREIGN KEY(profileNumberPay) REFERENCES Person(profileNumber),  FOREIGN KEY(revenueIDPay) REFERENCES RevenueCategory(revenueID)  ); |

1. Các câu lệnh tạo chỉ mục:

|  |
| --- |
| -- Create index on Person table  CREATE INDEX PersonIndex ON Person(profileNumber)  -- Create index on Project table  CREATE INDEX Project ON Project(projectID) |

1. Các câu lệnh insert dữ liệu:

Chèn dữ liệu vào bảng Person

|  |
| --- |
| INSERT INTO Person VALUES(0, 'ethan.roberts1112', 'eY5FXIcs3Gw5', 1, NULL, CONVERT(DATETIME, 11-12-2000, 105), 'Ethan', NULL, 'Roberts', 'Viet Nam', '618927954', CONVERT(DATETIME, 03-04-2017, 105), NULL, NULL, NULL, NULL, NULL, NULL);  INSERT INTO Person VALUES(1, 'mia.johnson1510', 'P2rNvS0cS4Qei6f94K61S', 1, NULL, CONVERT(DATETIME, 15-10-1989, 105), 'Mia', NULL, 'Johnson', 'Viet Nam', '637882913', CONVERT(DATETIME, 23-08-2006, 105), NULL, NULL, NULL, NULL, NULL, NULL);  INSERT INTO Person VALUES(2, 'mason.rodriguez287', '52f3cE67a20oYf', 1, NULL, CONVERT(DATETIME, 28-07-1983, 105), 'Mason', NULL, 'Rodriguez', 'Viet Nam', '777335998', CONVERT(DATETIME, 06-01-2000, 105), NULL, NULL, NULL, NULL, NULL, NULL);  INSERT INTO Person VALUES(3, 'margaret.rodriguez113', '2048jo14', 0, NULL, CONVERT(DATETIME, 11-03-1999, 105), 'Margaret', NULL, 'Rodriguez', 'Viet Nam', '618984143', CONVERT(DATETIME, 08-07-2015, 105), NULL, NULL, NULL, NULL, NULL, NULL);  INSERT INTO Person VALUES(4, 'sarah.rodriguez89', 't1nY619SN8fUxrBN0ok18', 1, NULL, CONVERT(DATETIME, 08-09-1992, 105), 'Sarah', NULL, 'Rodriguez', 'Viet Nam', '722828907', CONVERT(DATETIME, 07-12-2009, 105), NULL, NULL, NULL, NULL, NULL, NULL);  INSERT INTO Person VALUES(5, 'mason.wilson115', 'w259cpal7u503ZltKCWR', 1, NULL, CONVERT(DATETIME, 11-05-1999, 105), 'Mason', NULL, 'Wilson', 'Viet Nam', '229419698', CONVERT(DATETIME, 26-06-2014, 105), NULL, NULL, NULL, NULL, NULL, NULL);  INSERT INTO Person VALUES(6, 'daniel.taylor222', 'J3WXRbDg', 0, NULL, CONVERT(DATETIME, 22-02-1991, 105), 'Daniel', NULL, 'Taylor', 'Viet Nam', '582544161', CONVERT(DATETIME, 05-12-2008, 105), NULL, NULL, NULL, NULL, NULL, NULL);  INSERT INTO Person VALUES(7, 'richard.rodriguez2810', '3hMNE9X342p45ec8782J', 0, NULL, CONVERT(DATETIME, 28-10-1996, 105), 'Richard', NULL, 'Rodriguez', 'Viet Nam', '444429307', CONVERT(DATETIME, 25-07-2012, 105), NULL, NULL, NULL, NULL, NULL, NULL);  INSERT INTO Person VALUES(8, 'ethan.williams283', 'e795AU1Chf260', 1, NULL, CONVERT(DATETIME, 28-03-1987, 105), 'Ethan', NULL, 'Williams', 'Viet Nam', '034005310', CONVERT(DATETIME, 01-12-2004, 105), NULL, NULL, NULL, NULL, NULL, NULL);  INSERT INTO Person VALUES(9, 'barbara.wilson159', 'EMD8j8hSqo063oeDP5N32T7D4OM', 0, NULL, CONVERT(DATETIME, 15-09-1993, 105), 'Barbara', NULL, 'Wilson', 'Viet Nam', '111730048', CONVERT(DATETIME, 02-09-2009, 105), NULL, NULL, NULL, NULL, NULL, NULL);  INSERT INTO Person VALUES(10, 'liam.thomas16', 'H0r0Km7WqgOAMwDyxtbH2Yc1bP', 1, NULL, CONVERT(DATETIME, 01-06-1994, 105), 'Liam', NULL, 'Thomas', 'Viet Nam', '265324442', CONVERT(DATETIME, 03-06-2010, 105), NULL, NULL, NULL, NULL, NULL, NULL);  INSERT INTO Person VALUES(11, 'david.taylor244', 'v34Rj165IXz76KU', 0, NULL, CONVERT(DATETIME, 24-04-1982, 105), 'David', NULL, 'Taylor', 'Viet Nam', '103155221', CONVERT(DATETIME, 05-01-1998, 105), NULL, NULL, NULL, NULL, NULL, NULL);  INSERT INTO Person VALUES(12, 'liam.miller2512', '20i51u0pNwma95Nhqr6L1', 1, NULL, CONVERT(DATETIME, 25-12-1989, 105), 'Liam', NULL, 'Miller', 'Viet Nam', '034175332', CONVERT(DATETIME, 19-10-2004, 105), NULL, NULL, NULL, NULL, NULL, NULL);  INSERT INTO Person VALUES(13, 'isabella.jones177', 'cVa1M0Hwbz8W5rMmH356bR9', 0, NULL, CONVERT(DATETIME, 17-07-1989, 105), 'Isabella', NULL, 'Jones', 'Viet Nam', '505880132', CONVERT(DATETIME, 08-09-2005, 105), NULL, NULL, NULL, NULL, NULL, NULL);  INSERT INTO Person VALUES(14, 'jacob.rodriguez224', '95Pib3bG088O3KB', 0, NULL, CONVERT(DATETIME, 22-04-1990, 105), 'Jacob', NULL, 'Rodriguez', 'Viet Nam', '356583948', CONVERT(DATETIME, 18-12-2005, 105), NULL, NULL, NULL, NULL, NULL, NULL);  INSERT INTO Person VALUES(15, 'mary.davies141', 'X9U45w8Q3iiFdU4oQU5', 1, NULL, CONVERT(DATETIME, 14-01-1987, 105), 'Mary', NULL, 'Davies', 'Viet Nam', '015176002', CONVERT(DATETIME, 31-10-2003, 105), NULL, NULL, NULL, NULL, NULL, NULL);  INSERT INTO Person VALUES(16, 'mary.brown281', 'U6W3EUk546zZOD03z9zZf2', 1, NULL, CONVERT(DATETIME, 28-01-1997, 105), 'Mary', NULL, 'Brown', 'Viet Nam', '389287014', CONVERT(DATETIME, 12-02-2013, 105), NULL, NULL, NULL, NULL, NULL, NULL);  INSERT INTO Person VALUES(17, 'charlotte.wilson104', 'j7laa2RSiDT037ejtHc8bExXg6', 1, NULL, CONVERT(DATETIME, 10-04-1984, 105), 'Charlotte', NULL, 'Wilson', 'Viet Nam', '089360697', CONVERT(DATETIME, 05-04-2000, 105), NULL, NULL, NULL, NULL, NULL, NULL);  INSERT INTO Person VALUES(18, 'richard.johnson76', 'Tw6og5ZvqXwm7y', 0, NULL, CONVERT(DATETIME, 07-06-1980, 105), 'Richard', NULL, 'Johnson', 'Viet Nam', '049460362', CONVERT(DATETIME, 09-02-1996, 105), NULL, NULL, NULL, NULL, NULL, NULL);  INSERT INTO Person VALUES(19, 'isabella.wilson34', '2B9m8f6A8C7NYq28wybacKKk98C', 0, NULL, CONVERT(DATETIME, 03-04-1997, 105), 'Isabella', NULL, 'Wilson', 'Viet Nam', '072552148', CONVERT(DATETIME, 08-03-2013, 105), NULL, NULL, NULL, NULL, NULL, NULL);  INSERT INTO Person VALUES(20, 'thomas.smith2110', 'HoM4yAxvh0iEsgHJ430sn34', 0, NULL, CONVERT(DATETIME, 21-10-2000, 105), 'Thomas', NULL, 'Smith', 'Viet Nam', '662904931', CONVERT(DATETIME, 19-10-2015, 105), NULL, NULL, NULL, NULL, NULL, NULL);  INSERT INTO Person VALUES(21, 'emma.taylor262', '4TpnpyGiOqW25Q2OeS7vipL', 0, NULL, CONVERT(DATETIME, 26-02-1980, 105), 'Emma', NULL, 'Taylor', 'Viet Nam', '298227858', CONVERT(DATETIME, 23-12-1995, 105), NULL, NULL, NULL, NULL, NULL, NULL);  INSERT INTO Person VALUES(22, 'emily.miller113', 'gmKIdOk8qIPO5M', 1, NULL, CONVERT(DATETIME, 11-03-1997, 105), 'Emily', NULL, 'Miller', 'Viet Nam', '934788163', CONVERT(DATETIME, 08-12-2013, 105), NULL, NULL, NULL, NULL, NULL, NULL);  INSERT INTO Person VALUES(23, 'isabella.williams36', 'SabLm0m9j342t70m', 1, NULL, CONVERT(DATETIME, 03-06-1996, 105), 'Isabella', NULL, 'Williams', 'Viet Nam', '037395972', CONVERT(DATETIME, 23-07-2013, 105), NULL, NULL, NULL, NULL, NULL, NULL);  INSERT INTO Person VALUES(24, 'ava.davies288', 'GSe81t5j0ij68N56oWiYTD28h4', 0, NULL, CONVERT(DATETIME, 28-08-1989, 105), 'Ava', NULL, 'Davies', 'Viet Nam', '272584739', CONVERT(DATETIME, 04-10-2005, 105), NULL, NULL, NULL, NULL, NULL, NULL);  INSERT INTO Person VALUES(25, 'isabella.davis211', 'R4j0jm9HeQaudCJcYfVi6E', 0, NULL, CONVERT(DATETIME, 21-01-1994, 105), 'Isabella', NULL, 'Davis', 'Viet Nam', '898999420', CONVERT(DATETIME, 10-08-2011, 105), NULL, NULL, NULL, NULL, NULL, NULL);  INSERT INTO Person VALUES(26, 'patricia.davis2612', 'b1U3qlBwX9as4H0GNp5kmU', 1, NULL, CONVERT(DATETIME, 26-12-1989, 105), 'Patricia', NULL, 'Davis', 'Viet Nam', '079158639', CONVERT(DATETIME, 14-02-2004, 105), NULL, NULL, NULL, NULL, NULL, NULL);  INSERT INTO Person VALUES(27, 'joseph.roberts225', 'leJmnry9Ir9WcX23', 0, NULL, CONVERT(DATETIME, 22-05-1980, 105), 'Joseph', NULL, 'Roberts', 'Viet Nam', '075531888', CONVERT(DATETIME, 28-04-1996, 105), NULL, NULL, NULL, NULL, NULL, NULL);  INSERT INTO Person VALUES(28, 'charlotte.smith303', 'h2CNUNDfmZATck8Yz', 0, NULL, CONVERT(DATETIME, 30-03-1989, 105), 'Charlotte', NULL, 'Smith', 'Viet Nam', '311257621', CONVERT(DATETIME, 10-02-2005, 105), NULL, NULL, NULL, NULL, NULL, NULL);  INSERT INTO Person VALUES(29, 'linda.smith3112', 'L4GfP7kC1d6XZ01v32xiFFc9', 0, NULL, CONVERT(DATETIME, 31-12-1991, 105), 'Linda', NULL, 'Smith', 'Viet Nam', '518368861', CONVERT(DATETIME, 20-01-2007, 105), NULL, NULL, NULL, NULL, NULL, NULL); |

