**PHASE 3: SUBMISSION CHECKLIST/SIGNOFF SHEET**

**GROUP#:** 1 **GROUP NAME:** YOUTHPROGRAM database

Deliverables:

* [Revised] Description of the organization
* [Revised] ER diagram with min/max specifications
* [Revised] Constraints not in ER diagram
* [Revised] Relational Schema with Referential Integrity
* [Revised] Queries with brief description
* Completed ORM Implementation

Group Assessment

* Group Status Report

We have each reviewed the contents of this deliverable.

Phase Leader Rachel Gordon \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Phase Recorder Kadin Suga \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Phase Checker Max Simmer \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Technical Advisor David Sheleru \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Technical Advisor Taylor Fischer \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**1. Introduction**

The Centers for Social and Educational Development (CSED), a Midwest-based nonprofit, seeks to inspire people of all ages to reach their full potential through professional, leadership, cultural, and faith-based educational programs. As part of its mission, CSED supports a network of centers that offer youth programming, including series of ongoing classes as well as special events including workshops, service opportunities, and retreats. Mary Morgan, a staff member at CSED and expert user, has generously agreed to serve as a contact person and to provide input on the proposed database.

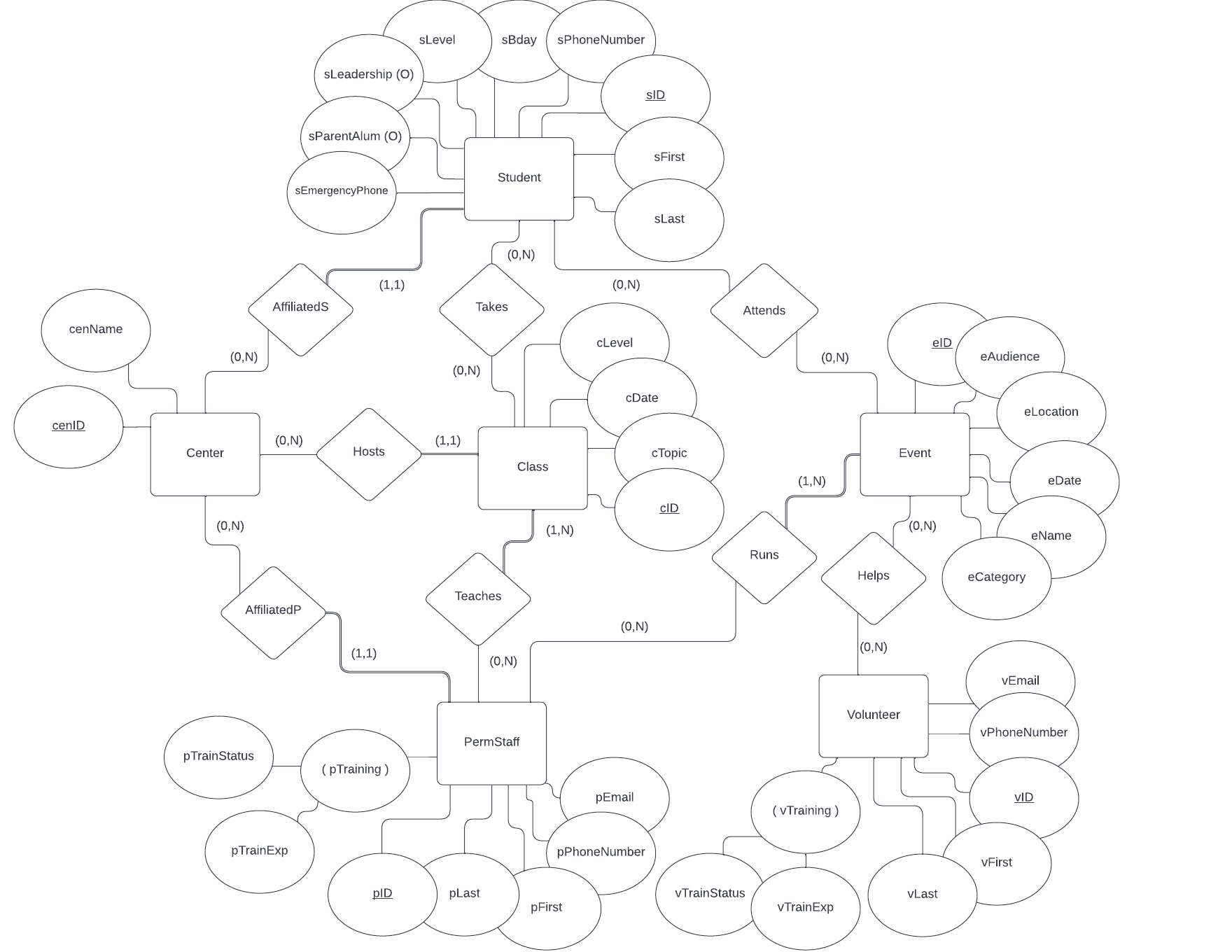
**2. Description of the Organization**

The Youth Program Database (YOUTHPROGRAM) stores information about the youth programs offered through CSED and participation at such events. The YOUTHPROGRAM Database contains data regarding the student attendees and their participation, the classes and events offered, and the staff and volunteers assisting with the various programs:

* Students that come to youth programs are included in the database. Each student has their student ID, first name, last name, phone number, birthday, email, and an emergency contact phone number stored in the database. Additional categories include a student’s level in school (e.g. high school or college), whether a student is involved in a leadership group, and whether a student’s parent(s) is an alum of CSED programs. Each student is mainly affiliated with one center, but a center can be affiliated with many students.
* Centers are physical branch locations within the CSED network that serve as hubs for classes. Each center has a name and a unique ID. Each center can host many different series of class sessions.
* A class session can only be hosted by one center. Each class session has an ID, a series level indicating the intended student audience, the date the session was held, and the topic presented. There is no maximum class size, and students can take multiple sessions (of the same series). Each session is taught by one or more staff members.
* Events are organized for students to attend, in categories such as workshops, service opportunities, and retreats. Each event is recorded with an event ID, name, date, location, event category, and intended student audience. Students can attend multiple special events, and there is not a set attendance capacity. Events are run by one or more permanent staff, often with the help of volunteers.
* Permanent staff can teach multiple class sessions or run multiple events. Each staff member is affiliated with one center, with no limit to the number of staff per center. Each permanent staff member is recorded with a unique ID, first name, last name, email, and phone number. Staff members must maintain satisfactory training, so training status and the date of training expiration are also recorded.
* Volunteers can help with special events. Each volunteer is recorded with an ID, first name, last name, email, and phone number. The training status and the next training expiration date are also recorded for volunteers.

**3. ER Diagram**

The figure below shows the ER diagram of the Youth Program Database (YOUTHPROGRAM).



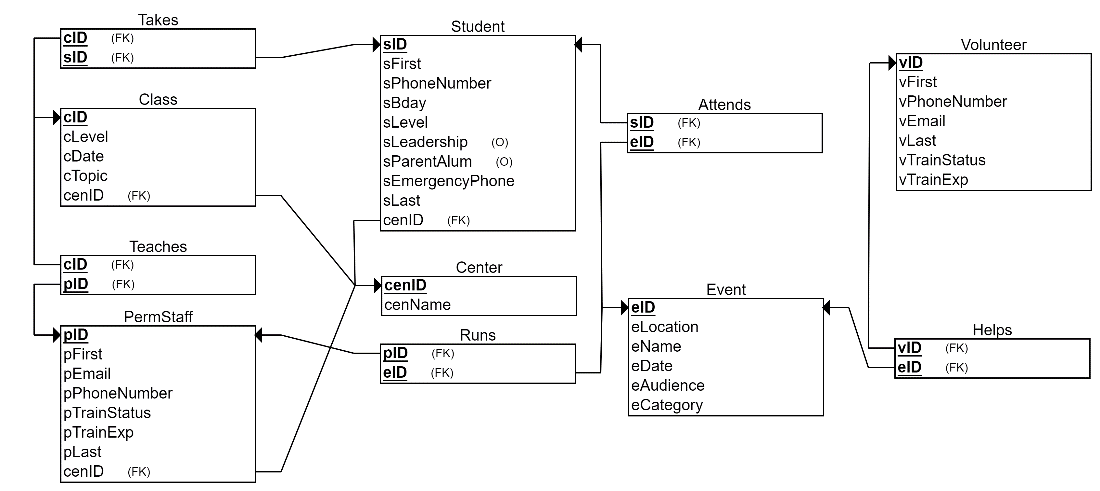
**4. ER Diagram Uncaptured Constraints**

The following is a list of constraints that are not captured by the ER diagram of YOUTHPROGRAM:

* Each event must have at least 2 adults running/present at the event. The minimum can be met with 2 permanent staff, or 1 permanent staff member and 1 volunteer.
* Students should only attend class sessions that match their student level.
* Student level in school (sLevel) can be either ms (middle school), hs (high school), co (college), ot (other)
* Training status of permanent staff (pTrainStatus) and volunteers (vTrainStatus) can either be not started, initiated, good, flag, or failed.
* Event categories (eCategory) can be either workshop, service, retreat, or other.

**5. Relational Schema**

This section provides the relational schema with referential integrity and the relational table details.



**5.1 Relational Schema with Referential Integrity**

Student(sID, sFirst, sLast, sPhoneNumber, sBday, sLevel, sLeadership, sParentAlum, sEmergencyPhone, cenID)

foreign key (cenID) references Center (cenID)rachel

Attends(sID,eID)

foreign key (sID) references Student (sID)

foreign key (eID) references Event (eID)

Class(cID, cLevel, cDate, cTopic, cenName)

foreign key (cenID) references Center(cenID)

Takes(cID,sID)

foreign key (cID) references Class(cID)

foreign key (sID) references Student(sID)

Teaches(cID, pID)

foreign key (cID) references Class(cID)

foreign key (pID) references PermStaff(pID)

Volunteer (vID, vFirst, vPhoneNumber, vEmail, vLast, vTrainStatus, vTrainExp)

Helps (vID, eID)

foreign key (vID) references Volunteer(vID)

foreign key (eID) references Event(eID)

Center(cenID, cenName)

Event (eID, eLocation, eName, eDate, eAudience, sCategory)

PermStaff(pID, pFirst, pEmail, pPhoneNumber, pTrainStatus, pTrainExp, pLast, cenID)

foreign key (cenID) references Center(cenID)

Runs(pID, eID)

foreign key (pID) references PermStaff(pID)

foreign key (eID) references Event(eID)

**5.2 Relational Table Details**

The relational schema given in Section 5.1 was mapped into the following tables in the YOUTHPROGRAM database. Primary keys have been underlined. Tables that have multiple attributes underlined represent composite keys.

|  |  |  |
| --- | --- | --- |
| **Table Name** | **Attribute** | **Description** |
| student | sID | Unique ID for identifying students |
| sFirst | Student first name |
| sLast | Student last name |
| sBday | Student birth date |
| sPhoneNumber | Student phone number |
| sLevel | Student’s level in school (ms, hs, col, grad, other) |
| sLeadership | Whether or not a student is involved in a leadership group |
| sParentAlum | Whether or not a student has a parent that is an alum of the organization |
| sEmergencyPhone | Emergency phone number for contacting a student’s guardian |
| cenID | Name of center student is affiliated with. Foreign key (cenID) references Center (cenID) |
| permStaff | pID | Unique ID for identifying permanent staff members |
| pFirst | Permanent staff member first name |
| pLast | Permanent staff member last name |
| pEmail | Permanent staff member email address |
| pPhoneNumber | Permanent staff member phone number |
| pTrainStatus | Training status of permanent staff member (not started, initiated, good, flag, failed) |
| pTrainExp | Date when permanent staff member’s training expires |
| cenID | Name of center staff member is primarily affiliated with. Foreign key (cenID) references Center (cenID) |
| class | cID | Unique ID for identifying a class session |
| cLevel | Class series name indicating the intended student level for the class |
| cDate | Dates the class is offered |
| cTopic | Topic covered in the class |
| cenID | Name of center that hosts the class. Foreign key (cenID) references Center (cenID) |
| event | eID | Unique ID for identifying one-time special event |
| eName | Name of one-time special event |
| eDate | Date event was held |
| eAudience | Intended student audience for the event |
| eLocation | Location where the event was held |
| eCategory | Category the event falls under (workshop, service, retreat, other) |
| volunteer | vID | Unique ID for identifying volunteers |
| vFirst | Volunteer first name |
| vLast | Volunteer last name |
| vEmail | Volunteer email address |
| vPhoneNumber | Volunteer phone number |
| vTrainStatus | Training status of volunteer (not started, initiated, good, flag, failed) |
| vTrainExp | Date when volunteer’s training expires |
| center | cenID | Unique ID for each individual branch location. |
| cenName | Name of physical branch location where classes and some events are held and with which students and staff are affiliated |
| attends | sID | Unique ID for identifying students. Foreign key (sID) references Student(sID) |
| eID | Unique ID for identifying one-time special event. Foreign key (eID) references Event(eID) |
| takes | cID | Unique ID for identifying a class. Foreign key (cID) references Class (cID) |
| sID | Unique ID for identifying students. Foreign key (sID) references Student (sID) |
| teaches | cID | Unique ID for a class session. Foreign key (cID) references Class (cID) |
| pID | Unique ID for permanent staff member teaching the course. Foreign key (pID) references PermStaff (pID) |
| runs | pID | Unique ID for permanent staff member running the event. Foreign key (pID) references PermStaff (pID) |
| eID | Unique ID for identifying one-time special event. Foreign key (eID) references Event (eID) |
| helps | vID | Unique ID for identifying volunteers. Foreign key (vID) references Volunteer (vID) |
| eID | Unique ID for identifying one-time special event. Foreign key (eID) references Event (eID) |

**6. Queries**

The following table summarizes the queries in the EMPTRAINING Database.

|  |  |  |  |
| --- | --- | --- | --- |
| **Query Name** | **Description** | **Output** | **Relations**  **Accessed** |
| VolunteerHelp (David) | For each event, list the volunteer’s ID, volunteer’s name, training status, and phone number. Only print out the results if the volunteer assigned to the event has participated in less than 2 events. Order by Volunteer’s name in ascending order. | VID  VFirst  VTrainStatus  VPhoneNumb  Count of event | Helps  Volunteer  Event |
| AvgStudentsClassLevel (Rachel) | Find the average number of students attending classes grouped by class level and center and ordered by center name. | CenID  CLevel  Avg\_students\_per\_class | Student  Takes  Class  Center |
| LeadershipGroupClassAttendance (Taylor) | For each center at each class level, print the number of distinct students attending classes and the number of those students who are part of the leadership group. Include the center name and class level. Order by center name and then class level. | cenName,  cLevel,  totalStudentCount,  leadershipStudentCount | center,  student,  takes,  class |
| NumberOfEvents (Max) | Find each student that attends more than one event and list the specific events they attend. Include student first and last name as well as their student ID. Order by numberOfEvents > 1 in descending order. | sID,  sFirst,  sLast,  numberOfEvents,  attendedEvents | student,  event,  attends |
| LowestStudent EventAttendance (Kadin) | Lowest student attendance where a student attends one class and attends an event. List the number of students attending one class and the number of events attended with first name in ascending order. | firstname,  lastname,  classattend, eventattend | Student,  Event,  Class,  Attends,  Takes |

**7. DDL + DML + SQL (Including output screenshots)**

The following is an SQL definition of the tables for the EMPTRAINING Database.

