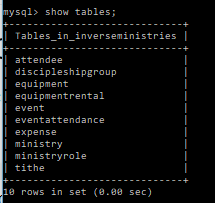
Phase 4 – SQL and Load Scripts

1. Scripts to create the Database Objects (DDL)
   1. Create the schema
      1. CREATE SCHEMA inverseministries;
   2. Create the Discipleship Group table
      1. CREATE TABLE DiscipleshipGroup (DGID int, City varchar(45), MeetingDayOfWeek varchar(45), PRIMARY KEY (DGID));
   3. Create the Attendee table
      1. CREATE TABLE Attendee (AttendeeID int, Name varchar(45), DateOfBirth date, PhoneNumber varchar(45), EmailAddress varchar(45), CityOfResidence varchar(45), MemberStatus tinyint, Salary float, DGID int, PRIMARY KEY (AttendeeID), FOREIGN KEY (DGID) REFERENCES DiscipleshipGroup(DGID));
   4. Create the Tithe table
      1. CREATE TABLE Tithe (TitheID int, TitheDate date, TitheAmount float, TitheType varchar(45), AttendeeID int, PRIMARY KEY (TitheID), FOREIGN KEY (AttendeeID) REFERENCES Attendee(AttendeeID));
   5. Create the Expense table
      1. CREATE TABLE Expense (ExpenseID int, ExpenseAmount float, Location varchar(45), Purpose varchar(45), PRIMARY KEY (ExpenseID), AttendeeID int, FOREIGN KEY (AttendeeID) REFERENCES Attendee(AttendeeID));
   6. Create the Event table
      1. CREATE TABLE Event (EventID int, EventName varchar(45), EventLeader varchar(45), StartDate date, EndDate date, EventBudget float, Location varchar(45), EventType varchar(45), PRIMARY KEY (EventID));
   7. Create the Event Attendance table
      1. CREATE TABLE EventAttendance (AttendanceType varchar(45), EventID int, AttendeeID int, PRIMARY KEY (EventID, AttendeeID), FOREIGN KEY (EventID) REFERENCES Event(EventID), FOREIGN KEY (AttendeeID) REFERENCES Attendee(AttendeeID));
   8. Create the Ministry table
      1. CREATE TABLE Ministry (MinistryID int, MinistryName varchar(45), MinistryLeader varchar(45), MinistryBudget float, PRIMARY KEY (MinistryID));
   9. Create the Ministry table
      1. CREATE TABLE MinistryRole (Role varchar(45), AttendeeID int, MinistryID int, PRIMARY KEY (AttendeeID, MinistryID), FOREIGN KEY (AttendeeID) REFERENCES Attendee(AttendeeID), FOREIGN KEY (MinistryID) REFERENCES Ministry(MinistryID));
   10. Create the Equipment table
       1. CREATE TABLE Equipment (EquipmentID int, Name varchar(45), Cost float, PRIMARY KEY (EquipmentID));
   11. Create the Equipment Rental table
       1. CREATE TABLE EquipmentRental (CheckoutDate date, ReturnDate date, MinistryID int, EquipmentID int, PRIMARY KEY (MinistryID, EquipmentID), FOREIGN KEY (MinistryID) REFERENCES Ministry(MinistryID), FOREIGN KEY (EquipmentID) REFERENCES Equipment(EquipmentID));
   12. Screenshot of all tables after DDL



1. Scripts to populate the Database Tables (DML)
   1. Add some discipleship groups
      1. INSERT INTO DiscipleshipGroup (DGID, City, MeetingDayOfWeek)

VALUES (1, ‘Diamond Bar’, ‘Thursday’);

* + 1. INSERT INTO DiscipleshipGroup (DGID, City, MeetingDayOfWeek)

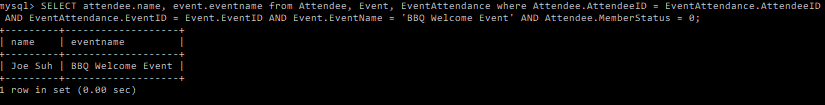
VALUES (2, ‘La Habra’, ‘Wednesday’);

* + 1. INSERT INTO DiscipleshipGroup (DGID, City, MeetingDayOfWeek)

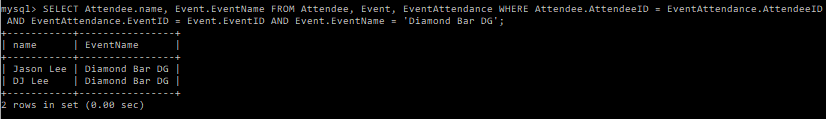
VALUES (3, ‘Brea’, ‘Tuesday’);