|  |
| --- |
| INSERT INTO WorkPosition VALUES(0, 'IT Staff', NULL, 0);  INSERT INTO WorkPosition VALUES(1, 'IT Vice Leader', NULL, 0);  INSERT INTO WorkPosition VALUES(2, 'IT Leader', NULL, 0);  INSERT INTO WorkPosition VALUES(3, 'HR Staff', NULL, 1);  INSERT INTO WorkPosition VALUES(4, 'HR Vice Leader', NULL, 1);  INSERT INTO WorkPosition VALUES(5, 'HR Leader', NULL, 1);  INSERT INTO WorkPosition VALUES(6, 'EE Staff', NULL, 2);  INSERT INTO WorkPosition VALUES(7, 'EE Vice Leader', NULL, 2);  INSERT INTO WorkPosition VALUES(8, 'EE Leader', NULL, 2);  INSERT INTO WorkPosition VALUES(9, 'ME Staff', NULL, 3);  INSERT INTO WorkPosition VALUES(10, 'ME Vice Leader', NULL, 3);  INSERT INTO WorkPosition VALUES(11, 'ME Leader', NULL, 3);  INSERT INTO WorkPosition VALUES(12, 'Director', NULL, 4);  INSERT INTO WorkPosition VALUES(13, 'Vice Director', NULL, 4);  INSERT INTO WorkPosition VALUES(14, 'Treasurer', NULL, 4); |

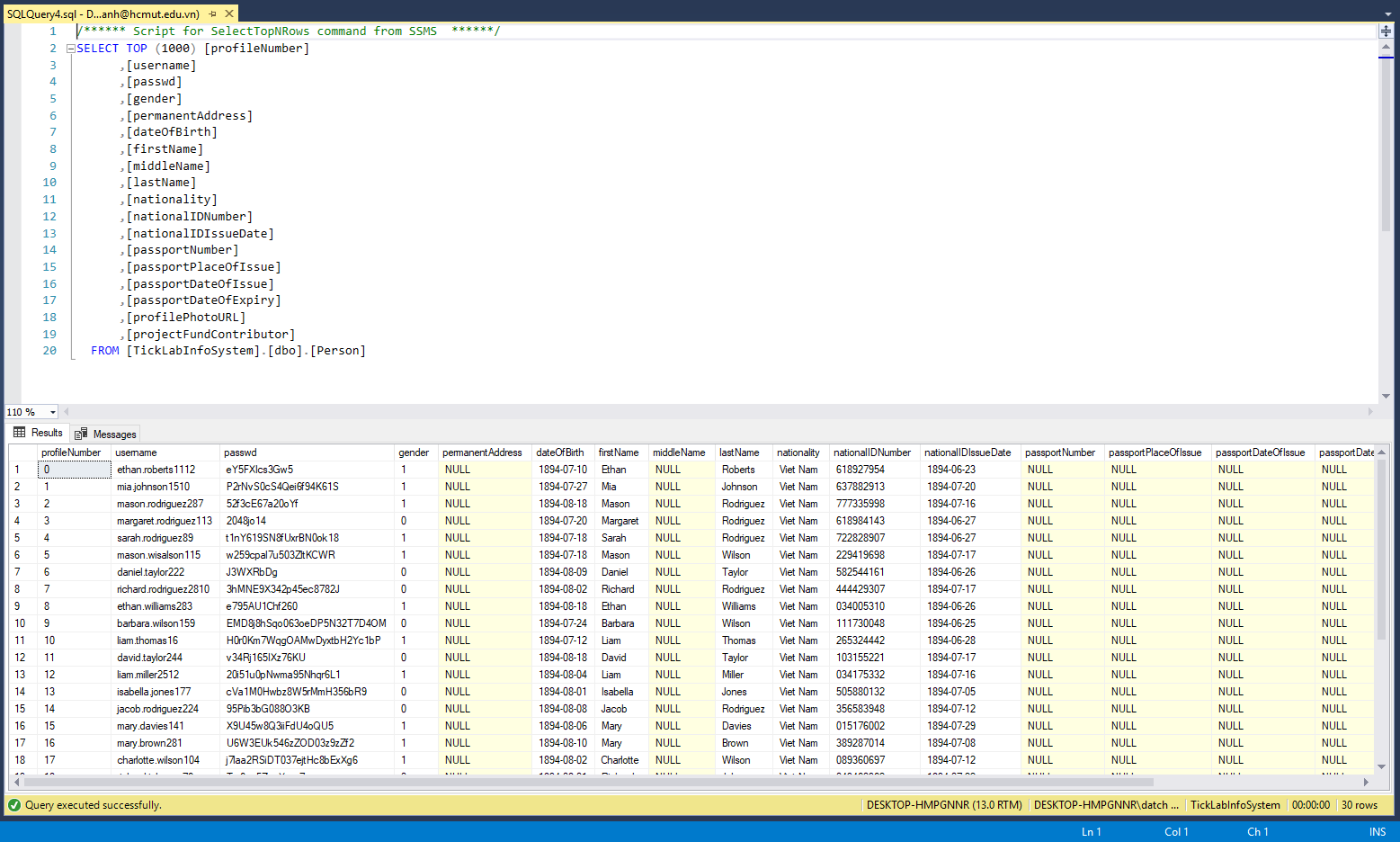
|  |
| --- |
| INSERT INTO Department VALUES(0, 'IT', NULL);  INSERT INTO Department VALUES(1, 'HR', NULL);  INSERT INTO Department VALUES(2, 'EE', NULL);  INSERT INTO Department VALUES(3, 'ME', NULL);  INSERT INTO Department VALUES(4, 'Administration', NULL); |