DDL:

CREATE TABLE Center

(

cenName VARCHAR NOT NULL,

cenID VARCHAR(2) NOT NULL,

PRIMARY KEY (cenID)

);

CREATE TABLE Student(

sID VARCHAR PRIMARY KEY,

sFirst VARCHAR NOT NULL,

sLast VARCHAR NOT NULL,

sPhoneNumber VARCHAR(12) NOT NULL,

sBday DATE NOT NULL,

sLevel VARCHAR NOT NULL

check (sLevel in ('co', 'hs', 'ms', 'ot')),

sLeadership VARCHAR,

sParentAlum VARCHAR,

sEmergencyPhone VARCHAR NOT NULL,

cenID VARCHAR(2) NOT NULL,

FOREIGN KEY (cenID) REFERENCES Center(cenID));

create table Class(

cID VARCHAR,

cLevel VARCHAR(2) NOT NULL,

cDate DATE NOT NULL,

cTopic VARCHAR NOT NULL,

cenID varchar(2) NOT NULL,

PRIMARY KEY (cID),

FOREIGN KEY (cenID) REFERENCES center(cenID));

CREATE TABLE PermStaff

(

pID VARCHAR NOT NULL,

pFirst VARCHAR NOT NULL,

pLast VARCHAR NOT NULL,

pEmail VARCHAR NOT NULL,

pPhoneNumber VARCHAR(12) NOT NULL,

pTrainStatus VARCHAR NOT NULL

check (pTrainStatus in (‘not started’, ‘initiated’, ‘good’, ‘flag’, ‘failed’)),

pTrainExp DATE,

cenID VARCHAR(2) NOT NULL,

PRIMARY KEY (pID),

FOREIGN KEY (cenID) REFERENCES Center(cenID)

);

CREATE TABLE Volunteer

(

vID VARCHAR NOT NULL,

vFirst VARCHAR NOT NULL,

vLast VARCHAR NOT NULL,

vPhoneNumber VARCHAR(12) NOT NULL,

vEmail VARCHAR NOT NULL,

vTrainStatus VARCHAR

check (vTrainStatus in (‘not started’, ‘initiated’, ‘good’, ‘flag’, ‘failed’)),

vTrainExp DATE,

PRIMARY KEY (vID)

);

CREATE TABLE Event

(

eID VARCHAR NOT NULL,

eLocation VARCHAR NOT NULL,

eName VARCHAR NOT NULL,

eDate DATE NOT NULL,

eAudience CHAR(2) NOT NULL,

eCategory VARCHAR NOT NULL

check (eCategory in (‘workshop’, ‘service’, ‘retreat’, ‘other’)),

PRIMARY KEY (eID)

);

CREATE TABLE Takes

(

cID VARCHAR NOT NULL,

sID VARCHAR NOT NULL,

PRIMARY KEY (cID, sID),

FOREIGN KEY (cID) REFERENCES Class(cID),

FOREIGN KEY (sID) REFERENCES Student(sID)

);

CREATE TABLE Teaches(

cID VARCHAR(10) NOT NULL,

pID VARCHAR NOT NULL,

PRIMARY KEY(cID, pID),

FOREIGN KEY (cID) REFERENCES Class (cID),

FOREIGN KEY (pID) REFERENCES PermStaff (pID)

);

create table runs

(pID varchar,

eID varchar,

Primary key(pID,eID),

Foreign key (pID) references PermStaff(pID),

Foreign key (eID) references Event(eID)

);

CREATE TABLE attends (

sID VARCHAR NOT NULL,

eID VARCHAR NOT NULL,

PRIMARY KEY (sID, eID),

FOREIGN KEY (sID) REFERENCES student(sID),

FOREIGN KEY (eID) REFERENCES event(eID)

);

CREATE TABLE HELPS (vID VARCHAR Not Null, eID VARCHAR Not Null,

Primary Key(vID,eID),

Foreign Key(vID) References Volunteer(vID),

Foreign Key(eID) References Event(eID));

DDL:

INSERT INTO Center (cenID, cenName)

VALUES

('EL', 'Elms'), ('FG', 'Forest Glen'), ('GR', 'Grove'), ('LD', 'Lindell'), ('PW', 'Petawa'), ('SB', 'Shellbourne'), ('SL', 'Sherlake'), ('SH', 'Southold');

INSERT INTO Student (sID,sFirst,sLast,sPhoneNumber,sBday,sLevel,sLeadership,sParentAlum,sEmergencyPhone,cenID)

VALUES

('LeKoch','Leah','Koch','331-336-8319',to\_date('01/05/05', 'MM/DD/YY'),'co','yes',NULL,'738-495-2160','EL'),

('SaJohnson','Sarah','Johnson','847-874-3158',to\_date('03/26/01', 'MM/DD/YY'),'co',NULL,NULL,'624-809-7315','EL'),

('RyWilliams','Ryan','Williams','331-924-7168',to\_date('09/15/01', 'MM/DD/YY'),'co',NULL,'yes','583-672-9140','EL'),

('EmDavis','Emily','Davis','312-565-0293',to\_date('06/02/02', 'MM/DD/YY'),'co','yes','yes','459-218-6370','EL'),

('DaMartinez','Daniel','Martinez','618-947-1162',to\_date('07/14/03', 'MM/DD/YY'),'co',NULL,NULL,'726-493-5180','EL'),

('OlSmith','Olivia','Smith','708-479-2157',to\_date('12/13/01', 'MM/DD/YY'),'co','yes',NULL,'362-840-1597','EL'),

('MaTaylor','Matthew','Taylor','773-466-6844',to\_date('08/24/01', 'MM/DD/YY'),'co','yes','yes','950-286-4713','EL'),

('AvBrown','Ava','Brown','847-647-9798',to\_date('10/05/04', 'MM/DD/YY'),'co','yes',NULL,'817-496-3250','EL'),

('AlLee','Alexander','Lee','331-805-3950',to\_date('11/29/02', 'MM/DD/YY'),'co',NULL,NULL,'204-736-9815','EL'),

('ChAnderson','Chloe','Anderson','312-384-6007',to\_date('01/12/05', 'MM/DD/YY'),'hs',NULL,NULL,'675-892-3140','GR'),

('BeWilson','Ben','Wilson','847-256-5819',to\_date('02/05/06', 'MM/DD/YY'),'hs','yes',NULL,'384-762-9150','GR'),

('MiClark','Mia','Clark','618-184-7893',to\_date('03/20/05', 'MM/DD/YY'),'hs','yes','yes','123-450-8976','GR'),

('SaWhite','Samuel','White','331-798-6579',to\_date('04/08/05', 'MM/DD/YY'),'hs','yes','yes','892-731-6450','GR'),

('ChHall','Charlotte','Hall','618-670-1205',to\_date('05/16/06', 'MM/DD/YY'),'hs',NULL,'yes','657-234-1890','GR'),

('AmRodriguez','Amelia','Rodriguez','224-889-4751',to\_date('06/25/06', 'MM/DD/YY'),'hs','yes',NULL,'941-582-7306','GR'),

('HeAdams','Henry','Adams','312-431-9277',to\_date('07/09/05', 'MM/DD/YY'),'hs','yes',NULL,'392-687-4150','GR'),

('NoMitchell','Noah','Mitchell','630-675-4148',to\_date('08/14/08', 'MM/DD/YY'),'hs',NULL,NULL,'546-123-7890','GR'),

('LiGreen','Lily','Green','630-675-4148',to\_date('09/02/06', 'MM/DD/YY'),'hs',NULL,NULL,'289-456-1073','GR'),

('JaMoore','James','Moore','773-971-9466',to\_date('10/18/06', 'MM/DD/YY'),'hs',NULL,NULL,'810-946-2357','GR'),

('SoParker','Sofia','Parker','224-416-4139',to\_date('11/27/05', 'MM/DD/YY'),'hs','yes','yes','631-487-5290','GR'),

('LiWilson','Liam','Wilson','847-795-4328',to\_date('08/15/09', 'MM/DD/YY'),'ms',NULL,NULL,'407-895-3621','GR'),

('WiHarris','William','Harris','708-778-0943',to\_date('11/03/09', 'MM/DD/YY'),'ms','yes',NULL,'263-750-8149','GR'),

('IsRodriguez','Isabella','Rodriguez','618-215-1850',to\_date('03/22/10', 'MM/DD/YY'),'ms','yes','yes','859-234-1760','GR'),

('JaThomas','Jackson','Thomas','630-914-3122',to\_date('07/09/11', 'MM/DD/YY'),'ms','yes',NULL,'512-406-7983','GR'),

('MiSmith','Michael','Smith','312-385-6097',to\_date('10/12/10', 'MM/DD/YY'),'ms',NULL,NULL,'746-931-5208','GR'),

('OlBrown','Olivia','Brown','485-726-9134',to\_date('01/05/09', 'MM/DD/YY'),'ms',NULL,'yes','372-916-5840','GR'),

('WiTaylor','William','Taylor','629-384-7150',to\_date('05/18/10', 'MM/DD/YY'),'ms',NULL,NULL,'981-734-6025','GR'),

('AvAnderson','Ava','Anderson','314-902-8765',to\_date('09/30/10', 'MM/DD/YY'),'ms','yes',NULL,'451-820-3796','GR'),

('BeLee','Benjamin','Lee','572-913-4680',to\_date('12/08/09', 'MM/DD/YY'),'ms',NULL,NULL,'617-482-3905','GR'),

('ZoCooper','Zoe','Cooper','718-649-2053',to\_date('06/08/08', 'MM/DD/YY'),'hs','yes','yes','586-042-3179','PW'),

('LoWard','Logan','Ward','109-785-4326',to\_date('08/24/05', 'MM/DD/YY'),'hs','yes',NULL,'295-607-3841','PW'),

('ViMurphy','Victoria','Murphy','587-364-1029',to\_date('03/17/08', 'MM/DD/YY'),'hs','yes',NULL,'730-149-8256','PW'),

('JaStewart','Jackson','Stewart','236-458-9701',to\_date('10/29/06', 'MM/DD/YY'),'hs',NULL,NULL,'165-429-7380','PW'),

('EmRivera','Emma','Rivera','890-612-3475',to\_date('01/14/08', 'MM/DD/YY'),'hs',NULL,NULL,'493-176-2805','PW'),

('DaCarter','Daniel','Carter','456-210-7893',to\_date('07/05/07', 'MM/DD/YY'),'hs','yes','yes','714-085-2369','PW'),

('AdReed','Addison','Reed','721-098-4365',to\_date('12/25/05', 'MM/DD/YY'),'hs',NULL,NULL,'369-814-5270','PW'),

('AnRoss','Andrew','Ross','304-917-5826',to\_date('05/02/06', 'MM/DD/YY'),'hs','yes',NULL,'872-301-4965','PW'),

('ElYoung','Ella','Young','638-574-2901',to\_date('02/09/06', 'MM/DD/YY'),'hs','yes',NULL,'504-972-6183','PW'),

('CaMitchell','Caleb','Mitchell','549-302-6871',to\_date('11/12/07', 'MM/DD/YY'),'hs',NULL,'yes','610-934-8275','PW'),

('LiWood','Lily','Wood','765-219-8340',to\_date('09/22/08', 'MM/DD/YY'),'hs',NULL,NULL,'283-650-1794','PW'),

('LiScott','Liam','Scott','823-705-6491',to\_date('04/18/05', 'MM/DD/YY'),'hs',NULL,NULL,'971-245-8063','PW'),

('AbMorris','Abigail','Morris','193-284-5067',to\_date('06/29/06', 'MM/DD/YY'),'hs','yes',NULL,'376-289-5401','SB'),

('ZaParker','Zach','Parker','408-576-3921',to\_date('08/14/07', 'MM/DD/YY'),'hs','yes','yes','219-674-8503','SB'),

('ErHayes','Ericka','Hayes','796-125-8340',to\_date('03/07/07', 'MM/DD/YY'),'hs',NULL,NULL,'465-312-7890','SB'),

('EmSullivan','Emily','Sullivan','678-149-2053',to\_date('10/08/05', 'MM/DD/YY'),'hs','yes','yes','850-129-3467','SB'),

('SoThompson','Sophia','Thompson','295-748-6130',to\_date('01/21/09', 'MM/DD/YY'),'hs',NULL,NULL,'792-438-5016','SB'),

('JaMorris','Jackson','Morris','437-289-6105',to\_date('07/19/06', 'MM/DD/YY'),'hs',NULL,'yes','136-724-5980','SB'),

('MiStewart','Mia','Stewart','180-725-9364',to\_date('12/11/05', 'MM/DD/YY'),'hs','yes',NULL,'408-917-6253','SB'),

('MaHernandez','Marie','Hernandez','610-245-7983',to\_date('05/26/07', 'MM/DD/YY'),'hs','yes',NULL,'567-812-4309','SB'),

('OwCarter','Owen','Carter','954-682-3017',to\_date('02/15/06', 'MM/DD/YY'),'hs',NULL,'yes','305-246-9817','SB'),

('JoPeterson','Joseph','Peterson','276-804-9135',to\_date('11/10/09', 'MM/DD/YY'),'hs','yes',NULL,'924-871-3560','SB'),

('NiJohnson','Nicholas','Johnson','542-970-1863',to\_date('09/05/08', 'MM/DD/YY'),'hs',NULL,'yes','738-019-4652','SB'),

('LuWilson','Luke','Wilson','832-167-4950',to\_date('04/10/07', 'MM/DD/YY'),'hs',NULL,NULL,'632-184-7905','SB'),

('BrMitchell','Brayden','Mitchell','691-237-8045',to\_date('02/14/09', 'MM/DD/YY'),'ms',NULL,NULL,'514-627-8309','SL'),

('NoClark','Nora','Clark','174-609-2385',to\_date('06/04/10', 'MM/DD/YY'),'ms','yes','yes','697-230-4851','SL'),

('JaLee','Jaxon','Lee','365-892-4701',to\_date('08/27/09', 'MM/DD/YY'),'ms','yes','yes','890-245-3617','SL'),

('LiNyugen','Lisa','Nyugen','972-830-1456',to\_date('11/11/10', 'MM/DD/YY'),'ms',NULL,NULL,'345-769-1820','SL'),

('DaMontgomery','David','Montgomery','536-478-9120',to\_date('04/07/09', 'MM/DD/YY'),'ms','yes',NULL,'271-398-4605','SL'),

('StHoffman','Stephanie','Hoffman','890-372-5146',to\_date('07/20/11', 'MM/DD/YY'),'ms',NULL,'yes','506-738-1294','SL'),

('StGuzman','Steven','Guzman','140-785-9623',to\_date('09/02/09', 'MM/DD/YY'),'ms','yes',NULL,'928-453-1760','SL'),

('BrRodgers','Brett','Rodgers','862-195-3470',to\_date('12/31/10', 'MM/DD/YY'),'ms','yes',NULL,'763-491-8250','SL'),

('LoShaw','Lorena','Shaw','429-083-6715',to\_date('03/01/09', 'MM/DD/YY'),'ms',NULL,NULL,'123-590-4678','SL'),

('SaGardner','Sara','Gardner','713-690-2458',to\_date('05/12/10', 'MM/DD/YY'),'ms',NULL,NULL,'419-587-2360','SL'),

('LoCunningham','Louise','Cunningham','807-912-3465',to\_date('06/27/06', 'MM/DD/YY'),'hs',NULL,'yes','874-062-9315','SL'),

('AnEllis','Andrew','Ellis','346-981-2570',to\_date('08/09/08', 'MM/DD/YY'),'hs','yes',NULL,'965-432-0781','SL'),

('ClHammond','Clara','Hammond','215-876-3490',to\_date('03/27/08', 'MM/DD/YY'),'hs','yes',NULL,'346-281-0975','SL'),

('TaOsborne','Tanya','Osborne','574-308-1629',to\_date('10/04/07', 'MM/DD/YY'),'hs',NULL,NULL,'283-961-4750','SL'),

('JoHopkins','Joey','Hopkins','968-145-7320',to\_date('01/30/06', 'MM/DD/YY'),'hs',NULL,NULL,'507-984-2163','SL'),

('EmSchneider','Emma','Schneider','341-205-7968',to\_date('07/08/08', 'MM/DD/YY'),'hs','yes','yes','619-827-5304','SL'),

('ArJensen','Arthur','Jensen','528-467-9130',to\_date('12/15/05', 'MM/DD/YY'),'hs',NULL,NULL,'652-140-3897','SL'),

('CaFlores','Carroll','Flores','629-781-0435',to\_date('05/15/06', 'MM/DD/YY'),'hs','yes',NULL,'430-789-2165','SL'),

('CeCollier','Cecil','Collier','195-634-8720',to\_date('02/23/07', 'MM/DD/YY'),'hs','yes',NULL,'728-391-0546','SL'),

('LeMorris','Leslie','Morris','478-921-6053',to\_date('11/21/06', 'MM/DD/YY'),'hs','yes','yes','296-741-5380','SL'),

('JoJohnson','John','Johnson','573-629-1480',to\_date('09/13/08', 'MM/DD/YY'),'hs',NULL,'yes','981-276-4305','SL'),

('PaWhite','Paul','White','204-789-6135',to\_date('04/25/05', 'MM/DD/YY'),'hs',NULL,NULL,'543-219-6780','SL'),

('LeYates','Lela','Yates','436-982-5701',to\_date('06/02/07', 'MM/DD/YY'),'hs','yes',NULL,'205-964-1873','SL'),

('NoWarner','Noah','Warner','681-093-2475',to\_date('08/31/06', 'MM/DD/YY'),'hs','yes',NULL,'486-321-0957','SH'),

('LeRuiz','Lena','Ruiz','903-817-6425',to\_date('03/04/08', 'MM/DD/YY'),'hs',NULL,NULL,'317-289-5046','SH'),

('FaCasey','Faith','Casey','372-546-8019',to\_date('10/01/07', 'MM/DD/YY'),'hs',NULL,NULL,'950-428-3671','SH'),

('MaHudson','Maria','Hudson','615-794-0283',to\_date('11/10/03', 'MM/DD/YY'),'co','yes','yes','760-348-1295','SH'),

('JaStevenson','Jake','Stevenson','287-415-0936',to\_date('04/02/05', 'MM/DD/YY'),'co','yes',NULL,'621-594-0837','SH'),

('CaBlake','Carter','Blake','549-861-2730',to\_date('09/14/01', 'MM/DD/YY'),'co',NULL,NULL,'397-186-2450','SH'),

('MaRoberts','Maria','Roberts','209-781-3645',to\_date('02/08/04', 'MM/DD/YY'),'co',NULL,NULL,'185-762-3490','SH'),

('MaMartinez','Marta','Martinez','836-241-9705',to\_date('05/20/03', 'MM/DD/YY'),'co',NULL,NULL,'349-275-6108','SH'),

('OlBurns','Olivia','Burns','794-083-1625',to\_date('08/17/05', 'MM/DD/YY'),'co',NULL,'yes','702-918-4356','SH'),