* 1. Add some attendees to the church
     1. INSERT INTO Attendee (AttendeeID, Name, DateOfBirth, PhoneNumber, EmailAddress, CityOfResidence, MemberStatus, DGID) VALUES (1, ‘Jason Lee’, ‘1985-05-28’, ‘714-111-1111’, ‘jasonlee@gmail.com’, ‘La Habra’, 1, 1);
     2. INSERT INTO Attendee (AttendeeID, Name, DateOfBirth, PhoneNumber, EmailAddress, CityOfResidence, MemberStatus, DGID) VALUES (2, ‘Jun Chung’, ‘1993-10-05’, ‘714-222-2222’, ‘jchung@gmail.com’, ‘La Habra’, 1, 3);
     3. INSERT INTO Attendee (AttendeeID, Name, DateOfBirth, PhoneNumber, EmailAddress, CityOfResidence, MemberStatus) VALUES (4, ‘Joe Suh’, ‘1982-11-02’, ‘714-333-3333’, ‘jsuh@gmail.com’, ‘Fullerton’, 0);
     4. INSERT INTO Attendee (AttendeeID, Name, DateOfBirth, PhoneNumber, EmailAddress, CityOfResidence, MemberStatus, DGID) VALUES (5, ‘DJ Lee’, ‘1991-2-20’, ‘909-777-7777’, ‘djlee@gmail.com’, ‘Diamond Bar’, 1, 1);
  2. Create an event called ‘BBQ Welcome Event’ to welcome newcomers to the church
     1. INSERT INTO Event (EventID, EventName, EventLeader, StartDate, EndDate, EventBudget, Location, EventType) VALUES (1, ‘BBQ Welcome Event’, ‘Patrick Cho’, ‘2017-12-2’, ‘2017-12-2’, 300, ‘Craig Regional Park’, ‘Outreach’);
     2. INSERT INTO EventAttendance (EventID, AttendeeID) VALUES (1, 1);
     3. INSERT INTO EventAttendance (EventID, AttendeeID) VALUES (1, 4);
  3. Create an event called ‘Diamond Bar DG’ for the last week of November. Diamond Bar DG usually meets on Thursdays.
     1. INSERT INTO Event (EventID, EventName, EventLeader, StartDate, EndDate, EventBudget, Location, EventType) VALUES (2, ‘Diamond Bar DG, ‘Jason Lee’, ‘2017-11-30’, ‘2017-11-30’, ‘Patricks House’, ‘DG Meeting’);
     2. INSERT INTO EventAttendance (EventID, AttendeeID) VALUES (2, 1);
     3. INSERT INTO EventAttendance (EventID, AttendeeID) VALUES (2, 5);
  4. Create a praise team ministry and add members to that team.
     1. INSERT INTO Ministry (MinistryID, MinistryName, MinistryLeader, MinistryBudget) VALUES (1, ‘Praise Team’, ‘Jason Lee’, 500);
     2. INSERT INTO MinistryRole (Role, AttendeeID, MinistryID) VALUES (‘Leader’, 1, 1);
     3. INSERT INTO MinistryRole (Role, AttendeeID, MinistryID) VALUES (‘Guitarist’, 2, 1);
  5. Create a program team ministry and add members to that team.
     1. INSERT INTO Ministry (MinistryID, MinistryName, MinistryLeader, MinistryBudget) VALUES (2, ‘Program Team’, ‘DJ Lee’, 200);
     2. INSERT INTO MinistryRole (Role, AttendeeID, MinistryID) VALUES (‘Leader’, 5, 2);
  6. Add all the tithes for the 11/26/17 Sunday service
     1. INSERT INTO Tithe (TitheID, TitheDate, TitheAmount, TitheType, AttendeeID) VALUES (1, ‘2017-11-26’, 300, ‘Check’, 5);
     2. INSERT INTO Tithe (TitheID, TitheDate, TitheAmount, TitheType) VALUES (2, ‘2017-11-26’, 20, ‘Cash’);
  7. Add all expenses pertaining to retreat purposes
     1. INSERT INTO Expense (ExpenseID, ExpenseAmount, Location, Purpose, AttendeeID) VALUES (1, 50, ‘Albertsons’, ‘Retreat Snacks’, 5);
  8. Add members that have salaries (i.e. pastors, staff, etc.)
     1. INSERT INTO Attendee (AttendeeID, Name, DateOfBirth, PhoneNumber, EmailAddress, CityOfResidence, MemberStatus, Salary, DGID) VALUES (3, ‘Patrick Cho’, ‘1980-07-15’, ‘909-123-1234’, ‘pcho@gmail.com’, ‘Diamond Bar’, 1, 50000, 1);
  9. Create an event for praise practice on 12/1/17
     1. INSERT INTO Event (EventID, EventName, EventLeader, StartDate, EndDate, Location, EventType) VALUES (3, ‘Praise practice’, ‘Jason Lee’, ‘2017-12-01’, ‘2017-12-01’, ‘Main Sanctuary’, ‘Practice’);
     2. INSERT INTO EventAttendance (EventID, AttendeeID) VALUES (3, 1);
     3. INSERT INTO EventAttendance (EventID, AttendeeID) VALUES (3, 2);