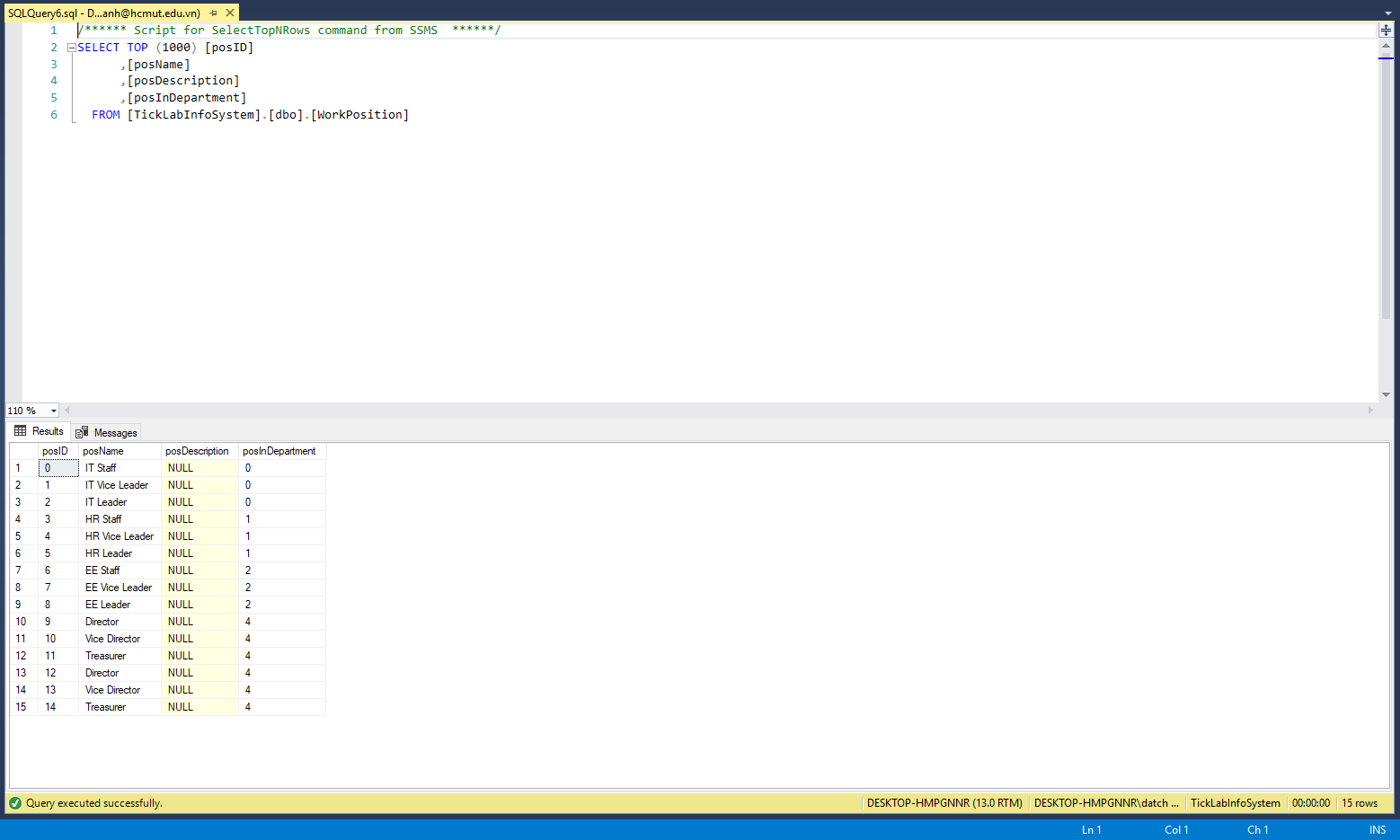
|  |
| --- |
| INSERT INTO Taking VALUES(28, 1, NULL, NULL);  INSERT INTO Taking VALUES(8, 2, NULL, NULL);  INSERT INTO Taking VALUES(10, 4, NULL, NULL);  INSERT INTO Taking VALUES(29, 5, NULL, NULL);  INSERT INTO Taking VALUES(1, 7, NULL, NULL);  INSERT INTO Taking VALUES(5, 8, NULL, NULL);  INSERT INTO Taking VALUES(12, 10, NULL, NULL);  INSERT INTO Taking VALUES(14, 11, NULL, NULL);  INSERT INTO Taking VALUES(24, 12, NULL, NULL);  INSERT INTO Taking VALUES(0, 14, NULL, NULL);  INSERT INTO Taking VALUES(20, 13, NULL, NULL);  INSERT INTO Taking VALUES(21, 3, NULL, NULL);  INSERT INTO Taking VALUES(22, 0, NULL, NULL);  INSERT INTO Taking VALUES(4, 3, NULL, NULL);  INSERT INTO Taking VALUES(25, 0, NULL, NULL);  INSERT INTO Taking VALUES(3, 3, NULL, NULL);  INSERT INTO Taking VALUES(6, 6, NULL, NULL);  INSERT INTO Taking VALUES(15, 3, NULL, NULL);  INSERT INTO Taking VALUES(2, 0, NULL, NULL);  INSERT INTO Taking VALUES(18, 6, NULL, NULL);  INSERT INTO Taking VALUES(19, 6, NULL, NULL);  INSERT INTO Taking VALUES(7, 3, NULL, NULL);  INSERT INTO Taking VALUES(27, 6, NULL, NULL);  INSERT INTO Taking VALUES(9, 9, NULL, NULL);  INSERT INTO Taking VALUES(16, 9, NULL, NULL);  INSERT INTO Taking VALUES(13, 3, NULL, NULL);  INSERT INTO Taking VALUES(17, 3, NULL, NULL);  INSERT INTO Taking VALUES(23, 6, NULL, NULL);  INSERT INTO Taking VALUES(26, 9, NULL, NULL);  INSERT INTO Taking VALUES(11, 3, NULL, NULL); |