('CaGrant','Casey','Grant','523-746-8901',to\_date('12/31/02', 'MM/DD/YY'),'co','yes','yes','518-439-2760','SH'),

('KaFleming','Karla','Fleming','461-208-9375',to\_date('07/05/04', 'MM/DD/YY'),'co','yes','yes','864-207-9315','SH'),

('RoHunter','Rosalie','Hunter','972-431-6085',to\_date('03/25/01', 'MM/DD/YY'),'co',NULL,NULL,'294-158-7603','SH'),

('HaMcguire','Harper','Mcguire','134-205-7896',to\_date('10/13/03', 'MM/DD/YY'),'co','yes',NULL,'637-819-2045','SH'),

('LoMiller','Logan','Miller','795-843-2610',to\_date('01/03/05', 'MM/DD/YY'),'co','yes',NULL,'970-582-3164','SH'),

('LuGomez','Lucia','Gomez','680-324-7915',to\_date('06/09/02', 'MM/DD/YY'),'co','yes','yes','541-283-0976','SH'),

('KeFisher','Kendra','Fisher','215-479-3608',to\_date('11/18/04', 'MM/DD/YY'),'co',NULL,NULL,'326-845-7190','SH'),

('MaBecker','Mary','Becker','769-804-2315',to\_date('04/14/01', 'MM/DD/YY'),'co',NULL,NULL,'187-420-9365','SH'),

('SoMorgan','Sofia','Morgan','301-874-5926',to\_date('09/28/03', 'MM/DD/YY'),'co',NULL,NULL,'913-684-2705','SH');

INSERT INTO Class (cID, cLevel, cDate, cTopic, cenID) VALUES

('ELco101122', 'co', to\_date('10/11/22', 'MM/DD/YY'), 'Cheerfulness', 'EL'),

('ELco110822', 'co', to\_date('11/08/22', 'MM/DD/YY'), 'Working Well', 'EL'),

('ELco121322', 'co', to\_date('12/13/22', 'MM/DD/YY'), 'Friendship', 'EL'),

('ELco011023', 'co', to\_date('01/10/23', 'MM/DD/YY'), 'Fortitude', 'EL'),

('ELco021423', 'co', to\_date('02/14/23', 'MM/DD/YY'), 'Justice', 'EL'),

('ELco031423', 'co', to\_date('03/14/23', 'MM/DD/YY'), 'Temperance', 'EL'),

('ELco041123', 'co', to\_date('04/11/23', 'MM/DD/YY'), 'Prudence', 'EL'),

('ELco050923', 'co', to\_date('05/09/23', 'MM/DD/YY'), 'Leadership', 'EL'),

('GRhs091522', 'hs', to\_date('09/15/22', 'MM/DD/YY'), 'Friendship', 'GR'),

('GRhs100622', 'hs', to\_date('10/06/22', 'MM/DD/YY'), 'Courage', 'GR'),

('GRhs102022', 'hs', to\_date('10/20/22', 'MM/DD/YY'), 'Study', 'GR'),

('GRhs110322', 'hs', to\_date('11/03/22', 'MM/DD/YY'), 'Cheerfulness', 'GR'),

('GRhs111722', 'hs', to\_date('11/17/22', 'MM/DD/YY'), 'Honesty', 'GR'),

('GRhs120122', 'hs', to\_date('12/01/22', 'MM/DD/YY'), 'Leadership', 'GR'),

('GRhs011923', 'hs', to\_date('01/19/23', 'MM/DD/YY'), 'Service', 'GR'),

('GRhs020223', 'hs', to\_date('02/02/23', 'MM/DD/YY'), 'Prudence', 'GR'),

('GRhs021623', 'hs', to\_date('02/16/23', 'MM/DD/YY'), 'Justice', 'GR'),

('GRhs030223', 'hs', to\_date('03/02/23', 'MM/DD/YY'), 'Temperance', 'GR'),

('GRhs031623', 'hs', to\_date('03/16/23', 'MM/DD/YY'), 'Fortitude', 'GR'),

('GRhs040623', 'hs', to\_date('04/06/23', 'MM/DD/YY'), 'Respect', 'GR'),

('GRhs042023', 'hs', to\_date('04/20/23', 'MM/DD/YY'), 'Motivation', 'GR'),

('GRhs050423', 'hs', to\_date('05/04/23', 'MM/DD/YY'), 'Prayer', 'GR'),

('GRms091922', 'ms', to\_date('09/19/22', 'MM/DD/YY'), 'Goal Setting', 'GR'),

('GRms101722', 'ms', to\_date('10/17/22', 'MM/DD/YY'), 'Friendship', 'GR'),

('GRms112122', 'ms', to\_date('11/21/22', 'MM/DD/YY'), 'Courage', 'GR'),

('GRms121922', 'ms', to\_date('12/19/22', 'MM/DD/YY'), 'Study', 'GR'),

('GRms011623', 'ms', to\_date('01/16/23', 'MM/DD/YY'), 'Cheerfulness', 'GR'),

('GRms022023', 'ms', to\_date('02/20/23', 'MM/DD/YY'), 'Honesty', 'GR'),

('GRms032023', 'ms', to\_date('03/20/23', 'MM/DD/YY'), 'Service', 'GR'),

('GRms041023', 'ms', to\_date('04/10/23', 'MM/DD/YY'), 'Gratitude', 'GR'),

('PWhs091522', 'hs', to\_date('09/15/22', 'MM/DD/YY'), 'Friendship', 'PW'),

('PWhs100622', 'hs', to\_date('10/06/22', 'MM/DD/YY'), 'Courage', 'PW'),

('PWhs102722', 'hs', to\_date('10/27/22', 'MM/DD/YY'), 'Study', 'PW'),

('PWhs110322', 'hs', to\_date('11/03/22', 'MM/DD/YY'), 'Cheerfulness', 'PW'),

('PWhs111722', 'hs', to\_date('11/17/22', 'MM/DD/YY'), 'Honesty', 'PW'),

('PWhs120122', 'hs', to\_date('12/01/22', 'MM/DD/YY'), 'Leadership', 'PW'),

('PWhs011923', 'hs', to\_date('01/19/23', 'MM/DD/YY'), 'Service', 'PW'),

('PWhs020223', 'hs', to\_date('02/02/23', 'MM/DD/YY'), 'Prudence', 'PW'),

('PWhs021623', 'hs', to\_date('02/16/23', 'MM/DD/YY'), 'Justice', 'PW'),

('PWhs030223', 'hs', to\_date('03/02/23', 'MM/DD/YY'), 'Temperance', 'PW'),

('PWhs031623', 'hs', to\_date('03/16/23', 'MM/DD/YY'), 'Fortitude', 'PW'),

('PWhs040623', 'hs', to\_date('04/06/23', 'MM/DD/YY'), 'Respect', 'PW'),

('PWhs042023', 'hs', to\_date('04/20/23', 'MM/DD/YY'), 'Motivation', 'PW'),

('PWhs050423', 'hs', to\_date('05/04/23', 'MM/DD/YY'), 'Prayer', 'PW'),

('PWhs051823', 'hs', to\_date('05/18/23', 'MM/DD/YY'), 'Gratitude', 'PW'),

('SBhs093022', 'hs', to\_date('09/30/22', 'MM/DD/YY'), 'Friendship', 'SB'),

('SBhs102822', 'hs', to\_date('10/28/22', 'MM/DD/YY'), 'Courage', 'SB'),

('SBhs111822', 'hs', to\_date('11/18/22', 'MM/DD/YY'), 'Study', 'SB'),

('SBhs121622', 'hs', to\_date('12/16/22', 'MM/DD/YY'), 'Cheerfulness', 'SB'),

('SBhs012723', 'hs', to\_date('01/27/23', 'MM/DD/YY'), 'Leadership', 'SB'),

('SBhs022423', 'hs', to\_date('02/24/23', 'MM/DD/YY'), 'Service', 'SB'),

('SBhs033123', 'hs', to\_date('03/31/23', 'MM/DD/YY'), 'Respect', 'SB'),

('SBhs042823', 'hs', to\_date('04/28/23', 'MM/DD/YY'), 'Motivation', 'SB'),

('SBhs052623', 'hs', to\_date('05/26/23', 'MM/DD/YY'), 'Gratitude', 'SB'),

('SLhs101222', 'hs', to\_date('10/12/22', 'MM/DD/YY'), 'Friendship', 'SL'),

('SLhs102622', 'hs', to\_date('10/26/22', 'MM/DD/YY'), 'Study', 'SL'),

('SLhs110922', 'hs', to\_date('11/09/22', 'MM/DD/YY'), 'Cheerfulness', 'SL'),

('SLhs112322', 'hs', to\_date('11/23/22', 'MM/DD/YY'), 'Gratitude', 'SL'),

('SLhs121422', 'hs', to\_date('12/14/22', 'MM/DD/YY'), 'Courage', 'SL'),

('SLhs011123', 'hs', to\_date('01/11/23', 'MM/DD/YY'), 'Prudence', 'SL'),

('SLhs012523', 'hs', to\_date('01/25/23', 'MM/DD/YY'), 'Justice', 'SL'),

('SLhs020823', 'hs', to\_date('02/08/23', 'MM/DD/YY'), 'Temperance', 'SL'),

('SLhs022223', 'hs', to\_date('02/22/23', 'MM/DD/YY'), 'Fortitude', 'SL'),

('SLhs030823', 'hs', to\_date('03/08/23', 'MM/DD/YY'), 'Respect', 'SL'),

('SLhs032223', 'hs', to\_date('03/22/23', 'MM/DD/YY'), 'Leadership', 'SL'),

('SLhs041223', 'hs', to\_date('04/12/23', 'MM/DD/YY'), 'Service', 'SL'),

('SLhs041923', 'hs', to\_date('04/19/23', 'MM/DD/YY'), 'Prayer', 'SL'),

('SLms092622', 'ms', to\_date('09/26/22', 'MM/DD/YY'), 'Goal Setting', 'SL'),

('SLms102422', 'ms', to\_date('10/24/22', 'MM/DD/YY'), 'Study', 'SL'),

('SLms112822', 'ms', to\_date('11/28/22', 'MM/DD/YY'), 'Friendship', 'SL'),

('SLms121922', 'ms', to\_date('12/19/22', 'MM/DD/YY'), 'Courage', 'SL'),

('SLms013023', 'ms', to\_date('01/30/23', 'MM/DD/YY'), 'Cheerfulness', 'SL'),

('SLms022723', 'ms', to\_date('02/27/23', 'MM/DD/YY'), 'Honesty', 'SL'),

('SLms032723', 'ms', to\_date('03/27/23', 'MM/DD/YY'), 'Service', 'SL'),

('SLms042423', 'ms', to\_date('04/24/23', 'MM/DD/YY'), 'Respect', 'SL'),

('SHhs090722', 'hs', to\_date('09/07/22', 'MM/DD/YY'), 'Friendship', 'SH'),

('SHhs110222', 'hs', to\_date('11/02/22', 'MM/DD/YY'), 'Study', 'SH'),

('SHhs010423', 'hs', to\_date('01/04/23', 'MM/DD/YY'), 'Cheerfulness', 'SH'),

('SHhs030723', 'hs', to\_date('03/07/23', 'MM/DD/YY'), 'Gratitude', 'SH'),

('SHhs050223', 'hs', to\_date('05/02/23', 'MM/DD/YY'), 'Leadership', 'SH'),

('SHco101322', 'co', to\_date('10/13/22', 'MM/DD/YY'), 'Working Well', 'SH'),

('SHco111022', 'co', to\_date('11/10/22', 'MM/DD/YY'), 'Friendship', 'SH'),

('SHco120822', 'co', to\_date('12/08/22', 'MM/DD/YY'), 'Fortitude', 'SH'),

('SHco020223', 'co', to\_date('02/02/23', 'MM/DD/YY'), 'Justice', 'SH'),

('SHco030123', 'co', to\_date('03/01/23', 'MM/DD/YY'), 'Temperance', 'SH'),

('SHco040523', 'co', to\_date('04/05/23', 'MM/DD/YY'), 'Prudence', 'SH'),

('SHco050323', 'co', to\_date('05/03/23', 'MM/DD/YY'), 'Leadership', 'SH');

INSERT INTO PermStaff (pID, pFirst, pLast, pEmail, pPhoneNumber, pTrainStatus, pTrainExp, cenID)

VALUES

('MaSmith', 'Margaret', 'Smith', 'margaretsmith87@gmail.com', '773-432-1234', 'initiated', NULL, 'EL'),

('LiBennett', 'Liam', 'Bennett', 'liam.bennett\_123@yahoo.com', '815-123-4567', 'good', '2024-08-09', 'EL'),

('OlCarter', 'Olivia', 'Carter', 'oliviacarter22@hotmail.com', '773-234-5678', 'initiated', NULL, 'EL'),

('NoDavis', 'Noah', 'Davis', 'noah.davis2000@gmail.com', '773-432-1098', 'good', '2024-03-08', 'EL'),

('EmEvans', 'Emma', 'Evans', 'e.evansofficial@gmail.com', '815-345-6789', 'initiated', NULL, 'EL'),

('JaFisher', 'Jackson', 'Fisher', 'jackson.fisher\_99@yahoo.com', '773-876-5432', 'flag', '2023-07-02', 'EL'),

('AvGreene', 'Ava', 'Greene', 'ava.greene\_5678@hotmail.com', '815-987-6543', 'good', '2026-07-07', 'FG'),

('AiHayes', 'Aiden', 'Hayes', 'aidenh345@gmail.com', '815-789-0123', 'initiated', NULL, 'FG'),

('IsIngram', 'Isabella', 'Ingram', 'isabella.ingram@outlook.com', '773-890-1234', 'good', '2024-09-06', 'FG'),

('MaSmith1', 'Mary', 'Smith', 'm.smithofficial@gmail.com', '773-678-9012', 'initiated', NULL, 'LD'),

('LuJensen', 'Lucas', 'Jensen', 'lucas.jensen22@yahoo.com', '815-678-9012', 'good', '2024-07-25', 'LD'),

('MiKeller', 'Mia', 'Keller', 'mia.keller\_456@hotmail.com', '773-234-5678', 'initiated', NULL, 'LD'),

('EtLawson', 'Ethan', 'Lawson', 'ethan.lawson\_789@gmail.com', '773-890-1234', 'good', '2026-08-12', 'LD'),

('HaMitchell', 'Harper', 'Mitchell', 'harper.mitchell\_321@yahoo.com', '815-789-0987', 'failed', NULL, 'LD'),

('OlNelson', 'Oliver', 'Nelson', 'o.nelsonofficial@gmail.com', '815-345-6789', 'good', '2024-11-30', 'LD'),

('AmOwens', 'Amelia', 'Owens', 'amelia.owens\_22@hotmail.com', '773-234-5678', 'good', '2026-02-07', 'PW'),

('ElParker', 'Elijah', 'Parker', 'elijah.parker2001@yahoo.com', '773-678-9012', 'initiated', NULL, 'PW'),

('EvQuinn', 'Evelyn', 'Quinn', 'evelyn.quinnofficial@gmail.com', '815-678-9012', 'good', '2025-10-14', 'PW'),

('JaReynolds', 'James', 'Reynolds', 'james\_reynolds22@yahoo.com', '773-432-1098', 'good', '2024-10-12', 'PW'),

('AbStewart', 'Abigail', 'Stewart', 'abigail.stewart\_567@hotmail.com', '815-987-6543', 'initiated', NULL, 'PW'),

('BeTurner', 'Benjamin', 'Turner', 'benjamin\_turner@gmail.com', '773-876-5432', 'good', '2025-02-21', 'PW'),

('ChVaughn', 'Charlotte', 'Vaughn', 'c.vaughnofficial@yahoo.com', '815-123-4567', 'initiated', NULL, 'SB'),

('WiWallace', 'William', 'Wallace', 'william.wallace\_123@hotmail.com', '815-789-0123', 'good', '2023-12-03', 'SB'),

('ScYoung', 'Scarlett', 'Young', 's.youngofficial@gmail.com', '773-432-1098', 'failed', NULL, 'SB'),

('HeZimmerman', 'Henry', 'Zimmerman', 'h.zimmerman\_789@yahoo.com', '773-432-1234', 'good', '2025-11-27', 'SB'),

('GrHarrison', 'Grace', 'Harrison', 'graceharrisonofficial@gmail.com', '773-890-1234', 'initiated', NULL, 'SL'),

('AlDixon', 'Alexander', 'Dixon', 'alexander\_dixon22@hotmail.com', '815-123-4567', 'good', '2025-08-09', 'SL'),

('LiMason', 'Lily', 'Mason', 'lily.mason\_456@yahoo.com', '773-432-1098', 'flag', '2023-09-08', 'SL'),

('SaPorter', 'Samuel', 'Porter', 'samuel.porterofficial@gmail.com', '773-234-5678', 'good', '2024-02-18', 'SL'),

('ChReed', 'Chloe', 'Reed', 'chloe.reed\_22@yahoo.com', '815-345-6789', 'initiated', NULL, 'SL'),

('DaSanders', 'Daniel', 'Sanders', 'daniel.sanders\_123@hotmail.com', '773-678-9012', 'good', '2026-06-20', 'SL'),

('ZoWarren', 'Zoey', 'Warren', 'zoey.warrenofficial@gmail.com', '815-789-0123', 'good', '2025-07-22', 'SL'),