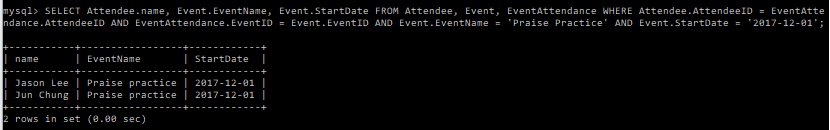
1. Sample queries and output for functional test cases
   1. List everyone that attended the ‘BBQ Welcome Event’ who were not already members of the church
      1. SELECT Attendee.name, Event.EventName FROM Attendee, Event, EventAttendance WHERE Attendee.AttendeeID = EventAttendance.AttendeeID AND EventAttendance.EventID = Event.EventID AND Event.EventName = ‘BBQ Welcome Event’ AND Attendee.MemberStatus = 0



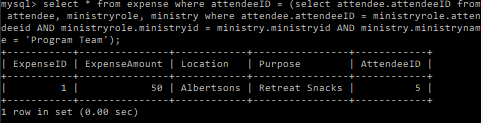
* 1. List members who attended ‘Diamond Bar DG’ this past Thursday on 11/30
     1. SELECT Attendee.name, Event.EventName FROM Attendee, Event, EventAttendance WHERE Attendee.AttendeeID = EventAttendance.AttendeeID AND EventAttendance.EventID = Event.EventID AND Event.EventName = ‘Diamond Bar DG’;



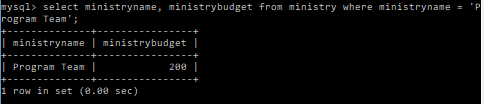
* 1. List members who attended praise practice on 12/1/17
     1. SELECT Attendee.name, Event.EventName, Event.StartDate FROM Attendee, Event, EventAttendance WHERE Attendee.AttendeeID = EventAttendance.AttendeeID AND EventAttendance.EventID = Event.EventID AND Event.EventName = ‘Praise Practice’ AND Event.StartDate = ‘2017-12-01’;



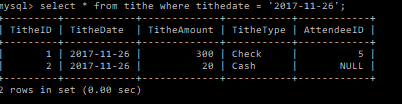
* 1. Show an expense performed by a member of the program team along with the budget allocated to the program team in order for someone to write a reimbursement
     1. SELECT \* FROM Expense WHERE AttendeeID = (SELECT Attendee.AttendeeID FROM Attendee, MinistryRole, Ministry WHERE Attendee.AttendeeID = MinistryRole.AttendeeID AND MinistryRole.MinistryID = Ministry.MinistryID AND Ministry.MinistryName = ‘Program Team’);



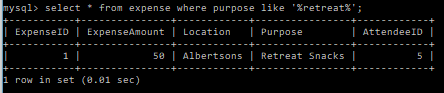
* + 1. SELECT MinistryName, MinistryBudget FROM Ministry WHERE MinistryName = ‘Program Team’;



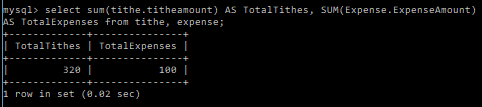
* 1. List all the tithes collected on Sunday 11/26/17
     1. SELECT \* FROM Tithe WHERE TitheDate = ‘2017-11-26’;



* 1. List all the expenses performed for retreat
     1. SELECT \* FROM Expense WHERE Purpose LIKE ‘%retreat%’;



* 1. Show the total amount of tithing and expenses along with all attendees that have a salary in order to process salaries
     1. SELECT SUM(Tithe.TitheAmount) AS TotalTithes, SUM(Expense.ExpenseAmount) AS TotalExpenses, From Tithe, Expense;



* + 1. SELECT Name, Salary FROM Attendee WHERE Salary > 0;