|  |
| --- |
| INSERT INTO PersonEmailAddress VALUES(0, 'ethan.roberts1112@gmail.com');  INSERT INTO PersonEmailAddress VALUES(0, 'ethan.roberts1112@yahoo.com');  INSERT INTO PersonEmailAddress VALUES(1, 'mia.johnson1510@yahoo.com');  INSERT INTO PersonEmailAddress VALUES(2, 'mason.rodriguez287@hotmail.com');  INSERT INTO PersonEmailAddress VALUES(2, 'mason.rodriguez287@yahoo.com');  INSERT INTO PersonEmailAddress VALUES(2, 'mason.rodriguez287@gmail.com');  INSERT INTO PersonEmailAddress VALUES(3, 'margaret.rodriguez113@hotmail.com');  INSERT INTO PersonEmailAddress VALUES(3, 'margaret.rodriguez113@yahoo.com');  INSERT INTO PersonEmailAddress VALUES(3, 'margaret.rodriguez113@gmail.com');  INSERT INTO PersonEmailAddress VALUES(4, 'sarah.rodriguez89@yahoo.com');  INSERT INTO PersonEmailAddress VALUES(4, 'sarah.rodriguez89@hotmail.com');  INSERT INTO PersonEmailAddress VALUES(4, 'sarah.rodriguez89@gmail.com');  INSERT INTO PersonEmailAddress VALUES(5, 'mason.wilson115@hotmail.com');  INSERT INTO PersonEmailAddress VALUES(5, 'mason.wilson115@gmail.com');  INSERT INTO PersonEmailAddress VALUES(5, 'mason.wilson115@yahoo.com');  INSERT INTO PersonEmailAddress VALUES(6, 'daniel.taylor222@yahoo.com');  INSERT INTO PersonEmailAddress VALUES(6, 'daniel.taylor222@hotmail.com');  INSERT INTO PersonEmailAddress VALUES(7, 'richard.rodriguez2810@yahoo.com');  INSERT INTO PersonEmailAddress VALUES(8, 'ethan.williams283@gmail.com');  INSERT INTO PersonEmailAddress VALUES(8, 'ethan.williams283@yahoo.com');  INSERT INTO PersonEmailAddress VALUES(9, 'barbara.wilson159@gmail.com');  INSERT INTO PersonEmailAddress VALUES(10, 'liam.thomas16@gmail.com');  INSERT INTO PersonEmailAddress VALUES(11, 'david.taylor244@gmail.com');  INSERT INTO PersonEmailAddress VALUES(11, 'david.taylor244@hotmail.com');  INSERT INTO PersonEmailAddress VALUES(12, 'liam.miller2512@hotmail.com');  INSERT INTO PersonEmailAddress VALUES(12, 'liam.miller2512@gmail.com');  INSERT INTO PersonEmailAddress VALUES(13, 'isabella.jones177@gmail.com');  INSERT INTO PersonEmailAddress VALUES(14, 'jacob.rodriguez224@hotmail.com');  INSERT INTO PersonEmailAddress VALUES(14, 'jacob.rodriguez224@gmail.com');  INSERT INTO PersonEmailAddress VALUES(15, 'mary.davies141@gmail.com');  INSERT INTO PersonEmailAddress VALUES(15, 'mary.davies141@yahoo.com');  INSERT INTO PersonEmailAddress VALUES(15, 'mary.davies141@hotmail.com');  INSERT INTO PersonEmailAddress VALUES(16, 'mary.brown281@hotmail.com');  INSERT INTO PersonEmailAddress VALUES(16, 'mary.brown281@yahoo.com');  INSERT INTO PersonEmailAddress VALUES(17, 'charlotte.wilson104@yahoo.com');  INSERT INTO PersonEmailAddress VALUES(17, 'charlotte.wilson104@gmail.com');  INSERT INTO PersonEmailAddress VALUES(18, 'richard.johnson76@yahoo.com');  INSERT INTO PersonEmailAddress VALUES(19, 'isabella.wilson34@yahoo.com');  INSERT INTO PersonEmailAddress VALUES(19, 'isabella.wilson34@hotmail.com');  INSERT INTO PersonEmailAddress VALUES(20, 'thomas.smith2110@gmail.com');  INSERT INTO PersonEmailAddress VALUES(20, 'thomas.smith2110@hotmail.com');  INSERT INTO PersonEmailAddress VALUES(21, 'emma.taylor262@hotmail.com');  INSERT INTO PersonEmailAddress VALUES(22, 'emily.miller113@hotmail.com');  INSERT INTO PersonEmailAddress VALUES(23, 'isabella.williams36@gmail.com');  INSERT INTO PersonEmailAddress VALUES(23, 'isabella.williams36@hotmail.com');  INSERT INTO PersonEmailAddress VALUES(23, 'isabella.williams36@yahoo.com');  INSERT INTO PersonEmailAddress VALUES(24, 'ava.davies288@yahoo.com');  INSERT INTO PersonEmailAddress VALUES(24, 'ava.davies288@hotmail.com');  INSERT INTO PersonEmailAddress VALUES(24, 'ava.davies288@gmail.com');  INSERT INTO PersonEmailAddress VALUES(25, 'isabella.davis211@gmail.com');  INSERT INTO PersonEmailAddress VALUES(26, 'patricia.davis2612@hotmail.com');  INSERT INTO PersonEmailAddress VALUES(26, 'patricia.davis2612@yahoo.com');  INSERT INTO PersonEmailAddress VALUES(27, 'joseph.roberts225@yahoo.com');  INSERT INTO PersonEmailAddress VALUES(27, 'joseph.roberts225@hotmail.com');  INSERT INTO PersonEmailAddress VALUES(27, 'joseph.roberts225@gmail.com');  INSERT INTO PersonEmailAddress VALUES(28, 'charlotte.smith303@gmail.com');  INSERT INTO PersonEmailAddress VALUES(29, 'linda.smith3112@yahoo.com');  INSERT INTO PersonEmailAddress VALUES(29, 'linda.smith3112@gmail.com');  INSERT INTO PersonEmailAddress VALUES(29, 'linda.smith3112@hotmail.com'); |

|  |
| --- |
| INSERT INTO TickLabIDCard VALUES(0, CONVERT(DATETIME, 24-01-2018, 105), '498949875');  INSERT INTO TickLabIDCard VALUES(1, CONVERT(DATETIME, 04-08-2007, 105), '878095690');  INSERT INTO TickLabIDCard VALUES(2, CONVERT(DATETIME, 20-08-2001, 105), '580177467');  INSERT INTO TickLabIDCard VALUES(2, CONVERT(DATETIME, 29-11-2003, 105), '121835714');  INSERT INTO TickLabIDCard VALUES(3, CONVERT(DATETIME, 09-02-2018, 105), '302788597');  INSERT INTO TickLabIDCard VALUES(4, CONVERT(DATETIME, 26-04-2014, 105), '137127534');  INSERT INTO TickLabIDCard VALUES(4, CONVERT(DATETIME, 22-01-2010, 105), '141266330');  INSERT INTO TickLabIDCard VALUES(5, CONVERT(DATETIME, 25-08-2018, 105), '816980139');  INSERT INTO TickLabIDCard VALUES(5, CONVERT(DATETIME, 29-04-2017, 105), '655405065');  INSERT INTO TickLabIDCard VALUES(6, CONVERT(DATETIME, 10-01-2009, 105), '622505252');  INSERT INTO TickLabIDCard VALUES(6, CONVERT(DATETIME, 18-06-2013, 105), '280068306');  INSERT INTO TickLabIDCard VALUES(7, CONVERT(DATETIME, 22-08-2017, 105), '520756181');  INSERT INTO TickLabIDCard VALUES(8, CONVERT(DATETIME, 09-09-2007, 105), '926523212');  INSERT INTO TickLabIDCard VALUES(8, CONVERT(DATETIME, 20-09-2008, 105), '874005995');  INSERT INTO TickLabIDCard VALUES(9, CONVERT(DATETIME, 27-10-2013, 105), '511123397');  INSERT INTO TickLabIDCard VALUES(9, CONVERT(DATETIME, 18-06-2011, 105), '048439249');  INSERT INTO TickLabIDCard VALUES(10, CONVERT(DATETIME, 02-06-2013, 105), '766309080');  INSERT INTO TickLabIDCard VALUES(11, CONVERT(DATETIME, 19-06-2001, 105), '395896257');  INSERT INTO TickLabIDCard VALUES(12, CONVERT(DATETIME, 29-03-2011, 105), '930251577');  INSERT INTO TickLabIDCard VALUES(12, CONVERT(DATETIME, 02-09-2010, 105), '785788423');  INSERT INTO TickLabIDCard VALUES(13, CONVERT(DATETIME, 26-06-2010, 105), '041772131');  INSERT INTO TickLabIDCard VALUES(13, CONVERT(DATETIME, 28-10-2007, 105), '150512215');  INSERT INTO TickLabIDCard VALUES(14, CONVERT(DATETIME, 22-07-2012, 105), '284275840');  INSERT INTO TickLabIDCard VALUES(14, CONVERT(DATETIME, 28-07-2008, 105), '358410334');  INSERT INTO TickLabIDCard VALUES(15, CONVERT(DATETIME, 18-05-2005, 105), '560983579');  INSERT INTO TickLabIDCard VALUES(15, CONVERT(DATETIME, 08-01-2006, 105), '740235416');  INSERT INTO TickLabIDCard VALUES(16, CONVERT(DATETIME, 18-11-2015, 105), '912499376');  INSERT INTO TickLabIDCard VALUES(17, CONVERT(DATETIME, 24-01-2004, 105), '977165013');  INSERT INTO TickLabIDCard VALUES(18, CONVERT(DATETIME, 09-07-1998, 105), '744906119');  INSERT INTO TickLabIDCard VALUES(18, CONVERT(DATETIME, 27-06-2001, 105), '197093331');  INSERT INTO TickLabIDCard VALUES(19, CONVERT(DATETIME, 31-03-2016, 105), '598839513');  INSERT INTO TickLabIDCard VALUES(19, CONVERT(DATETIME, 28-03-2018, 105), '819813343');  INSERT INTO TickLabIDCard VALUES(20, CONVERT(DATETIME, 18-08-2018, 105), '176173060');  INSERT INTO TickLabIDCard VALUES(21, CONVERT(DATETIME, 25-05-1999, 105), '720954325');  INSERT INTO TickLabIDCard VALUES(22, CONVERT(DATETIME, 28-02-2017, 105), '157579237');  INSERT INTO TickLabIDCard VALUES(23, CONVERT(DATETIME, 17-12-2014, 105), '638106876');  INSERT INTO TickLabIDCard VALUES(23, CONVERT(DATETIME, 20-11-2014, 105), '956154138');  INSERT INTO TickLabIDCard VALUES(24, CONVERT(DATETIME, 22-02-2007, 105), '133529048');  INSERT INTO TickLabIDCard VALUES(25, CONVERT(DATETIME, 14-03-2013, 105), '030321108');  INSERT INTO TickLabIDCard VALUES(25, CONVERT(DATETIME, 09-04-2012, 105), '289230282');  INSERT INTO TickLabIDCard VALUES(26, CONVERT(DATETIME, 13-02-2011, 105), '249882717');  INSERT INTO TickLabIDCard VALUES(26, CONVERT(DATETIME, 25-06-2009, 105), '340153423');  INSERT INTO TickLabIDCard VALUES(27, CONVERT(DATETIME, 06-05-2001, 105), '249044985');  INSERT INTO TickLabIDCard VALUES(27, CONVERT(DATETIME, 24-10-2002, 105), '017198832');  INSERT INTO TickLabIDCard VALUES(28, CONVERT(DATETIME, 27-10-2007, 105), '834020121');  INSERT INTO TickLabIDCard VALUES(29, CONVERT(DATETIME, 22-03-2011, 105), '805889573'); |