('CaHughes', 'Caleb', 'Hughes', 'caleb.hughes\_567@yahoo.com', '773-876-5432', 'initiated', NULL, 'SL'),

('ElRoss', 'Ella', 'Ross', 'ella\_ross22@hotmail.com', '815-987-6543', 'good', '2026-04-11', 'SH'),

('MaPowell', 'Mason', 'Powell', 'mason.powellofficial@gmail.com', '773-432-1234', 'flag', '2023-05-28', 'SH'),

('SoBennett', 'Sofia', 'Bennett', 'sofia.bennett\_321@yahoo.com', '815-789-0123', 'good', '2023-12-12', 'SH');

INSERT INTO Volunteer (vID, vFirst, vLast, vPhoneNumber, vEmail, vTrainStatus, vTrainExp)

VALUES

('V01', 'Jane', 'Angel', '333-444-5567', 'janeangel1@gmail.com', 'good', '2024-10-16'),

('V02', 'David', 'Smith', '111-224-3378', 'davidsmith@outlook.com', 'good', '2024-11-23'),

('V03', 'Angela', 'Precious', '208-447-5583', 'angelaprecious3@hotmail.com', 'good', '2024-06-04'),

('V04', 'Leroy', 'Prescot', '305-456-7890', 'leroyprescot14@gmail.com', 'good', '2025-02-24'),

('V05', 'Adam', 'Jake', '224-527-8765', 'adamjake@msn.com', 'initiated', NULL),

('V06', 'Logan', 'Peter', '220-657-7890', 'loganpeter@outlook.com', 'good', '2024-08-19'),

('V07', 'Toby', 'Tyler', '451-567-6578', 'tobytyler@gmail.com', 'initiated', NULL),

('V08', 'Tyson', 'Chandler', '312-876-4521', 'tysonchandler@msn.com', 'flag', '2024-09-14'),

('V09', 'Mary', 'Lovely', '512-489-0081', 'marylovely@gmail.com', 'good', '2025-04-16'),

('V10', 'James', 'Richard', '444-012-6890', 'jamesrichard12@outlook.com', 'good', '2024-07-15');

INSERT INTO event (eID, eLocation, eName, eDate, eAudience, eCategory)

VALUES

('093022ms\_wksp', 'Grove', 'Goal Setting', '2022-09-30', 'ms', 'workshop'),

('012023ms\_wksp', 'Grove', 'Study Techniques', '2023-01-20', 'ms', 'workshop'),

('121622ms\_service', 'St. Anne''s', 'Food Pantry', '2022-12-16', 'ms', 'service'),

('093022hs\_workshop', 'Sherlake', 'Optimal Work', '2022-09-30', 'hs', 'workshop'),

('100722hs\_workshop', 'Grove', 'Leadership', '2022-10-07', 'hs', 'workshop'),

('041322hs\_workshop', 'Lindell', 'Building Character', '2022-04-13', 'hs', 'workshop'),

('031622hs\_workshop', 'Sherlake', 'Leadership', '2022-03-16', 'hs', 'workshop'),

('100622hs\_workshop', 'Sherlake', 'Building Habits', '2022-10-06', 'hs', 'workshop'),

('111822hs\_service', 'St. Anne''s', 'Food Pantry', '2022-11-18', 'hs', 'service'),

('051823hs\_service', 'Arbor Care', 'Visit to Elderly', '2023-05-18', 'hs', 'service'),

('110422hs\_retreat', 'Shellbourne', 'HS Fall Retreat', '2022-11-04', 'hs', 'retreat'),

('032323hs\_retreat', 'Shellbourne', 'HS Spring Retreat', '2023-03-23', 'hs', 'retreat'),

('121722hs\_other', 'Sherlake', 'Christmas Party', '2022-12-17', 'hs', 'other'),

('102822co\_workshop', 'Elms', 'Optimal Work', '2022-10-28', 'co', 'workshop'),

('120922co\_service', 'Elms', 'Visit the Homeless', '2022-12-09', 'co', 'service'),

('021023co\_service', 'Southold', 'Visit the Homeless', '2023-02-10', 'co', 'service'),

('032323co\_service', 'St. Anne''s', 'Food Pantry', '2023-03-23', 'co', 'service'),

('011323co\_retreat', 'Shellbourne', 'College Winter Retreat', '2023-01-13', 'co', 'retreat'),

('042023co\_retreat', 'Shellbourne', 'College Spring Retreat', '2023-04-20', 'co', 'retreat'),

('110622co\_other', 'Southold', 'Dinner', '2022-11-06', 'co', 'other'),

('032623co\_other', 'Southold', 'Dinner', '2023-03-26', 'co', 'other'),

('042823co\_other', 'Elms', 'Movie Night', '2023-04-28', 'co', 'other');

INSERT INTO Takes (cID, sID)

VALUES

('ELco101122','LeKoch'),('ELco101122','RyWilliams'),('ELco101122','DaMartinez'),

('ELco110822','OlSmith'),('ELco110822','LeKoch'),('ELco110822','SaJohnson'),

('ELco110822','MaTaylor'),('ELco011023','AvBrown'),('ELco011023','OlSmith'),

('ELco021423','LeKoch'),('ELco021423','RyWilliams'),('ELco021423','EmDavis'),

('ELco021423','SaJohnson'),('ELco031423','DaMartinez'),('ELco031423','MaTaylor'),

('ELco031423','OlSmith'),('ELco041123','LeKoch'),('ELco041123','RyWilliams'),

('ELco050923','LeKoch'),('ELco050923','MaTaylor'),('ELco050923','SaJohnson'),

('GRhs091522','ChAnderson'),('GRhs091522','HeAdams'),('GRhs100622','ChHall'),

('GRhs100622','ChAnderson'),('GRhs100622','MiClark'),('GRhs100622','SoParker'),

('GRhs102022','LiGreen'),('GRhs102022','AmRodriguez'),('GRhs102022','NoMitchell'),

('GRhs110322','ChAnderson'),('GRhs110322','BeWilson'),('GRhs110322','SoParker'),

('GRhs111722','MiClark'),('GRhs111722','SaWhite'),('GRhs111722','BeWilson'),

('GRhs111722','NoMitchell'),('GRhs120122','BeWilson'),('GRhs120122','SoParker'),

('GRhs011923','SaWhite'),('GRhs011923','MiClark'),('GRhs011923','AmRodriguez'),

('GRhs020223','NoMitchell'),('GRhs020223','LiGreen'),('GRhs030223','BeWilson'),

('GRhs030223','MiClark'),('GRhs030223','SaWhite'),('GRhs031623','AmRodriguez'),

('GRhs031623','SoParker'),('GRhs031623','ChHall'),('GRhs040623','AmRodriguez'),

('GRhs040623','MiClark'),('GRhs040623','SaWhite'),('GRhs042023','NoMitchell'),

('GRhs042023','SoParker'),('GRhs042023','MiClark'),('GRhs050423','BeWilson'),

('GRhs050423','MiClark'),('GRhs050423','AmRodriguez'),('GRhs050423','SoParker'),

('GRms091922','IsRodriguez'),('GRms091922','JaThomas'),('GRms101722','AvAnderson'),

('GRms101722','IsRodriguez'),('GRms112122','IsRodriguez'),('GRms112122','JaThomas'),

('GRms112122','OlBrown'),('GRms121922','AvAnderson'),('GRms121922','IsRodriguez'),

('GRms011623','MiSmith'),('GRms011623','IsRodriguez'),('GRms011623','AvAnderson'),

('GRms022023','WiHarris'),('GRms022023','LiWilson'),('GRms041023','AvAnderson'),

('GRms041023','JaThomas'),('PWhs091522','ZoCooper'),('PWhs091522','LoWard'),

('PWhs102722','ZoCooper'),('PWhs102722','DaCarter'),('PWhs102722','CaMitchell'),

('PWhs102722','EmRivera'),('PWhs110322','EmRivera'),('PWhs110322','ZoCooper'),

('PWhs111722','ZoCooper'),('PWhs111722','EmRivera'),('PWhs120122','ZoCooper'),

('PWhs120122','DaCarter'),('PWhs020223','LiScott'),('PWhs020223','LoWard'),

('PWhs021623','ZoCooper'),('PWhs021623','DaCarter'),('PWhs030223','EmRivera'),

('PWhs030223','ZoCooper'),('PWhs031623','JaStewart'),('PWhs031623','DaCarter'),

('PWhs031623','AnRoss'),('PWhs042023','ZoCooper'),('PWhs042023','ElYoung'),

('PWhs050423','ZoCooper'),('PWhs050423','EmRivera'),('PWhs051823','CaMitchell'),

('PWhs051823','ZoCooper'),('SBhs093022','EmSullivan'),('SBhs093022','MaHernandez'),

('SBhs102822','LuWilson'),('SBhs111822','ZaParker'),('SBhs111822','MiStewart'),

('SBhs111822','EmSullivan'),('SBhs121622','AbMorris'),('SBhs012723','EmSullivan'),

('SBhs012723','MaHernandez'),('SBhs012723','OwCarter'),('SBhs022423','EmSullivan'),

('SBhs022423','ZaParker'),('SBhs033123','AbMorris'),('SBhs033123','JoPeterson'),

('SBhs042823','JoPeterson'),('SBhs042823','EmSullivan'),('SBhs042823','ZaParker'),

('SBhs052623','NiJohnson'),('SBhs052623','ZaParker'),('SLhs101222','EmSchneider'),

('SLhs101222','AnEllis'),('SLhs102622','ClHammond'),('SLhs102622','LeYates'),

('SLhs102622','CeCollier'),('SLhs110922','CaFlores'),('SLhs110922','TaOsborne'),

('SLhs110922','EmSchneider'),('SLhs112322','EmSchneider'),('SLhs112322','LeMorris'),

('SLhs121422','ClHammond'),('SLhs011123','LoCunningham'),('SLhs011123','CeCollier'),

('SLhs011123','EmSchneider'),('SLhs012523','JoJohnson'),('SLhs012523','LoCunningham'),

('SLhs020823','EmSchneider'),('SLhs020823','LeMorris'),('SLhs022223','LeYates'),

('SLhs022223','PaWhite'),('SLhs022223','CaFlores'),('SLhs030823','EmSchneider'),

('SLhs032223','EmSchneider'),('SLhs032223','LoCunningham'),('SLhs041223','AnEllis'),

('SLhs041223','LeMorris'),('SLhs041923','EmSchneider'),('SLhs041923','ClHammond'),

('SLms092622','NoClark'),('SLms092622','LiNyugen'),('SLms102422','StGuzman'),

('SLms102422','StHoffman'),('SLms102422','NoClark'),('SLms112822','DaMontgomery'),

('SLms112822','NoClark'),('SLms121922','BrMitchell'),('SLms121922','JaLee'),

('SLms121922','NoClark'),('SLms013023','StGuzman'),('SLms013023','SaGardner'),

('SLms013023','StHoffman'),('SLms013023','NoClark'),('SLms022723','NoClark'),

('SLms022723','JaLee'),('SLms032723','JaLee'),('SLms032723','StGuzman'),

('SLms042423','SaGardner'),('SLms042423','NoClark'),('SHhs090722','NoWarner'),

('SHhs090722','LeRuiz'),('SHhs090722','FaCasey'),('SHhs110222','NoWarner'),

('SHhs110222','LeRuiz'),('SHhs010423','NoWarner'),('SHhs010423','LeRuiz'),

('SHhs010423','FaCasey'),('SHhs030723','FaCasey'),('SHhs050223','NoWarner'),

('SHhs050223','FaCasey'),('SHco101322','HaMcguire'),('SHco101322','MaRoberts'),

('SHco111022','HaMcguire'),('SHco120822','KaFleming'),('SHco120822','HaMcguire'),

('SHco020223','LuGomez'),('SHco020223','HaMcguire'),('SHco030123','KaFleming'),

('SHco030123','LuGomez'),('SHco040523','HaMcguire'),('SHco040523','CaBlake'),

('SHco050323','HaMcguire'),('SHco050323','MaHudson');

INSERT INTO Teaches (cID, pID) VALUES

('ELco101122', 'LiBennett'),('ELco110822', 'EmEvans'),('ELco121322', 'MaSmith'),

('ELco011023', 'MiKeller'),('ELco021423', 'JaReynolds'),('ELco031423', 'WiWallace'),

('ELco041123', 'AlDixon'),('ELco050923', 'ChReed'),('GRhs091522', 'CaHughes'),

('GRhs100622', 'MaSmith'),('GRhs102022', 'NoDavis'),('GRhs110322', 'AvGreene'),

('GRhs111722', 'IsIngram'),('GRhs120122', 'LuJensen'),('GRhs011923', 'EtLawson'),

('GRhs020223', 'OlNelson'),('GRhs021623', 'AmOwens'),('GRhs030223', 'ElParker'),

('GRhs031623', 'AbStewart'),('GRhs040623', 'WiWallace'),('GRhs042023', 'HeZimmerman'),

('GRhs050423', 'GrHarrison'),('GRms091922', 'AlDixon'),('GRms091922', 'LiMason'),

('GRms101722', 'SaPorter'),('GRms101722', 'DaSanders'),('GRms112122', 'ZoWarren'),

('GRms112122', 'CaHughes'),('GRms112122', 'ElRoss'),('GRms121922', 'LiBennett'),

('GRms121922', 'AvGreene'),('GRms011623', 'AiHayes'),('GRms011623', 'IsIngram'),

('GRms022023', 'MaSmith1'),('GRms032023', 'LuJensen'),('GRms032023', 'MiKeller'),

('GRms032023', 'EtLawson'),('GRms041023', 'ChVaughn'),('GRms041023', 'HeZimmerman'),

('GRms041023', 'GrHarrison'),('PWhs091522', 'AlDixon'),('PWhs100622', 'LiMason'),

('PWhs102722', 'SaPorter'),('PWhs102722', 'ChReed'),('PWhs111722', 'CaHughes'),

('PWhs111722', 'ElRoss'),('PWhs120122', 'SoBennett'),('PWhs120122', 'OlNelson'),

('PWhs120122', 'AmOwens'),('PWhs020223', 'BeTurner'),('PWhs021623', 'ScYoung'),

('PWhs021623', 'HeZimmerman'),('PWhs021623', 'AlDixon'),('PWhs030223', 'LiMason'),

('PWhs030223', 'SaPorter'),('PWhs031623', 'ChReed'),('PWhs031623', 'ZoWarren'),

('PWhs040623', 'CaHughes'),('PWhs040623', 'ElRoss'),('PWhs042023', 'LiBennett'),

('PWhs042023', 'AvGreene'),('PWhs050423', 'AiHayes'),('PWhs050423', 'IsIngram'),

('SBhs093022', 'MaSmith1'),('SBhs093022', 'LuJensen'),('SBhs102822', 'MiKeller'),

('SBhs102822', 'EtLawson'),('SBhs111822', 'ChVaughn'),('SBhs111822', 'HeZimmerman'),

('SBhs121622', 'GrHarrison'),('SBhs121622', 'LiMason'),('SBhs012723', 'SaPorter'),

('SBhs022423', 'ChReed'),('SBhs022423', 'DaSanders'),('SBhs033123', 'OlNelson'),

('SBhs033123', 'BeTurner'),('SBhs042823', 'ChVaughn'),('SBhs042823', 'WiWallace'),

('SBhs052623', 'ScYoung'),('SBhs052623', 'HeZimmerman'),('SLhs101222', 'GrHarrison'),

('SLhs101222', 'LiMason'),('SLhs102622', 'SaPorter'),('SLhs102622', 'ChReed'),

('SLhs110922', 'DaSanders'),('SLhs110922', 'WiWallace'),('SLhs112322', 'HeZimmerman'),

('SLhs112322', 'GrHarrison'),('SLhs121422', 'AlDixon'),('SLhs121422', 'LiMason'),

('SLhs011123', 'SaPorter'),('SLhs011123', 'ChReed'),('SLhs012523', 'DaSanders'),

('SLhs020823', 'SaPorter'),('SLhs020823', 'ChReed'),('SLhs020823', 'DaSanders'),

('SLhs022223', 'GrHarrison'),('SLhs022223', 'AlDixon'),('SLhs030823', 'LiMason'),

('SLhs030823', 'WiWallace'),('SLms092622', 'ScYoung'),('SLms092622', 'HeZimmerman'),

('SLms092622', 'GrHarrison'),('SLms102422', 'AlDixon'),('SLms102422', 'LiMason'),

('SLms102422', 'SaPorter'),('SLms112822', 'ChReed'),('SLms112822', 'DaSanders'),

('SLms112822', 'ZoWarren'),('SLms121922', 'CaHughes'),('SLms121922', 'ElRoss'),

('SLms121922', 'MaPowell'),('SLms013023', 'SoBennett'),('SLms013023', 'AmOwens'),

('SLms013023', 'OlNelson'),('SLms022723', 'BeTurner'),('SLms022723', 'ChVaughn'),

('SLms022723', 'WiWallace'),('SHhs090722', 'HeZimmerman'),('SHhs090722', 'GrHarrison'),