Các câu lệnh truy vấn bảng và dữ liệu được hiển thị sau khi thực hiện insert, màn hình hiển thị kết quả dự liệu của các bảng từ DBMS:





1. Phần riêng

**Thành viên 1:**

**Họ tên: Huỳnh Hoàng Kha**

**MSSV: 1511449**

1. Thủ tục chèn dữ liệu:

|  |
| --- |
| -- Insert to WorkPosition--  CREATE PROC workPosIns  @posID INT,  @posName TEXT,  @posDescription TEXT,  @posInDepartment INT  AS  BEGIN  IF @posID IS NULL PRINT N'posID can not be NULL'  ELSE BEGIN  IF EXISTS(SELECT posID FROM WorkPosition WHERE posID = @posID) PRINT N'Dupplicate posID'  ELSE BEGIN  IF NOT EXISTS(SELECT departmentID FROM Department WHERE departmentID = @posInDepartment)  PRINT N'Foreign key not found'  ELSE INSERT INTO WorkPosition VALUES(@posID, @posName, @posDescription, @posInDepartment)  END  END  END  GO |

1. Trigger:

|  |
| --- |
| -- Before  CREATE TRIGGER aftDeleteWorkPos  ON WorkPosition  INSTEAD OF DELETE  AS  DECLARE  @posID INT,  @posName VARCHAR(256),  @posDescription VARCHAR(256),  @posInDepartment INT  SELECT @posID = deleted.posID FROM deleted  BEGIN TRAN  DELETE FROM Taking WHERE posIDTake = @posID  DELETE FROM WorkPosition WHERE posID = @posID  COMMIT  GO |

|  |
| --- |
| -- Before  CREATE TRIGGER bfEmailInsert  ON PersonEmailAddress  INSTEAD OF INSERT  AS  DECLARE  @profileNumber INT,  @emailAddress VARCHAR(256)  SELECT @profileNumber = inserted.profileNumber FROM inserted  SELECT @emailAddress = inserted.emailAddress FROM inserted  BEGIN TRAN  IF @profileNumber IS NULL OR @emailAddress IS NULL  BEGIN  PRINT N'Primary key cannot be NULL'  ROLLBACK  END  ELSE  IF NOT EXISTS(SELECT profileNumber FROM Person WHERE profileNumber = @profileNumber)  BEGIN  PRINT N'No one has that profile number.'  ROLLBACK  END  ELSE  BEGIN  INSERT INTO PersonEmailAddress VALUES(@profileNumber, @emailAddress)  COMMIT  END  GO |

|  |
| --- |
| -- After  CREATE TRIGGER aftInsertBorrowRecord ON BorrowRecord  FOR INSERT  AS  DECLARE  @borrowRecordID INT,  @infraID INT,  @numberOfItem INT  SELECT @borrowRecordID = inserted.borrowID FROM inserted  BEGIN  SELECT @numberOfItem = numberOfItem, @infraID = infraIDInclude  FROM Including  WHERE borrowIDInclude = @borrowRecordID  UPDATE Infrastructure  SET numberOfAvailable -= @numberOfItem  WHERE infraID = @infraID  END  GO |

1. Thủ tục chứa câu SQL:

|  |
| --- |
| CREATE PROC getPhoneNumberByEmail  @email VARCHAR(256)  AS  BEGIN  DECLARE  @profileNumber INT  SELECT @profileNumber = profileNumber FROM PersonEmailAddress WHERE emailAddress = @email  SELECT phoneNumber FROM PersonPhoneNumber WHERE profileNumber = @profileNumber  END  GO |

|  |
| --- |
| -- Count how many person just use only one phone number  CREATE PROC numberOfPersonUseJustOneNumber  AS  BEGIN  SELECT COUNT(numberOfPhoneNumber) AS numberOfPersonUseJustOneNumber FROM  (  SELECT COUNT(phoneNumber) AS numberOfPhoneNumber FROM  (SELECT firstName, lastName, phoneNumber, Person.profileNumber  FROM  Person INNER JOIN PersonPhoneNumber  ON Person.profileNumber = PersonPhoneNumber.profileNumber) AS subsub  GROUP BY profileNumber  HAVING COUNT(phoneNumber) = 1) AS SUB  END  GO |

1. Hàm:

|  |
| --- |
| ---- Count how many man or woman employee in the whole team--------  CREATE FUNCTION sexCount  ( @sex BIT )  RETURNS INT  AS  BEGIN  DECLARE  @result INT  IF @sex = 1 SELECT @result = temp FROM (SELECT COUNT(gender) AS temp FROM Person WHERE gender = 1) AS sub  ELSE IF @sex = 0 SELECT @result = temp FROM (SELECT COUNT(gender) AS temp FROM Person WHERE gender = 0) AS sub  ELSE RETURN -1  RETURN @result  END |

|  |
| --- |
|  |

1. Giao diện ứng dụng và các hình ảnh minh hoạ:

**Thành viên 2:**

**Họ tên: Châu Thành Đạt**

**MSSV: 1710934**

1. Thủ tục insert dữ liệu:

Thủ tục chèn dữ liệu vào bảng Department có kiểm tra khoá chính NULL hoặc đã tồn tại hay không

|  |
| --- |
| CREATE PROC UP\_InsertDepartment  @departmentID INT,  @departmentName TEXT,  @departmentDescription TEXT  AS  BEGIN  IF @departmentID IS NULL  PRINT N'Khoa chinh NULL'  ELSE IF EXISTS(SELECT departmentID FROM Department WHERE departmentID = @departmentID)  PRINT N'Khoa chinh da ton tai'  ELSE  INSERT INTO Department VALUES(@departmentID, @departmentName, @departmentDescription)  END  GO |

1. Trigger:

Before trigger không cho phép xoá thông tin trong bảng Taking của thành viên thuộc phòng ban Administrator

|  |
| --- |
| CREATE TRIGGER UTG\_DeleteTaking  ON Taking  FOR DELETE  AS  BEGIN  DECLARE @Count INT  SELECT @Count = COUNT(\*)  FROM DELETED  WHERE DELETED.posIDTake >= 9 AND DELETED.posIDTake <= 14  IF @Count > 0 ROLLBACK TRAN  END  GO |

After trigger cập nhật số tiền quỹ trong bảng Fund khi thêm một khoản chi trong bảng ExpeditureCategory

|  |
| --- |
| CREATE TRIGGER UTG\_InsertExpeditureCategory  ON ExpeditureCategory  FOR INSERT  AS  BEGIN  DECLARE @ExpeditureMoney MONEY  SELECT @ExpeditureMoney = INSERTED.expeditureMoney FROM INSERTED  UPDATE Fund  SET currentBudget -= @ExpeditureMoney  WHERE fundID = 0  END  GO |

1. Thủ tục chứa câu SQL:

Thủ tục nhận vào tên chức vụ và trả về tên của những người có chức vụ đó

|  |
| --- |
| CREATE PROC UP\_SelectWhoTakingPosition  @posName VARCHAR(256)  AS  BEGIN  DECLARE  @posID INT  SELECT @posID = posID FROM WorkPosition WHERE posName = @posName  SELECT lastName, middleName, firstName  FROM Person  WHERE profileNumber IN (SELECT profileNumberTake  FROM Taking  WHERE posIDTake = @posID)  ORDER BY firstName  END  GO |

|  |
| --- |
| CREATE PROC UP\_Year  AS  BEGIN  SELECT COUNT(profileNumber), YEAR(dateOfBirth)  FROM (SELECT Person.profileNumber, Person.dateOfBirth, Taking.posIDTake  FROM Person INNER JOIN Taking  ON Person.profileNumber = Taking.profileNumberTake) AS JoinTable  GROUP BY YEAR(dateOfBirth)  ORDER BY COUNT(profileNumber) DESC  END  GO |

1. Hàm:

|  |
| --- |
| -- Return age of profileNumber --  CREATE FUNCTION UF\_Age (@profileNumber INT)  RETURNS INT  AS  BEGIN  DECLARE @Age INT, @Year DATE  IF @profileNumber IS NULL OR @profileNumber < 0  SET @Age = 0  ELSE  BEGIN  SELECT @Year = dateOfBirth FROM Person WHERE profileNumber = @profileNumber  SET @Age = Year(getdate()) - Year(@Year)  END  RETURN @Age  END |

|  |
| --- |
|  |

1. Giao diện ứng dụng và các hình ảnh minh hoạ:

**Thành viên 3:**

**Họ tên: Nguyễn Ngọc Duy Phong**

**MSSV: 1712617**

1. Thủ tục chèn dữ liệu:

|  |
| --- |
| -- Insert/add person phone number  CREATE PROC addPersonPhoneNumber  @profileNumber int,  @phoneNumber varchar(11)  AS  BEGIN  IF EXISTS(SELECT \* FROM Person WHERE profileNumber = @profileNumber)  BEGIN  IF NOT EXISTS(SELECT \* FROM PersonPhoneNumber WHERE phoneNumber = @phoneNumber)  INSERT INTO PersonPhoneNumber(profileNumber, phoneNumber)  VALUES (@profileNumber, @phoneNumber);  ELSE PRINT N'Duplicate Phone Number'  END  END  GO |

1. Trigger:

|  |
| --- |
|  |

|  |
| --- |
|  |

1. Thủ tục chứa câu SQL:

|  |
| --- |
| -- Find person First/Last name and print out national ID number and TickLab ID number  CREATE PROCEDURE printPersonIDByFirstNameOrLastName  @name varchar(256)  AS  BEGIN  DECLARE @likename varchar(256);  SET @likename='%'+@name;  SELECT firstName+' '+lastName AS name,nationalIDNumber,rfidNumber  FROM Person,TickLabIDCard  WHERE (firstName LIKE @likename OR lastName LIKE @likename) and Person.profileNumber = TickLabIDCard.profileNumber  ORDER BY name  RETURN @@ROWCOUNT  END |

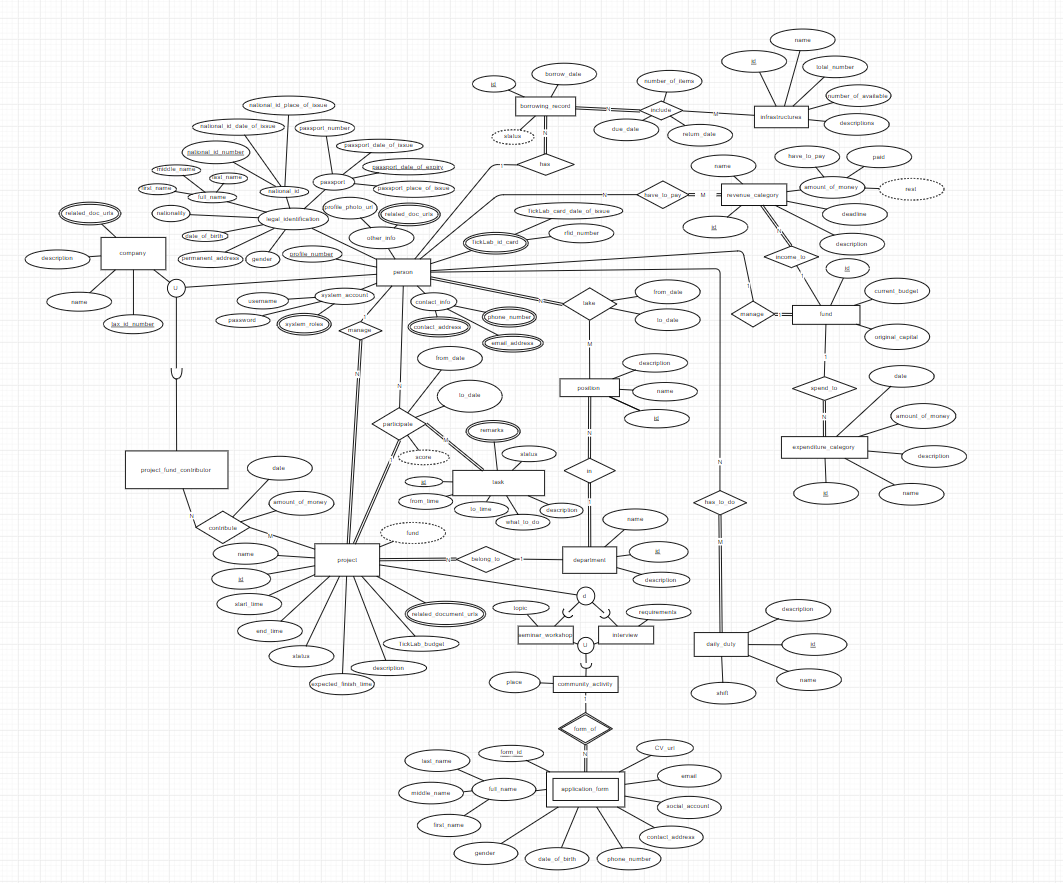
|  |
| --- |
|  |

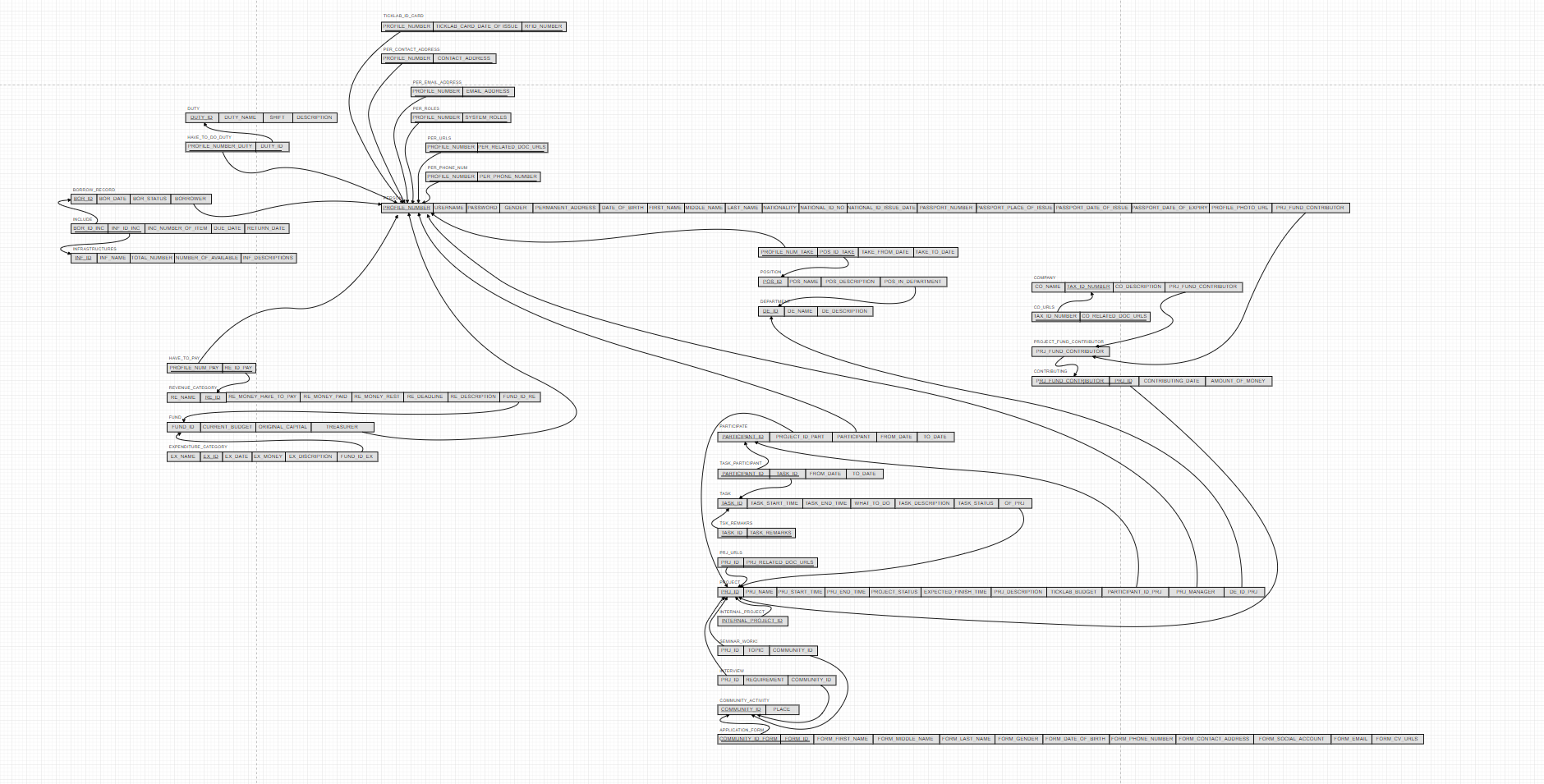
1. Hàm:

|  |
| --- |
| -- Get age person by profileNumber  CREATE FUNCTION AGE  ( @profileNumber int )  RETURN int  AS  BEGIN  DECLARE @age int  if NOT EXISTS(SELECT \* FROM Person WHERE profileNumber=@profileNumber) RETURN -1;  ELSE  SELECT @age = year(getdate())-year((SELECT dateOfBirth FROM Person WHERE profileNumber=@profileNumber));  RETURN @age  END |

|  |
| --- |
|  |

1. Giao diện ứng dụng và hình ảnh minh hoạ:
2. Phụ lục
3. Báo cáo bài tập lớn số 1





File: ****softwareSpecifications.txt****

* **Kha**: I, II, VI, VII
* **Dat**: III, IV, V, VIII

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  | | --- | |  | |  | SYSTEM Specifications | |  | I. Employee profile management | |  | 1. Features: | |  | - Add new profile | |  | - Edit profile | |  | + Change information | |  | + Delete profile | |  | - View profile | |  | - Profile change request | |  | 2. Users: | |  | - System admin: | |  | + Add new, edit, and view profiles | |  | - Others users: View and request changes, the changes must be accepted | |  | by the System Admin | |  | 3. Data: | |  | - Person: | |  | + Profile number | |  | + Legal identification | |  | -- Nationality | |  | -- Fullname | |  | ++ First name | |  | ++ Middle name | |  | ++ Last name | |  | -- Date of birth | |  | -- Gender | |  | -- Permanent address | |  | -- National ID information | |  | ++ ID number | |  | ++ Date of issue | |  | ++ Place of issue | |  | -- Passport (Option) | |  | ++ Passport number | |  | ++ Date of issue | |  | ++ Date of expiry | |  | ++ Place of issue | |  | + Local identification: TickLab ID card (many, in the case of reissuance) | |  | -- RFID number | |  | -- Date of issue | |  | + Contact information | |  | -- Phone number (many): Mark some phone number as recently used | |  | -- Contact address (many): Mark some address as recently used | |  | -- Email address (many): Mark some email addr as recently used | |  | + Other information | |  | -- Profile photo url | |  | -- Related documents links (many) | |  | + System login account | |  | -- username; | |  | -- password; | |  | -- System role: System admin/ Director/ Manager... (would be described later) | |  | + Project participation record list | |  | + Position took place (list by time) | |  | II. Organizational structure management | |  | 1. Features | |  | - Add/edit/remove department info | |  | - View department info | |  | - Manage department human resouce | |  | + Assign/remove employee to/from department | |  | + Enroll department position | |  | 2. Users | |  | - System admin: full permission | |  | - Other user: View department info | |  | 3. Data | |  | - Department information | |  | + ID | |  | + Name | |  | + Description | |  | + Department human resouce (list of position) | |  | + Department projects (list of projects by time) | |  | - Position: | |  | + ID | |  | + Name | |  | + Descriptions | |  | + Employee | |  | + Start Date | |  | + Stop date | |  | III. Finance management | |  | 1. Features: | |  | - View current budget | |  | - View finance statistic | |  | - Revenue categories management | |  | + Add revenue categories | |  | + Edit revenue categories | |  | + View revenue categories | |  | + Add/remove/edit employee | |  | -- Add/remove/edit by person | |  | -- Add/remove/edit by list | |  | + Add/remove records | |  | + Print receipt | |  | - Expenditure categories management | |  | + Add expenditure categories | |  | + Edit expenditure categories | |  | + View expenditure categories | |  | + Add/remove records | |  | + Print receipt | |  | 2. Users: | |  | - Admin users: | |  | + Full authority | |  | - Other users: | |  | + View budget | |  | + View revenue and expenditure categories | |  | + View personal finance statistic | |  | 3. Data: | |  | - Revenue category: | |  | + Employee (many) | |  | + ID | |  | + Name | |  | + Amount of money | |  | -- Rest | |  | -- Paid | |  | -- Have to pay | |  | + Deadline | |  | + Status | |  | + Description | |  | - Expenditure categories: | |  | + ID | |  | + Name | |  | + Amount of money | |  | + Description | |  | + Date | |  | - Fund | |  | + Employee | |  | + ID | |  | + Current budget | |  | + Original capital | |  | IV. Projects management | |  | 1. Features: | |  | - Create/Edit/Remove project | |  | - Add/Remove employees to project | |  | - Assign task | |  | - Update task log | |  | - Start/Stop/Pause project | |  | - View project (description, progress, participants, time...) | |  | - Participants performance evaluation | |  | 2. Users: | |  | - System admin: Create project | |  | - Project Manager: Edit/Remove, Add/Remove employee, assign tasks, evaluate participant's performance | |  | - Participants: update task log | |  | - Other user: View | |  | 3. Data: | |  | - Project | |  | + ID | |  | + Name | |  | + Descriptions | |  | + Project owner (department) | |  | + Project Manager (employee) | |  | + Start time | |  | + Expected finish time | |  | + Status (in progress/ finished/ cancelled/ postponed) | |  | + End time | |  | + Participants (list of employees who participate in the project) | |  | + Timeline (list of timeline note) | |  | -- FromTime | |  | -- ToTime | |  | -- What to do | |  | -- Status (Have not started, in progress, or finished) | |  | + Task | |  | -- ID | |  | -- Employee list (one or many) | |  | -- Description | |  | -- Begin time | |  | -- Remark | |  | -- Status | |  | + Project fund: | |  | -- TickLab's budget | |  | -- List of contributions: | |  | ++ Origin of contribution (partner company/ partner person...) | |  | ++ Amount of money | |  | ++ Date of contribution | |  | + Related document links (a list of urls); | |  | - Project participation records | |  | + Employee (who) | |  | + Project | |  | + Start time | |  | + Endtime | |  | + TaskList | |  | + Score | |  | V. Infrastructures management | |  | 1. Features: | |  | - View list of infrastructures | |  | - View status of infrastructures | |  | - Borrow/Return requests | |  | - Borrow infrastructures | |  | + Send borrow request | |  | + Accept borrow request | |  | - Return infrastructures | |  | + Confirm the borrower has returned | |  | - Add/Remove/Edit infrastructures | |  | 2. Users: | |  | - Admin users: | |  | + Add/Remove/Edit infrastructures | |  | + View status of infrastructures | |  | + Accept borrow request | |  | + Confirm the borrower has returned | |  | - Other users: | |  | + View status of infrastructures | |  | + Send borrow/return request | |  | 3. Data: | |  | - Infrastructures | |  | + ID (for kind of item only) | |  | + Name | |  | + Total number of items | |  | + Number of available | |  | + Descriptions | |  | - Borrow record | |  | + Employee | |  | + ID | |  | + Item | |  | + Status | |  | + Date | |  | VI. Local user services | |  | 1. Features: | |  | - Duty register system | |  | + Create/Remove/Edit a duty | |  | -- Daily | |  | -- Weekly | |  | -- Monthly | |  | + View duty | |  | + Join/Out shift | |  | + Swap shift with another user | |  | -- Send swap request | |  | -- Accept/Reject swap request | |  | + Move to an available shift | |  | - Internal event | |  | + Create/Remove an internal event | |  | + View event | |  | + Join/Out event | |  | + Note | |  | 2. Users: | |  | - Admin users | |  | + Create/Remove a service | |  | - Other users | |  | + View services | |  | + Join/Out services | |  | + Swap shift with another user | |  | + Move to an available shift | |  | 3. Data: | |  | - Daily duty | |  | + ID | |  | + Name | |  | + Shift | |  | + Descriptions | |  | VII. Community services | |  | 1. Features: | |  | - Create/Edit/Remove event (workshop/seminar/interview) | |  | - Register event | |  | - Create interview schedule | |  | - Manage applicants | |  | 2. Users: | |  | - Admin users: Create/Edit/Remove event | |  | - Guess: Register event | |  | 3. Data: | |  | - Seminar/Workshop | |  | + ID | |  | + Name | |  | + Speaker | |  | + Subject | |  | + Time | |  | -- Begin time | |  | -- End time | |  | + Place | |  | - Application form | |  | + Form ID | |  | + Name | |  | -- First name | |  | -- Middle name | |  | -- Last name | |  | + Date of birth | |  | + Gender | |  | + Email | |  | + Phone number | |  | + Contact address | |  | + Social account | |  | + CV link | |