('SHhs090722', 'LiMason'),('SHhs110222', 'SaPorter'),('SHhs110222', 'ChReed'),

('SHhs110222', 'DaSanders'),('SHhs010423', 'WiWallace'),('SHhs010423', 'ScYoung'),

('SHhs010423', 'HeZimmerman'),('SHhs030723', 'GrHarrison'),('SHhs030723', 'AlDixon'),

('SHhs030723', 'LiMason'),('SHhs050223', 'SaPorter'),('SHhs050223', 'ChReed'),

('SHhs050223', 'DaSanders'),('SHco101322', 'SaPorter'),('SHco101322', 'ChReed'),

('SHco101322', 'DaSanders'),('SHco111022', 'GrHarrison'),('SHco111022', 'AlDixon'),

('SHco111022', 'LiMason'),('SHco120822', 'WiWallace'),('SHco120822', 'ScYoung'),

('SHco120822', 'HeZimmerman'),('SHco020223', 'GrHarrison'),('SHco020223', 'AlDixon'),

('SHco020223', 'LiMason'),('SHco030123', 'SaPorter'),('SHco030123', 'ChReed'),

('SHco030123', 'DaSanders'),('SHco040523', 'ZoWarren'),('SHco040523', 'CaHughes'),

('SHco050323', 'ElRoss');

INSERT INTO Runs (pID, eID)

VALUES ('MaSmith', '093022ms\_wksp'), ('LiBennett', '093022ms\_wksp'), ('OlCarter', '012023ms\_wksp'),

('IsIngram', '012023ms\_wksp'), ('NoDavis', '121622ms\_service'), ('MaSmith', '121622ms\_service'),

('EmEvans', '093022hs\_workshop'), ('JaFisher', '093022hs\_workshop'), ('LiMason', '093022hs\_workshop'),

('AvGreene', '100722hs\_workshop'), ('AiHayes', '100722hs\_workshop'), ('IsIngram', '041322hs\_workshop'),

('MaSmith1', '041322hs\_workshop'), ('LuJensen', '031622hs\_workshop'), ('IsIngram', '031622hs\_workshop'),

('AbStewart', '100622hs\_workshop'), ('MiKeller', '100622hs\_workshop'), ('EtLawson', '111822hs\_service'),

('HaMitchell', '111822hs\_service'), ('OlNelson', '111822hs\_service'), ('IsIngram', '051823hs\_service'),

('AmOwens', '051823hs\_service'), ('ElParker', '051823hs\_service'), ('WiWallace', '110422hs\_retreat'),

('EvQuinn', '110422hs\_retreat'), ('ElParker', '110422hs\_retreat'), ('JaReynolds', '110422hs\_retreat'),

('AbStewart', '110422hs\_retreat'), ('ScYoung', '121722hs\_other'), ('BeTurner', '032323hs\_retreat'),

('ChVaughn', '032323hs\_retreat'), ('WiWallace', '032323hs\_retreat'), ('HeZimmerman', '121722hs\_other'),

('GrHarrison', '102822co\_workshop'), ('AlDixon', '102822co\_workshop'), ('ScYoung', '120922co\_service'),

('LiMason', '120922co\_service'), ('SaPorter', '021023co\_service'), ('ScYoung', '021023co\_service'),

('EtLawson', '032323co\_service'), ('LiMason', '032323co\_service'), ('DaSanders', '011323co\_retreat'),

('ZoWarren', '011323co\_retreat'), ('HaMitchell', '042023co\_retreat'), ('CaHughes', '042023co\_retreat'),

('ElRoss', '110622co\_other'), ('DaSanders', '110622co\_other'), ('IsIngram', '032623co\_other'),

('ChVaughn', '032623co\_other'), ('JaReynolds', '042823co\_other'), ('ElParker', '042823co\_other'),

('HeZimmerman', '042823co\_other');

INSERT INTO attends (sID, eID) VALUES

('ChAnderson', '093022ms\_wksp'), ('AvAnderson', '012023ms\_wksp'), ('AnEllis', '121622ms\_service'), ('JaMorris', '093022hs\_workshop'),

('LoShaw', '100722hs\_workshop'), ('HaMcguire', '041322hs\_workshop'), ('AvBrown', '031622hs\_workshop'), ('KaFleming', '100622hs\_workshop'),

('EmDavis', '111822hs\_service'), ('MaHudson', '051823hs\_service'), ('LeYates', '110422hs\_retreat'), ('BrMitchell', '032323hs\_retreat'), ('JaLee', '121722hs\_other'),

('FaCasey', '102822co\_workshop'), ('FaCasey', '120922co\_service'), ('MaBecker', '021023co\_service'), ('BrMitchell', '032323co\_service'), ('LiWood', '011323co\_retreat'),

('DaCarter', '042023co\_retreat'), ('LoMiller', '110622co\_other'), ('ZaParker', '032623co\_other'), ('MaRoberts', '042823co\_other'), ('JaMorris', '093022ms\_wksp'),

('ChAnderson', '012023ms\_wksp'), ('EmDavis', '121622ms\_service'), ('ChAnderson', '093022hs\_workshop'), ('JaMoore', '100722hs\_workshop'), ('ClHammond', '041322hs\_workshop'),

('OlBrown', '031622hs\_workshop'), ('AnEllis', '100622hs\_workshop'), ('CeCollier', '111822hs\_service'), ('KaFleming', '051823hs\_service'), ('SoParker', '110422hs\_retreat'),

('SaGardner', '032323hs\_retreat'), ('ElYoung', '121722hs\_other'), ('CaFlores', '102822co\_workshop'), ('NoClark', '120922co\_service'), ('ZoCooper', '021023co\_service'),

('WiHarris', '032323co\_service'), ('SoMorgan', '011323co\_retreat');

INSERT INTO HELPS(vID, eID) VALUES

('V01','093022hs\_workshop'), ('V05', '102822co\_workshop'), ('V10', '032323co\_service'), ('V08', '110622co\_other'), ('V03', '042823co\_other'),

('V01','102822co\_workshop'), ('V10', '042823co\_other'), ('V08', '032323co\_service'), ('V08', '110422hs\_retreat'), ('V03', '100722hs\_workshop');

Query Name: VolunteerHelp

Student Name: David Sheleru

Description: For each event, list the volunteer’s ID, volunteer’s name, training status, and phone number. Only print out the results if the volunteer assigned to the event has participated in less than 2 events. Order by Volunteer’s name in ascending order.

Query:

Select V.VID,V.vFirst,V.VtrainStatus,V.vPhoneNumber, Count(DISTINCT H.eid) AS Numberofeventsmanaged

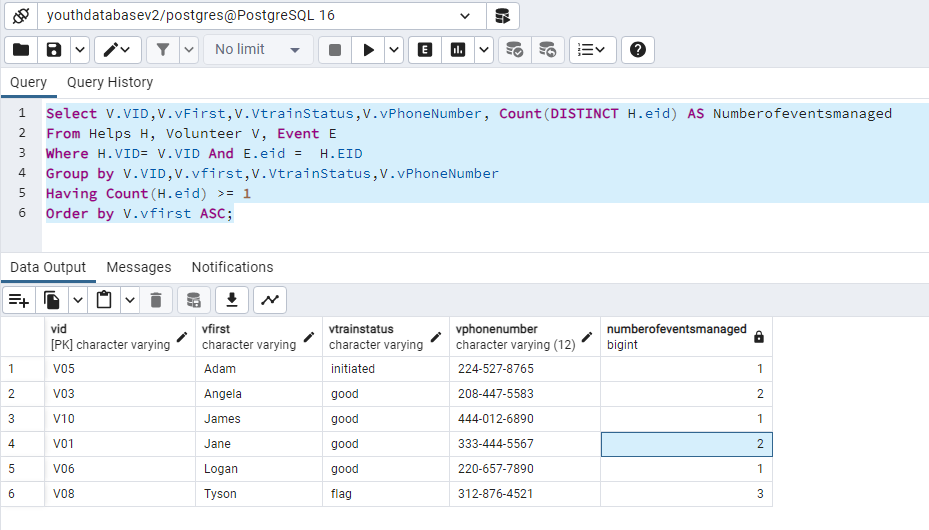
From Helps H, Volunteer V, Event E

Where H.VID= V.VID And E.eid = H.EID

Group by V.VID,V.vfirst,V.VtrainStatus,V.vPhoneNumber

Having Count(H.eid) >= 1

Order by V.vfirst ASC;



Query Name: AvgStudentsClassLevel

Student Name: Rachel Gordon

Description: Find the average number of students attending classes grouped by class level and center and ordered by center name. The inner query obtains the total number of distinct classes of a particular level offered at each center as well as the total number of students who attend classes of each level at each center. The outer query then finds the average number of students per class by dividing num\_students by num\_classes, grouping by class level and center, and ordering by center name in alphabetical order.

Query:

SELECT cenName, cLevel, num\_students/num\_classes as avg\_students\_per\_class

FROM (/\* find the total number of classes of a particular level offered at each center

and the total number of students who have attended those classes\*/

SELECT C.cenID, C.cenName, CL.cLevel, count(S.sID) as num\_students, count(distinct CL.cID) as num\_classes

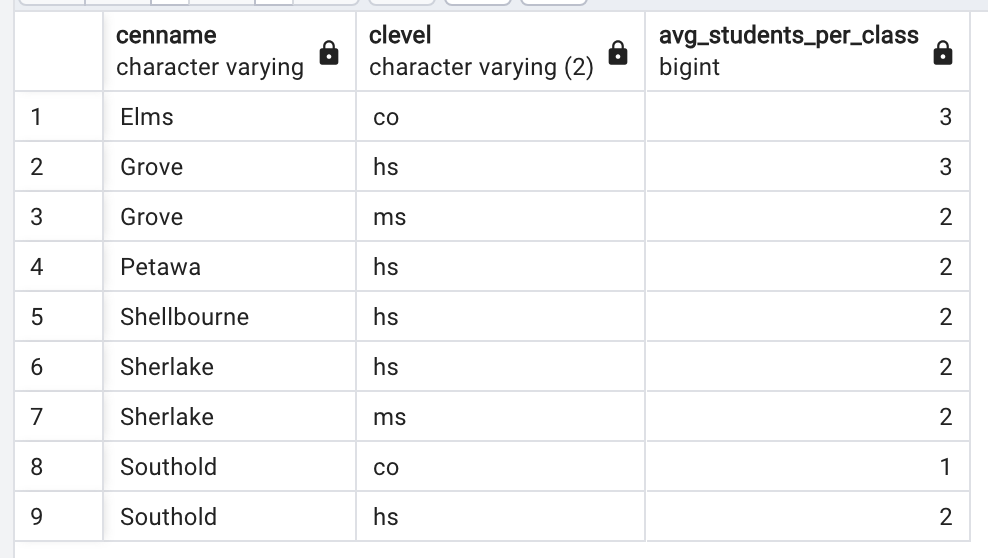
FROM student S, takes T, class CL, center C

WHERE S.sID = T.sID and T.cID = CL.cID and CL.cenID = C.cenID

GROUP BY CL.clevel, C.cenID, C.cenName)

GROUP BY cLevel, cenID, cenName, num\_students, num\_classes

ORDER BY cenName;



Query Name: LeadershipGroupClassAttendance

Student Name: Taylor Fischer

Description: For each center at each class level, print the number of distinct students attending classes and the number of those students who are part of the leadership group. Include the center name and class level. Order by center name and then class level.

Query:

SELECT Cen.cenName, Tot.cLevel, TotalStudentCount, LeadershipStudentCount

FROM Center Cen,

/\*Inner query for count of distinct leadership group students attending classes\*/

(SELECT C.cenID, C.cLevel, count(distinct S.sID) as LeadershipStudentCount

FROM Student S, Takes T, Class C

WHERE T.sID=S.sID and C.cID=T.cID and S.sLeadership='yes'

GROUP BY C.cenID, c.cLevel

) Lead,

/\*Inner query for count of total distinct students attending classes\*/

(SELECT C.cenID, C.cLevel, count(distinct S.sID) as TotalStudentCount

FROM Student S, Takes T, Class C

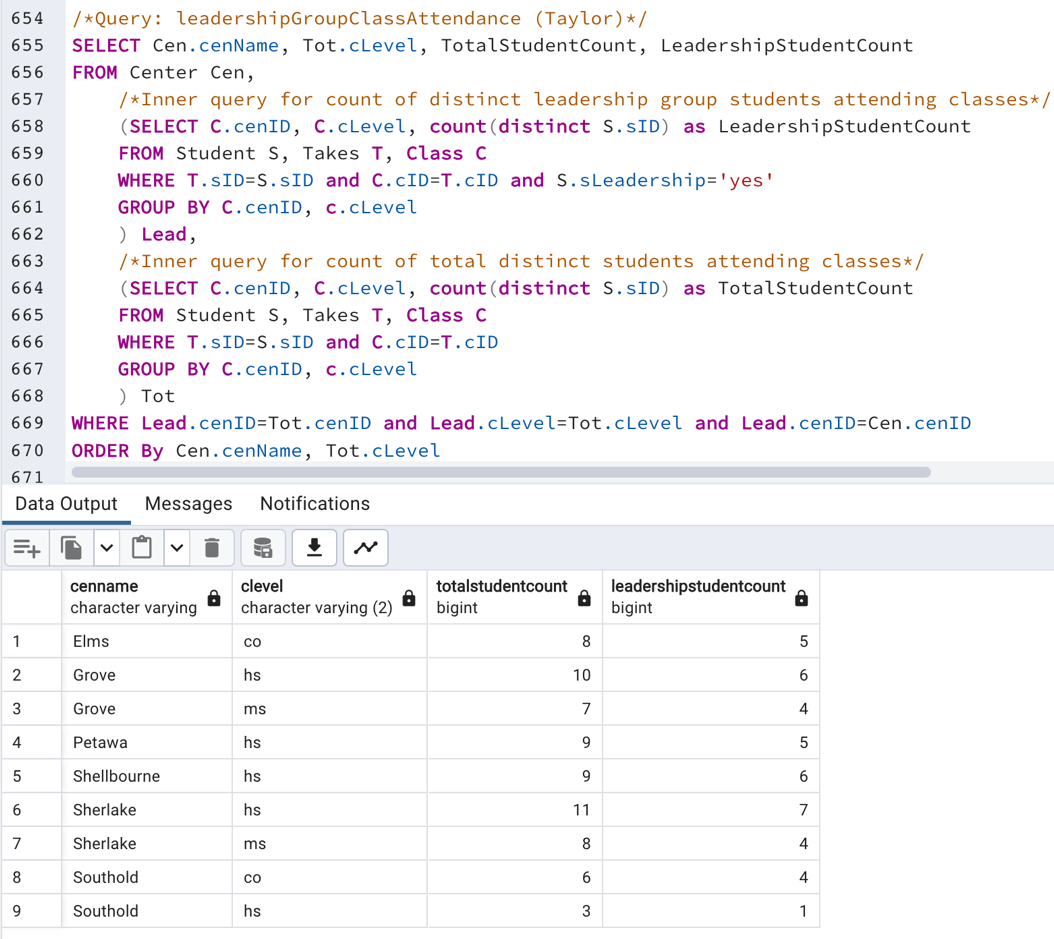
WHERE T.sID=S.sID and C.cID=T.cID

GROUP BY C.cenID, c.cLevel

) Tot

WHERE Lead.cenID=Tot.cenID and Lead.cLevel=Tot.cLevel and Lead.cenID=Cen.cenID

ORDER By Cen.cenName, Tot.cLevel;



Query Name: NumberOfEvents

Student Name: Max Simmer

Description: Find each student that attends more than one event and list the specific events they attend. Include student first and last name as well as their student ID. Order by numberOfEvents > 1 in descending order.

Query:

SELECT

s.sID AS StudentID,

s.sFirst AS FirstName,

s.sLast AS LastName,

COUNT(DISTINCT a.eID) AS numberOfEvents,

STRING\_AGG(e.eName, ', ' ORDER BY e.eName) AS AttendedEvents

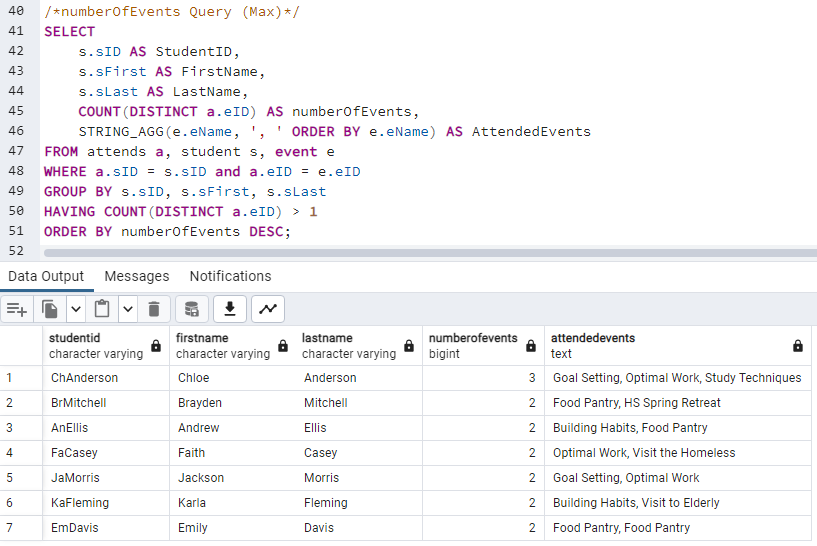
FROM attends a, student s, event e

WHERE a.sID = s.sID and a.eID = e.eID

GROUP BY s.sID, s.sFirst, s.sLast

HAVING COUNT(DISTINCT a.eID) > 1

ORDER BY numberOfEvents DESC;



Query Name: LowestStudentEventAttendance

Student Name: Kadin Suga

Description: Lowest student attendance where a student attends one class and attends an event. List the number of students attending one class and the number of events attended with first name in ascending order.

Query:

SELECT s.sFirst as FirstName,

s.sLast as LastName,

COUNT(DISTINCT C.cID) as classAttend,

COUNT(DISTINCT E.eID) as eventAttend

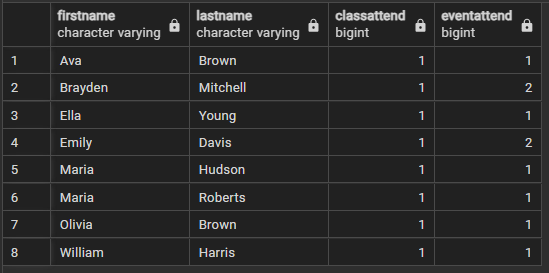
FROM Student S, Event E, Class C, Takes T, Attends A

WHERE A.eID = E.eID and A.sID = S.sID and S.sID = T.sID and T.cID = C.cID

GROUP BY s.sFirst, s.sLast

HAVING COUNT(DISTINCT C.cID) <= 1

ORDER BY classAttend ASC;



**8. ORM Implementation**

from sqlite3 import Date

from sqlalchemy import CHAR, Date

from typing import List

from typing import Optional

from sqlalchemy import ForeignKey

from sqlalchemy import String, Integer

from sqlalchemy.orm import DeclarativeBase

from sqlalchemy.orm import Mapped

from sqlalchemy.orm import mapped\_column

from sqlalchemy.orm import relationship

from sqlalchemy import create\_engine

from sqlalchemy.orm import Session

from sqlalchemy import select

#DB Connection: create\_engine(DBMS\_name+driver://<username>:<password>@<hostname>/<database\_name>)

engine = create\_engine("postgresql+psycopg2://postgres:Gordon0572!@localhost/")

#Define Classes/Tables

class Base(DeclarativeBase):

pass

#Taylor

class Center(Base):

\_\_tablename\_\_ = "center" #NOTE LOWERCASE

cenid: Mapped[str] = mapped\_column(String(2), primary\_key=True) #NOTE LOWERCASE cenid

cenname: Mapped[str] = mapped\_column(String)

#relation to Student: AffiliatedS (M Student : 1 Center)

students: Mapped[List["Student"]] = relationship(back\_populates="center", cascade="all, delete-orphan")

#relation to Class: Hosts (M Class : 1 Center)

classes: Mapped[List["Class"]] = relationship(back\_populates="center", cascade="all, delete-orphan")

#relation to PermStaff: AffiliatedP (M PermStaff : 1 Center)

permstaffs: Mapped[List["PermStaff"]] = relationship(back\_populates="center", cascade="all, delete-orphan")

def \_\_repr\_\_(self) -> str: #represents the object as a string

return f"Center(cenid={self.cenid!r}, cenname={self.cenname!r})"

#Taylor

class Student(Base):

\_\_tablename\_\_ = "student"

sid: Mapped[str] = mapped\_column(String, primary\_key=True)

sfirst: Mapped[str] = mapped\_column(String)

slast: Mapped[str] = mapped\_column(String)

sphonenumber: Mapped[str] = mapped\_column(String(12))

sbday: Mapped[Date] = mapped\_column(Date)

slevel: Mapped[str] = mapped\_column(String)

sleadership: Mapped[str] = mapped\_column(String, nullable=True)

sparentalum: Mapped[str] = mapped\_column(String, nullable=True)

semergencyphone: Mapped[str] = mapped\_column(String)

cenid: Mapped[str] = mapped\_column(String(2), ForeignKey("center.cenid"))

#relation to Center: AffiliatedS (M Student : 1 Center)

center: Mapped["Center"] = relationship(back\_populates="students")

#relation to Class: Takes (M : M)

classes: Mapped[List["Takes"]] = relationship(back\_populates="student") #<--Note the List!

#relation to Event: Attends (M : M)

events: Mapped[List["Attends"]] = relationship(back\_populates="student")

def \_\_repr\_\_(self):

return f"Student(sid={self.sid!r}, sfirst={self.sfirst!r}, slast={self.slast!r})"

#Rachel

class PermStaff(Base):

\_\_tablename\_\_ = "permstaff"

pid: Mapped[str] = mapped\_column(String, primary\_key=True, nullable=False)

pfirst: Mapped[str] = mapped\_column(String, nullable=False)

plast: Mapped[str] = mapped\_column(String, nullable=False)

pemail: Mapped[str] = mapped\_column(String, nullable=False)

pphonenumber: Mapped[str] = mapped\_column(String(12))

ptrainstatus: Mapped[str] = mapped\_column(String, nullable=False)

ptrainexp: Mapped[Date] = mapped\_column(Date, nullable=True)

cenid: Mapped[str] = mapped\_column(String(2), ForeignKey('center.cenid'), nullable=False)

#relation to Center: AffiliatedP (M PermStaff: 1 Center

center: Mapped["Center"] = relationship(back\_populates="permstaffs")

#relation to Class: Teaches (M : M)

classes: Mapped[List["Teaches"]] = relationship(back\_populates="permstaff")

#relation to Event: Runs (M : M)

events: Mapped[List["Runs"]] = relationship(back\_populates="permstaff")

def \_\_repr\_\_(self) -> str: #represents the object as a string

return f"PermStaff(pid={self.pid!r}, pfirst={self.pfirst!r}, plast={self.plast!r}, pemail={self.pemail!r}, pphonenumber={self.pphonenumber!r}, ptrainstatus={self.ptrainstatus!r}, ptrainexp={self.ptrainexp!r})"

#Kadin

class Class(Base):

\_\_tablename\_\_ = 'class'

cid: Mapped[str] = mapped\_column(String, primary\_key=True, nullable=False)

clevel: Mapped[str] = mapped\_column(String(2), nullable=False)

cdate: Mapped[Date] = mapped\_column(Date, nullable=False)

ctopic: Mapped[str] = mapped\_column(String, nullable=False)

cenid: Mapped[str] = mapped\_column(String(2), ForeignKey('center.cenid'), nullable=False)

#relation to Center: Hosts (m Class : 1 Center)

center: Mapped["Center"] = relationship(back\_populates='classes')

#relation to Student: Takes (M:M)

students: Mapped[List["Takes"]] = relationship(back\_populates="class\_")

#relation to PermStaff: Teaches (M:M)

permstaffs: Mapped[List["Teaches"]] = relationship(back\_populates="class\_")

def \_\_repr\_\_(self) -> str:

return f"class(cid={self.cid!r}, clevel={self.clevel!r}, cdate={self.cdate!r}, ctopic={self.ctopic!r}, cenid={self.cenid!r})"

#David

class Volunteer(Base):

\_\_tablename\_\_ = 'volunteer'

vid: Mapped[str] = mapped\_column(String, primary\_key=True)

vfirst: Mapped[str] = mapped\_column(String, nullable=False)

vlast: Mapped[str] = mapped\_column(String, nullable=False)

vphonenumber: Mapped[str] = mapped\_column(String(12), nullable=False)

vemail: Mapped[str] = mapped\_column(String, nullable=False)

vtrainstatus: Mapped[Optional[str]] = mapped\_column(String)

vtrainexp: Mapped[Optional[Date]] = mapped\_column(Date)

#relation to Events: Helps (M:M)

events: Mapped[List["Helps"]] = relationship(back\_populates="volunteer")

def \_\_repr\_\_(self) -> str:

return (

f"volunteer(vid={self.vid!r}, vfirst={self.vfirst!r}, vlast={self.vlast!r}, "

f"vphonenumber={self.vphonenumber!r}, vemail={self.vemail!r}, "

f"vtrainstatus={self.vtrainstatus!r}, vtrainexp={self.vtrainexp!r})"

)

#Max

class Event(Base):

\_\_tablename\_\_ = 'event'

eid: Mapped[str] = mapped\_column(String, primary\_key=True)

elocation: Mapped[str] = mapped\_column(String, nullable=False)

ename: Mapped[str] = mapped\_column(String, nullable=False)

edate: Mapped[Date] = mapped\_column(Date, nullable=False)

eaudience: Mapped[CHAR] = mapped\_column(CHAR(2), nullable=False)

ecategory: Mapped[str] = mapped\_column(String, nullable=False)

# relationship to volunteer, permstaff, and students

# attends, runs, helps?

students: Mapped[List["Attends"]] = relationship(back\_populates="event")

permstaffs: Mapped[List["Runs"]] = relationship(back\_populates="event")

volunteers: Mapped[List["Helps"]] = relationship(back\_populates="event")

def \_\_repr\_\_(self) -> str:

return (

f"event(eid={self.eid!r}, elocation={self.elocation!r}, ename={self.ename!r}, "

f"edate={self.edate!r}, eaudience={self.eaudience!r}, ecategory={self.ecategory!r})"

)

class Takes(Base):

\_\_tablename\_\_ = 'takes'

cid: Mapped[str] = mapped\_column(String, ForeignKey('class.cid'), primary\_key=True)

sid: Mapped[str] = mapped\_column(String, ForeignKey('student.sid'), primary\_key=True)

#many to many

#relation to Student: Takes (M:M)

student: Mapped["Student"] = relationship(back\_populates='classes')

##relation to Class: Takes (M : M)

class\_: Mapped["Class"] = relationship(back\_populates='students')

def \_\_repr\_\_(self) -> str:

return f"takes(cid={self.cid!r}, sid={self.sid!r})"

#Kadin

class Teaches(Base):

\_\_tablename\_\_ = 'teaches'

cid: Mapped[str] = mapped\_column(String, ForeignKey('class.cid'), primary\_key=True)

pid: Mapped[str] = mapped\_column(String, ForeignKey('permstaff.pid'), primary\_key=True)

# many to many

permstaff: Mapped["PermStaff"] = relationship(back\_populates='classes')

class\_: Mapped["Class"] = relationship(back\_populates='permstaffs')

def \_\_repr\_\_(self) -> str:

return f"teaches(cid={self.cid!r}, pid={self.pid!r})"

#Max

class Runs(Base):

\_\_tablename\_\_ = 'runs'

pid: Mapped[str] = mapped\_column(String, ForeignKey('permstaff.pid'), primary\_key=True)

eid: Mapped[str] = mapped\_column(String, ForeignKey('event.eid'), primary\_key=True)

#many to many

permstaff: Mapped["PermStaff"] = relationship(back\_populates='events')

event: Mapped["Event"] = relationship(back\_populates='permstaffs')

def \_\_repr\_\_(self) -> str:

return f"runs(pid={self.pid!r}, eid={self.eid!r})"

class Attends(Base):

\_\_tablename\_\_ = 'attends'

sid: Mapped[str] = mapped\_column(String, ForeignKey('student.sid'), primary\_key=True)

eid: Mapped[str] = mapped\_column(String, ForeignKey('event.eid'), primary\_key=True)

# many to many

student: Mapped["Student"] = relationship(back\_populates='events')

event: Mapped["Event"] = relationship(back\_populates='students')

def \_\_repr\_\_(self) -> str:

return f"attends(sid={self.sid!r}, eid={self.eid!r})"

#David

class Helps(Base):

\_\_tablename\_\_ = 'helps'

vid: Mapped[str] = mapped\_column(String, ForeignKey('volunteer.vid'), primary\_key=True)

eid: Mapped[str] = mapped\_column(String, ForeignKey('event.eid'), primary\_key=True)

# many to many

volunteer: Mapped["Volunteer"] = relationship(back\_populates='events')

event: Mapped["Event"] = relationship(back\_populates='volunteers')

def \_\_repr\_\_(self) -> str:

return f"helps(vid={self.vid!r}, eid={self.eid!r})"

Base.metadata.create\_all(engine)

# Create a session

session = Session(engine)

#Insert Data

with Session(engine) as session:

sample\_centers = [

Center(cenid='EL', cenname='Elms'),

Center(cenid='FG', cenname='Forest Glen'),

Center(cenid='GR', cenname='Grove'),

Center(cenid='LD', cenname='Lindell'),

Center(cenid='PW', cenname='Petawa'),

Center(cenid='SB', cenname='Shellbourne'),

Center(cenid='SL', cenname='Sherlake'),

Center(cenid='SH', cenname='Southold'),

]

# Student

sample\_students = [

Student(sid='LeKoch', sfirst='Leah', slast='Koch', sphonenumber='331-336-8319', sbday='01/05/05', slevel='co', sleadership='yes', sparentalum=None, semergencyphone='738-495-2160', cenid='EL'),

Student(sid='SaJohnson', sfirst='Sarah', slast='Johnson', sphonenumber='847-874-3158', sbday='03/26/01', slevel='co', sleadership=None, sparentalum=None, semergencyphone='624-809-7315', cenid='EL'),

Student(sid='RyWilliams', sfirst='Ryan', slast='Williams', sphonenumber='331-924-7168', sbday='09/15/01', slevel='co', sleadership=None, sparentalum='yes', semergencyphone='583-672-9140', cenid='EL'),

Student(sid='EmDavis', sfirst='Emily', slast='Davis', sphonenumber='312-565-0293', sbday='06/02/02', slevel='co', sleadership='yes', sparentalum='yes', semergencyphone='459-218-6370', cenid='EL'),

Student(sid='DaMartinez', sfirst='Daniel', slast='Martinez', sphonenumber='618-947-1162', sbday='07/14/03', slevel='co', sleadership=None, sparentalum=None, semergencyphone='726-493-5180', cenid='EL'),

Student(sid='OlSmith', sfirst='Olivia', slast='Smith', sphonenumber='708-479-2157', sbday='12/13/01', slevel='co', sleadership='yes', sparentalum=None, semergencyphone='362-840-1597', cenid='EL'),

Student(sid='MaTaylor', sfirst='Matthew', slast='Taylor', sphonenumber='773-466-6844', sbday='08/24/01', slevel='co', sleadership='yes', sparentalum='yes', semergencyphone='950-286-4713', cenid='EL'),

Student(sid='AvBrown', sfirst='Ava', slast='Brown', sphonenumber='847-647-9798', sbday='10/05/04', slevel='co', sleadership='yes', sparentalum=None, semergencyphone='817-496-3250', cenid='EL'),

Student(sid='AlLee', sfirst='Alexander', slast='Lee', sphonenumber='331-805-3950', sbday='11/29/02', slevel='co', sleadership=None, sparentalum=None, semergencyphone='204-736-9815', cenid='EL'),

Student(sid='ChAnderson', sfirst='Chloe', slast='Anderson', sphonenumber='312-384-6007', sbday='01/12/05', slevel='hs', sleadership=None, sparentalum=None, semergencyphone='675-892-3140', cenid='GR'),

Student(sid='BeWilson', sfirst='Ben', slast='Wilson', sphonenumber='847-256-5819', sbday='02/05/06', slevel='hs', sleadership='yes', sparentalum=None, semergencyphone='384-762-9150', cenid='GR'),

Student(sid='MiClark', sfirst='Mia', slast='Clark', sphonenumber='618-184-7893', sbday='03/20/05', slevel='hs', sleadership='yes', sparentalum='yes', semergencyphone='123-450-8976', cenid='GR'),

Student(sid='SaWhite', sfirst='Samuel', slast='White', sphonenumber='331-798-6579', sbday='04/08/05', slevel='hs', sleadership='yes', sparentalum='yes', semergencyphone='892-731-6450', cenid='GR'),

Student(sid='ChHall', sfirst='Charlotte', slast='Hall', sphonenumber='618-670-1205', sbday='05/16/06', slevel='hs', sleadership=None, sparentalum='yes', semergencyphone='657-234-1890', cenid='GR'),

Student(sid='AmRodriguez', sfirst='Amelia', slast='Rodriguez', sphonenumber='224-889-4751', sbday='06/25/06', slevel='hs', sleadership='yes', sparentalum=None, semergencyphone='941-582-7306', cenid='GR'),

Student(sid='HeAdams', sfirst='Henry', slast='Adams', sphonenumber='312-431-9277', sbday='07/09/05', slevel='hs', sleadership='yes', sparentalum=None, semergencyphone='392-687-4150', cenid='GR'),

Student(sid='NoMitchell', sfirst='Noah', slast='Mitchell', sphonenumber='630-675-4148', sbday='08/14/08', slevel='hs', sleadership=None, sparentalum=None, semergencyphone='546-123-7890', cenid='GR'),

Student(sid='LiGreen', sfirst='Lily', slast='Green', sphonenumber='630-675-4148', sbday='09/02/06', slevel='hs', sleadership=None, sparentalum=None, semergencyphone='289-456-1073', cenid='GR'),

Student(sid='JaMoore', sfirst='James', slast='Moore', sphonenumber='773-971-9466', sbday='10/18/06', slevel='hs', sleadership=None, sparentalum=None, semergencyphone='810-946-2357', cenid='GR'),