File: ****entitiesList.txt****

* **Phong**: read the specification and make a list of entities/relationship

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  | | --- | | I. Entities: | |  | 1. Person | |  | - Profile number (key) | |  | - Legal identification | |  | + Nationality | |  | + Fullname | |  | -- First name | |  | -- Middle name | |  | -- Last name | |  | + Birthday | |  | + Gender | |  | + Permanent address | |  | + National ID information | |  | -- ID number (key) | |  | -- Date of issue | |  | -- Place of issue | |  | + Passport (optional) | |  | -- Passport number | |  | -- Date of issue | |  | -- Date of expiry | |  | -- Place of issue | |  | - TickLab ID card (multivalue) | |  | + RFID number | |  | + Date of issue | |  | - Contact infomation | |  | + Phone number (multivalue) | |  | + Contact address (multivalue) | |  | + Email address (multivatlue) | |  | - Other infomation | |  | + Profile photo url | |  | + Related document urls (multivalue) | |  | - System login account | |  | + Username | |  | + Password | |  | + System role (multivalue) | |  | 2. Project | |  | - ID (key) | |  | - Name | |  | - Description | |  | - Start time | |  | - End time | |  | - Expected finish time | |  | - Status | |  | - Fund (derived) | |  | - TickLab's budget | |  | - Related document urls (multivalue) | |  | 3. Department | |  | - ID (key) | |  | - Name | |  | - Description | |  | 4. Position | |  | - ID (key) | |  | - Name | |  | - Description | |  | 5. Task | |  | - ID (key) | |  | - Status | |  | - Description | |  | - Remark (multivalue) | |  | - From time | |  | - To time | |  | - What to do | |  | 6. Company | |  | - Tax ID number (key) | |  | - Name | |  | - Description | |  | - Related document urls (multivalue) | |  | 7. Infrastuctures | |  | - ID (key) | |  | - Name | |  | - Description | |  | - Total number of items | |  | - Number of available (derived) | |  | 8. Project fund contributor | |  | 9. Daily duty | |  | - ID (key) | |  | - Name | |  | - Description | |  | - Shift (multivalue) | |  | 10. Revenue category | |  | - ID (key) | |  | - Name | |  | - Description | |  | - Employee | |  | - Amount of money | |  | + Paid | |  | + Have to pay | |  | + Rest (derived) | |  | - Deadline | |  | 11. Expenditure category | |  | - ID (key) | |  | - Name | |  | - Description | |  | - Amount of money | |  | - Time | |  | 12. Borrowing record | |  | - ID (key) | |  | - Borrow date | |  | - Status (derived) | |  | 13. Comunity activity | |  | - Place | |  | 14. Seminar workshop | |  | - Topic | |  | 15. Interview | |  | - Requirement | |  | 16. Application form | |  | - Form ID (partial key) | |  | - Full name | |  | + First name | |  | + Middle name | |  | + Last name | |  | - Gender | |  | - Date of birth | |  | - Phone number | |  | - CV url | |  | - Email | |  | - Social account | |  | - Contact address | |  | 17. Fund | |  | - ID (key) | |  | - Current budget | |  | - Original capital | |  | II. Relationships: | |  | 1. Manage project | |  | - Entity: | |  | + Person | |  | -- Mandatory: no | |  | -- Cardinality: 1 | |  | + Project | |  | -- Mandatory: yes | |  | -- Cardinality: N | |  | 2. Manage fund | |  | - Entity: | |  | + Person | |  | -- Mandatory: no | |  | -- Cardinality: 1 | |  | + Fund | |  | -- Mandatory: yes | |  | -- Cardinality: 1 | |  | 3. Participate | |  | - Relationship attribute: | |  | + From date | |  | + To date | |  | + Score (derived) | |  | - Entity: | |  | + Person | |  | -- Mandatory: no | |  | -- Cardinality: N | |  | + Project | |  | -- Mandatory: yes | |  | -- Cardinality: 1 | |  | + Task | |  | -- Mandatory: yes | |  | -- Cardinality: M | |  | 4. Contribute | |  | - Relationship attribute: | |  | + Amount of money | |  | + Date | |  | - Entity: | |  | + Project fund contributor: | |  | -- Mandatory: no | |  | -- Cardinality: N | |  | + Project: | |  | -- Mandatory: no | |  | -- Cardinality: M | |  | 5. Belong to | |  | - Entity: | |  | + Project | |  | -- Mandatory: yes | |  | -- Cardinality: N | |  | + Department | |  | -- Mandatory: no | |  | -- Cardinality: 1 | |  | 6. In | |  | - Entity: | |  | + Department | |  | -- Mandatory: yes | |  | -- Cardinality: 1 | |  | + Position | |  | -- Mandatory: yes | |  | -- Cardinality: N | |  | 7. Take | |  | - Relationship attribute: | |  | + From date | |  | + To date | |  | - Entity: | |  | + Person | |  | -- Mandatory: yes | |  | -- Cardinality: N | |  | + Position | |  | -- Mandatory: no | |  | -- Cardinality: M | |  | 8. Have to pay | |  | - Entity: | |  | + Person | |  | -- Mandatory: no | |  | -- Cardinality: N | |  | + Revenue category | |  | -- Mandatory: yes | |  | -- Cardinality: M | |  | 9. Has | |  | - Entity: | |  | + Person | |  | -- Mandatory: no | |  | -- Cardinality: 1 | |  | + Borrowing record | |  | -- Mandatory: yes | |  | -- Cardinality: N | |  |  | |  | 10. Include | |  | - Relationship attribute: | |  | + Number of items | |  | + Due date | |  | + Return date | |  | - Entity: | |  | + Infrastructure | |  | -- Mandatory: no | |  | -- Cardinality: M | |  | + Borrowing record | |  | -- Mandatory: yes | |  | -- Cardinality: N | |  | 11. Income to | |  | - Entity: | |  | + Revenue category | |  | -- Mandatory: yes | |  | -- Cardinality: N | |  | + Fund | |  | -- Mandatory: no | |  | -- Cardinality: 1 | |  | 12. Spend to | |  | - Entity: | |  | + Fund | |  | -- Mandatory: no | |  | -- Cardinality: 1 | |  | + Expenditure category | |  | -- Mandatory: yes | |  | -- Cardinality: N | |  | 13. Form of (identifying) | |  | - Entity: | |  | + Community activity | |  | -- Mandatory: no | |  | -- Cardinality: 1 | |  | + Application form (weak) | |  | -- Mandatory: yes | |  | -- Cardinality: N | |  | 14. Has to do | |  | - Entity: | |  | + Person | |  | -- Mandatory: no | |  | -- Cardinality: N | |  | + Borrowing record | |  | -- Mandatory: yes | |  | -- Cardinality: M | |

1. Source code chương trình
2. Bảng phân công nhiệm vụ cho phần chung và bài tập lớn 1

**Phần chung**

Kha + Phong: Generate data and SQL code for inserting into tables

Đạt: Write SQL code for creating tables and create indexes

**Bài tập lớn 1**

Vẽ ERD: Kha, Đạt

Vẽ Mapping: Đạt