Student(sid='SoParker', sfirst='Sofia', slast='Parker', sphonenumber='224-416-4139', sbday='11/27/05', slevel='hs', sleadership='yes', sparentalum='yes', semergencyphone='631-487-5290', cenid='GR'),

Student(sid='LiWilson', sfirst='Liam', slast='Wilson', sphonenumber='847-795-4328', sbday='08/15/09', slevel='ms', sleadership=None, sparentalum=None, semergencyphone='407-895-3621', cenid='GR'),

Student(sid='WiHarris', sfirst='William', slast='Harris', sphonenumber='708-778-0943', sbday='11/03/09', slevel='ms', sleadership='yes', sparentalum=None, semergencyphone='263-750-8149', cenid='GR'),

Student(sid='IsRodriguez', sfirst='Isabella', slast='Rodriguez', sphonenumber='618-215-1850', sbday='03/22/10', slevel='ms', sleadership='yes', sparentalum='yes', semergencyphone='859-234-1760', cenid='GR'),

Student(sid='JaThomas', sfirst='Jackson', slast='Thomas', sphonenumber='630-914-3122', sbday='07/09/11', slevel='ms', sleadership='yes', sparentalum=None, semergencyphone='512-406-7983', cenid='GR'),

Student(sid='MiSmith', sfirst='Michael', slast='Smith', sphonenumber='312-385-6097', sbday='10/12/10', slevel='ms', sleadership=None, sparentalum=None, semergencyphone='746-931-5208', cenid='GR'),

Student(sid='OlBrown', sfirst='Olivia', slast='Brown', sphonenumber='485-726-9134', sbday='01/05/09', slevel='ms', sleadership=None, sparentalum='yes', semergencyphone='372-916-5840', cenid='GR'),

Student(sid='WiTaylor', sfirst='William', slast='Taylor', sphonenumber='629-384-7150', sbday='05/18/10', slevel='ms', sleadership=None, sparentalum=None, semergencyphone='981-734-6025', cenid='GR'),

Student(sid='AvAnderson', sfirst='Ava', slast='Anderson', sphonenumber='314-902-8765', sbday='09/30/10', slevel='ms', sleadership='yes', sparentalum=None, semergencyphone='451-820-3796', cenid='GR'),

Student(sid='BeLee', sfirst='Benjamin', slast='Lee', sphonenumber='572-913-4680', sbday='12/08/09', slevel='ms', sleadership=None, sparentalum=None, semergencyphone='617-482-3905', cenid='GR'),

Student(sid='ZoCooper', sfirst='Zoe', slast='Cooper', sphonenumber='718-649-2053', sbday='06/08/08', slevel='hs', sleadership='yes', sparentalum='yes', semergencyphone='586-042-3179', cenid='PW'),

Student(sid='LoWard', sfirst='Logan', slast='Ward', sphonenumber='109-785-4326', sbday='08/24/05', slevel='hs', sleadership='yes', sparentalum=None, semergencyphone='295-607-3841', cenid='PW'),

Student(sid='ViMurphy', sfirst='Victoria', slast='Murphy', sphonenumber='587-364-1029', sbday='03/17/08', slevel='hs', sleadership='yes', sparentalum=None, semergencyphone='730-149-8256', cenid='PW'),

Student(sid='JaStewart', sfirst='Jackson', slast='Stewart', sphonenumber='236-458-9701', sbday='10/29/06', slevel='hs', sleadership=None, sparentalum=None, semergencyphone='165-429-7380', cenid='PW'),

Student(sid='EmRivera', sfirst='Emma', slast='Rivera', sphonenumber='890-612-3475', sbday='01/14/08', slevel='hs', sleadership=None, sparentalum=None, semergencyphone='493-176-2805', cenid='PW'),

Student(sid='DaCarter', sfirst='Daniel', slast='Carter', sphonenumber='456-210-7893', sbday='07/05/07', slevel='hs', sleadership='yes', sparentalum='yes', semergencyphone='714-085-2369', cenid='PW'),

Student(sid='AdReed', sfirst='Addison', slast='Reed', sphonenumber='721-098-4365', sbday='12/25/05', slevel='hs', sleadership=None, sparentalum=None, semergencyphone='369-814-5270', cenid='PW'),

Student(sid='AnRoss', sfirst='Andrew', slast='Ross', sphonenumber='304-917-5826', sbday='05/02/06', slevel='hs', sleadership='yes', sparentalum=None, semergencyphone='872-301-4965', cenid='PW'),

Student(sid='ElYoung', sfirst='Ella', slast='Young', sphonenumber='638-574-2901', sbday='02/09/06', slevel='hs', sleadership=None, sparentalum=None, semergencyphone='504-972-6183', cenid='PW'),

Student(sid='CaMitchell', sfirst='Caleb', slast='Mitchell', sphonenumber='549-302-6871', sbday='11/12/07', slevel='hs', sleadership=None, sparentalum='yes', semergencyphone='610-934-8275', cenid='PW'),

Student(sid='LiWood', sfirst='Lily', slast='Wood', sphonenumber='765-219-8340', sbday='09/22/08', slevel='hs', sleadership=None, sparentalum=None, semergencyphone='283-650-1794', cenid='PW'),

Student(sid='LiScott', sfirst='Liam', slast='Scott', sphonenumber='823-705-6491', sbday='04/18/05', slevel='hs', sleadership=None, sparentalum=None, semergencyphone='971-245-8063', cenid='PW'),

Student(sid='AbMorris', sfirst='Abigail', slast='Morris', sphonenumber='193-284-5067', sbday='06/29/06', slevel='hs', sleadership='yes', sparentalum=None, semergencyphone='376-289-5401', cenid='SB'),

Student(sid='ZaParker', sfirst='Zach', slast='Parker', sphonenumber='408-576-3921', sbday='08/14/07', slevel='hs', sleadership='yes', sparentalum='yes', semergencyphone='219-674-8503', cenid='SB'),

Student(sid='ErHayes', sfirst='Ericka', slast='Hayes', sphonenumber='796-125-8340', sbday='03/07/07', slevel='hs', sleadership=None, sparentalum=None, semergencyphone='465-312-7890', cenid='SB'),

Student(sid='EmSullivan', sfirst='Emily', slast='Sullivan', sphonenumber='678-149-2053', sbday='10/08/05', slevel='hs', sleadership='yes', sparentalum='yes', semergencyphone='850-129-3467', cenid='SB'),

Student(sid='SoThompson', sfirst='Sophia', slast='Thompson', sphonenumber='295-748-6130', sbday='01/21/09', slevel='hs', sleadership=None, sparentalum=None, semergencyphone='792-438-5016', cenid='SB'),

Student(sid='JaMorris', sfirst='Jackson', slast='Morris', sphonenumber='437-289-6105', sbday='07/19/06', slevel='hs', sleadership='yes', sparentalum=None, semergencyphone='136-724-5980', cenid='SB'),

Student(sid='MiStewart', sfirst='Mia', slast='Stewart', sphonenumber='180-725-9364', sbday='12/11/05', slevel='hs', sleadership=None, sparentalum=None, semergencyphone='408-917-6253', cenid='SB'),

Student(sid='MaHernandez', sfirst='Marie', slast='Hernandez', sphonenumber='610-245-7983', sbday='05/26/07', slevel='hs', sleadership=None, sparentalum=None, semergencyphone='567-812-4309', cenid='SB'),

Student(sid='OwCarter', sfirst='Owen', slast='Carter', sphonenumber='954-682-3017', sbday='02/15/06', slevel='hs', sleadership=None, sparentalum=None, semergencyphone='305-246-9817', cenid='SB'),

Student(sid='JoPeterson', sfirst='Joseph', slast='Peterson', sphonenumber='276-804-9135', sbday='11/10/09', slevel='hs', sleadership=None, sparentalum=None, semergencyphone='924-871-3560', cenid='SB'),

Student(sid='NiJohnson', sfirst='Nicholas', slast='Johnson', sphonenumber='542-970-1863', sbday='09/05/08', slevel='hs', sleadership=None, sparentalum=None, semergencyphone='738-019-4652', cenid='SB'),

Student(sid='LuWilson', sfirst='Luke', slast='Wilson', sphonenumber='832-167-4950', sbday='04/10/07', slevel='hs', sleadership=None, sparentalum=None, semergencyphone='632-184-7905', cenid='SB'),

Student(sid='BrMitchell', sfirst='Brayden', slast='Mitchell', sphonenumber='691-237-8045', sbday='02/14/09', slevel='ms', sleadership=None, sparentalum=None, semergencyphone='514-627-8309', cenid='SL'),

Student(sid='NoClark', sfirst='Nora', slast='Clark', sphonenumber='174-609-2385', sbday='06/04/10', slevel='ms', sleadership='yes', sparentalum=None, semergencyphone='697-230-4851', cenid='SL'),

Student(sid='JaLee', sfirst='Jaxon', slast='Lee', sphonenumber='365-892-4701', sbday='08/27/09', slevel='ms', sleadership='yes', sparentalum=None, semergencyphone='890-245-3617', cenid='SL'),

Student(sid='LiNyugen', sfirst='Lisa', slast='Nyugen', sphonenumber='972-830-1456', sbday='11/11/10', slevel='ms', sleadership=None, sparentalum=None, semergencyphone='345-769-1820', cenid='SL'),

Student(sid='DaMontgomery', sfirst='David', slast='Montgomery', sphonenumber='536-478-9120', sbday='04/07/09', slevel='ms', sleadership='yes', sparentalum=None, semergencyphone='271-398-4605', cenid='SL'),

Student(sid='StHoffman', sfirst='Stephanie', slast='Hoffman', sphonenumber='890-372-5146', sbday='07/20/11', slevel='ms', sleadership=None, sparentalum='yes', semergencyphone='506-738-1294', cenid='SL'),

Student(sid='StGuzman', sfirst='Steven', slast='Guzman', sphonenumber='140-785-9623', sbday='09/02/09', slevel='ms', sleadership=None, sparentalum=None, semergencyphone='928-453-1760', cenid='SL'),

Student(sid='BrRodgers', sfirst='Brett', slast='Rodgers', sphonenumber='862-195-3470', sbday='12/31/10', slevel='ms', sleadership='yes', sparentalum=None, semergencyphone='763-491-8250', cenid='SL'),

Student(sid='LoShaw', sfirst='Lorena', slast='Shaw', sphonenumber='429-083-6715', sbday='03/01/09', slevel='ms', sleadership=None, sparentalum=None, semergencyphone='123-590-4678', cenid='SL'),

Student(sid='SaGardner', sfirst='Sara', slast='Gardner', sphonenumber='713-690-2458', sbday='05/12/10', slevel='ms', sleadership=None, sparentalum=None, semergencyphone='419-587-2360', cenid='SL'),

Student(sid='LoCunningham', sfirst='Louise', slast='Cunningham', sphonenumber='807-912-3465', sbday='06/27/06', slevel='hs', sleadership=None, sparentalum='yes', semergencyphone='874-062-9315', cenid='SL'),

Student(sid='AnEllis', sfirst='Andrew', slast='Ellis', sphonenumber='346-981-2570', sbday='08/09/08', slevel='hs', sleadership=None, sparentalum=None, semergencyphone='965-432-0781', cenid='SL'),

Student(sid='ClHammond', sfirst='Clara', slast='Hammond', sphonenumber='215-876-3490', sbday='03/27/08', slevel='hs', sleadership=None, sparentalum=None, semergencyphone='346-281-0975', cenid='SL'),

Student(sid='TaOsborne', sfirst='Tanya', slast='Osborne', sphonenumber='574-308-1629', sbday='10/04/07', slevel='hs', sleadership=None, sparentalum=None, semergencyphone='283-961-4750', cenid='SL'),

Student(sid='JoHopkins', sfirst='Joey', slast='Hopkins', sphonenumber='968-145-7320', sbday='01/30/06', slevel='hs', sleadership=None, sparentalum=None, semergencyphone='507-984-2163', cenid='SL'),

Student(sid='EmSchneider', sfirst='Emma', slast='Schneider', sphonenumber='341-205-7968', sbday='07/08/08', slevel='hs', sleadership='yes', sparentalum=None, semergencyphone='619-827-5304', cenid='SL'),

Student(sid='ArJensen', sfirst='Arthur', slast='Jensen', sphonenumber='528-467-9130', sbday='12/15/05', slevel='hs', sleadership=None, sparentalum=None, semergencyphone='652-140-3897', cenid='SL'),

Student(sid='CaFlores', sfirst='Carroll', slast='Flores', sphonenumber='629-781-0435', sbday='05/15/06', slevel='hs', sleadership=None, sparentalum=None, semergencyphone='430-789-2165', cenid='SL'),

Student(sid='CeCollier', sfirst='Cecil', slast='Collier', sphonenumber='195-634-8720', sbday='02/23/07', slevel='hs', sleadership=None, sparentalum=None, semergencyphone='728-391-0546', cenid='SL'),

Student(sid='LeMorris', sfirst='Leslie', slast='Morris', sphonenumber='478-921-6053', sbday='11/21/06', slevel='hs', sleadership='yes', sparentalum=None, semergencyphone='296-741-5380', cenid='SL'),

Student(sid='JoJohnson', sfirst='John', slast='Johnson', sphonenumber='573-629-1480', sbday='09/13/08', slevel='hs', sleadership=None, sparentalum=None, semergencyphone='981-276-4305', cenid='SL'),

Student(sid='PaWhite', sfirst='Paul', slast='White', sphonenumber='204-789-6135', sbday='04/25/05', slevel='hs', sleadership=None, sparentalum=None, semergencyphone='543-219-6780', cenid='SL'),

Student(sid='LeYates', sfirst='Lela', slast='Yates', sphonenumber='436-982-5701', sbday='06/02/07', slevel='hs', sleadership=None, sparentalum=None, semergencyphone='205-964-1873', cenid='SL'),

Student(sid='NoWarner', sfirst='Noah', slast='Warner', sphonenumber='681-093-2475', sbday='08/31/06', slevel='hs', sleadership=None, sparentalum=None, semergencyphone='486-321-0957', cenid='SH'),

Student(sid='LeRuiz', sfirst='Lena', slast='Ruiz', sphonenumber='903-817-6425', sbday='03/04/08', slevel='hs', sleadership=None, sparentalum=None, semergencyphone='317-289-5046', cenid='SH'),

Student(sid='FaCasey', sfirst='Faith', slast='Casey', sphonenumber='372-546-8019', sbday='10/01/07', slevel='hs', sleadership=None, sparentalum=None, semergencyphone='950-428-3671', cenid='SH'),

Student(sid='MaHudson', sfirst='Maria', slast='Hudson', sphonenumber='615-794-0283', sbday='11/10/03', slevel='co', sleadership='yes', sparentalum='yes', semergencyphone='760-348-1295', cenid='SH'),

Student(sid='JaStevenson', sfirst='Jake', slast='Stevenson', sphonenumber='287-415-0936', sbday='04/02/05', slevel='co', sleadership='yes', sparentalum=None, semergencyphone='621-594-0837', cenid='SH'),

Student(sid='CaBlake', sfirst='Carter', slast='Blake', sphonenumber='549-861-2730', sbday='09/14/01', slevel='co', sleadership=None, sparentalum=None, semergencyphone='397-186-2450', cenid='SH'),

Student(sid='MaRoberts', sfirst='Maria', slast='Roberts', sphonenumber='209-781-3645', sbday='02/08/04', slevel='co', sleadership=None, sparentalum=None, semergencyphone='185-762-3490', cenid='SH'),

Student(sid='MaMartinez', sfirst='Marta', slast='Martinez', sphonenumber='836-241-9705', sbday='05/20/03', slevel='co', sleadership=None, sparentalum=None, semergencyphone='349-275-6108', cenid='SH'),

Student(sid='OlBurns', sfirst='Olivia', slast='Burns', sphonenumber='794-083-1625', sbday='08/17/05', slevel='co', sleadership='yes', sparentalum=None, semergencyphone='702-918-4356', cenid='SH'),

Student(sid='CaGrant', sfirst='Casey', slast='Grant', sphonenumber='523-746-8901', sbday='12/31/02', slevel='co', sleadership='yes', sparentalum='yes', semergencyphone='518-439-2760', cenid='SH'),

Student(sid='KaFleming', sfirst='Karla', slast='Fleming', sphonenumber='461-208-9375', sbday='07/05/04', slevel='co', sleadership='yes', sparentalum='yes', semergencyphone='864-207-9315', cenid='SH'),

Student(sid='RoHunter', sfirst='Rosalie', slast='Hunter', sphonenumber='972-431-6085', sbday='03/25/01', slevel='co', sleadership=None, sparentalum=None, semergencyphone='294-158-7603', cenid='SH'),

Student(sid='HaMcguire', sfirst='Harper', slast='Mcguire', sphonenumber='134-205-7896', sbday='10/13/03', slevel='co', sleadership='yes', sparentalum=None, semergencyphone='637-819-2045', cenid='SH'),

Student(sid='LoMiller', sfirst='Logan', slast='Miller', sphonenumber='795-843-2610', sbday='01/03/05', slevel='co', sleadership='yes', sparentalum=None, semergencyphone='970-582-3164', cenid='SH'),

Student(sid='LuGomez', sfirst='Lucia', slast='Gomez', sphonenumber='680-324-7915', sbday='06/09/02', slevel='co', sleadership='yes', sparentalum=None, semergencyphone='541-283-0976', cenid='SH'),

Student(sid='KeFisher', sfirst='Kendra', slast='Fisher', sphonenumber='215-479-3608', sbday='11/18/04', slevel='co', sleadership=None, sparentalum=None, semergencyphone='326-845-7190', cenid='SH'),

Student(sid='MaBecker', sfirst='Mary', slast='Becker', sphonenumber='769-804-2315', sbday='04/14/01', slevel='co', sleadership=None, sparentalum=None, semergencyphone='187-420-9365', cenid='SH'),

Student(sid='SoMorgan', sfirst='Sofia', slast='Morgan', sphonenumber='301-874-5926', sbday='09/28/03', slevel='co', sleadership=None, sparentalum=None, semergencyphone='913-684-2705', cenid='SH'),

]

# Class

sample\_classes = [

Class(cid='ELco101122', clevel='co', cdate=('10/11/22'), ctopic='Cheerfulness', cenid='EL'),

Class(cid='ELco110822', clevel='co', cdate=('11/08/22'), ctopic='Working Well', cenid='EL'),

Class(cid='ELco121322', clevel='co', cdate=('12/13/22'), ctopic='Friendship', cenid='EL'),

Class(cid='ELco011023', clevel='co', cdate=('01/10/23'), ctopic='Fortitude', cenid='EL'),

Class(cid='ELco021423', clevel='co', cdate=('02/14/23'), ctopic='Justice', cenid='EL'),

Class(cid='ELco031423', clevel='co', cdate=('03/14/23'), ctopic='Temperance', cenid='EL'),

Class(cid='ELco041123', clevel='co', cdate=('04/11/23'), ctopic='Prudence', cenid='EL'),

Class(cid='ELco050923', clevel='co', cdate=('05/09/23'), ctopic='Leadership', cenid='EL'),

Class(cid='GRhs091522', clevel='hs', cdate=('09/15/22'), ctopic='Friendship', cenid='GR'),

Class(cid='GRhs100622', clevel='hs', cdate=('10/06/22'), ctopic='Courage', cenid='GR'),

Class(cid='GRhs102022', clevel='hs', cdate=('10/20/22'), ctopic='Study', cenid='GR'),

Class(cid='GRhs110322', clevel='hs', cdate=('11/03/22'), ctopic='Cheerfulness', cenid='GR'),

Class(cid='GRhs111722', clevel='hs', cdate=('11/17/22'), ctopic='Honesty', cenid='GR'),

Class(cid='GRhs120122', clevel='hs', cdate=('12/01/22'), ctopic='Leadership', cenid='GR'),

Class(cid='GRhs011923', clevel='hs', cdate=('01/19/23'), ctopic='Service', cenid='GR'),

Class(cid='GRhs020223', clevel='hs', cdate=('02/02/23'), ctopic='Prudence', cenid='GR'),

Class(cid='GRhs021623', clevel='hs', cdate=('02/16/23'), ctopic='Justice', cenid='GR'),

Class(cid='GRhs030223', clevel='hs', cdate=('03/02/23'), ctopic='Temperance', cenid='GR'),

Class(cid='GRhs031623', clevel='hs', cdate=('03/16/23'), ctopic='Fortitude', cenid='GR'),

Class(cid='GRhs040623', clevel='hs', cdate=('04/06/23'), ctopic='Respect', cenid='GR'),

Class(cid='GRhs042023', clevel='hs', cdate=('04/20/23'), ctopic='Motivation', cenid='GR'),

Class(cid='GRhs050423', clevel='hs', cdate=('05/04/23'), ctopic='Prayer', cenid='GR'),

Class(cid='GRms091922', clevel='ms', cdate=('09/19/22'), ctopic='Goal Setting', cenid='GR'),

Class(cid='GRms101722', clevel='ms', cdate=('10/17/22'), ctopic='Friendship', cenid='GR'),

Class(cid='GRms112122', clevel='ms', cdate=('11/21/22'), ctopic='Courage', cenid='GR'),

Class(cid='GRms121922', clevel='ms', cdate=('12/19/22'), ctopic='Study', cenid='GR'),

Class(cid='GRms011623', clevel='ms', cdate=('01/16/23'), ctopic='Cheerfulness', cenid='GR'),

Class(cid='GRms022023', clevel='ms', cdate=('02/20/23'), ctopic='Honesty', cenid='GR'),

Class(cid='GRms032023', clevel='ms', cdate=('03/20/23'), ctopic='Service', cenid='GR'),

Class(cid='GRms041023', clevel='ms', cdate=('04/10/23'), ctopic='Gratitude', cenid='GR'),

Class(cid='PWhs091522', clevel='hs', cdate=('09/15/22'), ctopic='Friendship', cenid='PW'),

Class(cid='PWhs100622', clevel='hs', cdate=('10/06/22'), ctopic='Courage', cenid='PW'),

Class(cid='PWhs102722', clevel='hs', cdate=('10/27/22'), ctopic='Study', cenid='PW'),

Class(cid='PWhs110322', clevel='hs', cdate=('11/03/22'), ctopic='Cheerfulness', cenid='PW'),

Class(cid='PWhs111722', clevel='hs', cdate=('11/17/22'), ctopic='Honesty', cenid='PW'),

Class(cid='PWhs120122', clevel='hs', cdate=('12/01/22'), ctopic='Leadership', cenid='PW'),

Class(cid='PWhs011923', clevel='hs', cdate=('01/19/23'), ctopic='Service', cenid='PW'),

Class(cid='PWhs020223', clevel='hs', cdate=('02/02/23'), ctopic='Prudence', cenid='PW'),

Class(cid='PWhs021623', clevel='hs', cdate=('02/16/23'), ctopic='Justice', cenid='PW'),

Class(cid='PWhs030223', clevel='hs', cdate=('03/02/23'), ctopic='Temperance', cenid='PW'),

Class(cid='PWhs031623', clevel='hs', cdate=('03/16/23'), ctopic='Fortitude', cenid='PW'),

Class(cid='PWhs040623', clevel='hs', cdate=('04/06/23'), ctopic='Respect', cenid='PW'),

Class(cid='PWhs042023', clevel='hs', cdate=('04/20/23'), ctopic='Motivation', cenid='PW'),

Class(cid='PWhs050423', clevel='hs', cdate=('05/04/23'), ctopic='Prayer', cenid='PW'),

Class(cid='PWhs051823', clevel='hs', cdate=('05/18/23'), ctopic='Gratitude', cenid='PW'),

Class(cid='SBhs093022', clevel='hs', cdate=('09/30/22'), ctopic='Friendship', cenid='SB'),

Class(cid='SBhs102822', clevel='hs', cdate=('10/28/22'), ctopic='Courage', cenid='SB'),

Class(cid='SBhs111822', clevel='hs', cdate=('11/18/22'), ctopic='Study', cenid='SB'),

Class(cid='SBhs121622', clevel='hs', cdate=('12/16/22'), ctopic='Cheerfulness', cenid='SB'),

Class(cid='SBhs012723', clevel='hs', cdate=('01/27/23'), ctopic='Leadership', cenid='SB'),

Class(cid='SBhs022423', clevel='hs', cdate=('02/24/23'), ctopic='Service', cenid='SB'),

Class(cid='SBhs033123', clevel='hs', cdate=('03/31/23'), ctopic='Respect', cenid='SB'),

Class(cid='SBhs042823', clevel='hs', cdate=('04/28/23'), ctopic='Motivation', cenid='SB'),

Class(cid='SBhs052623', clevel='hs', cdate=('05/26/23'), ctopic='Gratitude', cenid='SB'),

Class(cid='SLhs101222', clevel='hs', cdate=('10/12/22'), ctopic='Friendship', cenid='SL'),

Class(cid='SLhs102622', clevel='hs', cdate=('10/26/22'), ctopic='Study', cenid='SL'),

Class(cid='SLhs110922', clevel='hs', cdate=('11/09/22'), ctopic='Cheerfulness', cenid='SL'),

Class(cid='SLhs112322', clevel='hs', cdate=('11/23/22'), ctopic='Gratitude', cenid='SL'),

Class(cid='SLhs121422', clevel='hs', cdate=('12/14/22'), ctopic='Courage', cenid='SL'),

Class(cid='SLhs011123', clevel='hs', cdate=('01/11/23'), ctopic='Prudence', cenid='SL'),

Class(cid='SLhs012523', clevel='hs', cdate=('01/25/23'), ctopic='Justice', cenid='SL'),

Class(cid='SLhs020823', clevel='hs', cdate=('02/08/23'), ctopic='Temperance', cenid='SL'),

Class(cid='SLhs022223', clevel='hs', cdate=('02/22/23'), ctopic='Fortitude', cenid='SL'),

Class(cid='SLhs030823', clevel='hs', cdate=('03/08/23'), ctopic='Respect', cenid='SL'),

Class(cid='SLhs032223', clevel='hs', cdate=('03/22/23'), ctopic='Leadership', cenid='SL'),

Class(cid='SLhs041223', clevel='hs', cdate=('04/12/23'), ctopic='Service', cenid='SL'),

Class(cid='SLhs041923', clevel='hs', cdate=('04/19/23'), ctopic='Prayer', cenid='SL'),

Class(cid='SLms092622', clevel='ms', cdate=('09/26/22'), ctopic='Goal Setting', cenid='SL'),

Class(cid='SLms102422', clevel='ms', cdate=('10/24/22'), ctopic='Study', cenid='SL'),

Class(cid='SLms112822', clevel='ms', cdate=('11/28/22'), ctopic='Friendship', cenid='SL'),

Class(cid='SLms121922', clevel='ms', cdate=('12/19/22'), ctopic='Courage', cenid='SL'),

Class(cid='SLms013023', clevel='ms', cdate=('01/30/23'), ctopic='Cheerfulness', cenid='SL'),

Class(cid='SLms022723', clevel='ms', cdate=('02/27/23'), ctopic='Honesty', cenid='SL'),

Class(cid='SLms032723', clevel='ms', cdate=('03/27/23'), ctopic='Service', cenid='SL'),

Class(cid='SLms042423', clevel='ms', cdate=('04/24/23'), ctopic='Respect', cenid='SL'),

Class(cid='SHhs090722', clevel='hs', cdate=('09/07/22'), ctopic='Friendship', cenid='SH'),

Class(cid='SHhs110222', clevel='hs', cdate=('11/02/22'), ctopic='Study', cenid='SH'),

Class(cid='SHhs010423', clevel='hs', cdate=('01/04/23'), ctopic='Cheerfulness', cenid='SH'),

Class(cid='SHhs030723', clevel='hs', cdate=('03/07/23'), ctopic='Gratitude', cenid='SH'),

Class(cid='SHhs050223', clevel='hs', cdate=('05/02/23'), ctopic='Leadership', cenid='SH'),

Class(cid='SHco101322', clevel='co', cdate=('10/13/22'), ctopic='Working Well', cenid='SH'),

Class(cid='SHco111022', clevel='co', cdate=('11/10/22'), ctopic='Friendship', cenid='SH'),

Class(cid='SHco120822', clevel='co', cdate=('12/08/22'), ctopic='Fortitude', cenid='SH'),

Class(cid='SHco020223', clevel='co', cdate=('02/02/23'), ctopic='Justice', cenid='SH'),

Class(cid='SHco030123', clevel='co', cdate=('03/01/23'), ctopic='Temperance', cenid='SH'),

Class(cid='SHco040523', clevel='co', cdate=('04/05/23'), ctopic='Prudence', cenid='SH'),

Class(cid='SHco050323', clevel='co', cdate=('05/03/23'), ctopic='Leadership', cenid='SH')

]

# PermStaff

sample\_perm\_staff = [

PermStaff(pid='MaSmith', pfirst='Margaret', plast='Smith', pemail='margaretsmith87@gmail.com', pphonenumber='773-432-1234', ptrainstatus='initiated', ptrainexp=None, cenid='EL'),

PermStaff(pid='LiBennett', pfirst='Liam', plast='Bennett', pemail='liam.bennett\_123@yahoo.com', pphonenumber='815-123-4567', ptrainstatus='good', ptrainexp='2024-08-09', cenid='EL'),

PermStaff(pid='OlCarter', pfirst='Olivia', plast='Carter', pemail='oliviacarter22@hotmail.com', pphonenumber='773-234-5678', ptrainstatus='initiated', ptrainexp=None, cenid='EL'),

PermStaff(pid='NoDavis', pfirst='Noah', plast='Davis', pemail='noah.davis2000@gmail.com', pphonenumber='773-432-1098', ptrainstatus='good', ptrainexp='2024-03-08', cenid='EL'),

PermStaff(pid='EmEvans', pfirst='Emma', plast='Evans', pemail='e.evansofficial@gmail.com', pphonenumber='815-345-6789', ptrainstatus='initiated', ptrainexp=None, cenid='EL'),

PermStaff(pid='JaFisher', pfirst='Jackson', plast='Fisher', pemail='jackson.fisher\_99@yahoo.com', pphonenumber='773-876-5432', ptrainstatus='flag', ptrainexp='2023-07-02', cenid='EL'),

PermStaff(pid='AvGreene', pfirst='Ava', plast='Greene', pemail='ava.greene\_5678@hotmail.com', pphonenumber='815-987-6543', ptrainstatus='good', ptrainexp='2026-07-07', cenid='FG'),

PermStaff(pid='AiHayes', pfirst='Aiden', plast='Hayes', pemail='aidenh345@gmail.com', pphonenumber='815-789-0123', ptrainstatus='initiated', ptrainexp=None, cenid='FG'),

PermStaff(pid='IsIngram', pfirst='Isabella', plast='Ingram', pemail='isabella.ingram@outlook.com', pphonenumber='773-890-1234', ptrainstatus='good', ptrainexp='2024-09-06', cenid='FG'),

PermStaff(pid='MaSmith1', pfirst='Mary', plast='Smith', pemail='m.smithofficial@gmail.com', pphonenumber='773-678-9012', ptrainstatus='initiated', ptrainexp=None, cenid='LD'),

PermStaff(pid='LuJensen', pfirst='Lucas', plast='Jensen', pemail='lucas.jensen22@yahoo.com', pphonenumber='815-678-9012', ptrainstatus='good', ptrainexp='2024-07-25', cenid='LD'),

PermStaff(pid='MiKeller', pfirst='Mia', plast='Keller', pemail='mia.keller\_456@hotmail.com', pphonenumber='773-234-5678', ptrainstatus='initiated', ptrainexp=None, cenid='LD'),

PermStaff(pid='EtLawson', pfirst='Ethan', plast='Lawson', pemail='ethan.lawson\_789@gmail.com', pphonenumber='773-890-1234', ptrainstatus='good', ptrainexp='2026-08-12', cenid='LD'),

PermStaff(pid='HaMitchell', pfirst='Harper', plast='Mitchell', pemail='harper.mitchell\_321@yahoo.com', pphonenumber='815-789-0987', ptrainstatus='failed', ptrainexp=None, cenid='LD'),

PermStaff(pid='OlNelson', pfirst='Oliver', plast='Nelson', pemail='o.nelsonofficial@gmail.com', pphonenumber='815-345-6789', ptrainstatus='good', ptrainexp='2024-11-30', cenid='LD'),

PermStaff(pid='AmOwens', pfirst='Amelia', plast='Owens', pemail='amelia.owens\_22@hotmail.com', pphonenumber='773-234-5678', ptrainstatus='good', ptrainexp='2026-02-07', cenid='PW'),

PermStaff(pid='ElParker', pfirst='Elijah', plast='Parker', pemail='elijah.parker2001@yahoo.com', pphonenumber='773-678-9012', ptrainstatus='initiated', ptrainexp=None, cenid='PW'),

PermStaff(pid='EvQuinn', pfirst='Evelyn', plast='Quinn', pemail='evelyn.quinnofficial@gmail.com', pphonenumber='815-678-9012', ptrainstatus='good', ptrainexp='2025-10-14', cenid='PW'),

PermStaff(pid='JaReynolds', pfirst='James', plast='Reynolds', pemail='james\_reynolds22@yahoo.com', pphonenumber='773-432-1098', ptrainstatus='good', ptrainexp='2024-10-12', cenid='PW'),

PermStaff(pid='AbStewart', pfirst='Abigail', plast='Stewart', pemail='abigail.stewart\_567@hotmail.com', pphonenumber='815-987-6543', ptrainstatus='initiated', ptrainexp=None, cenid='PW'),

PermStaff(pid='BeTurner', pfirst='Benjamin', plast='Turner', pemail='benjamin\_turner@gmail.com', pphonenumber='773-876-5432', ptrainstatus='good', ptrainexp='2025-02-21', cenid='PW'),

PermStaff(pid='ChVaughn', pfirst='Charlotte', plast='Vaughn', pemail='c.vaughnofficial@yahoo.com', pphonenumber='815-123-4567', ptrainstatus='initiated', ptrainexp=None, cenid='SB'),

PermStaff(pid='WiWallace', pfirst='William', plast='Wallace', pemail='william.wallace\_123@hotmail.com', pphonenumber='815-789-0123', ptrainstatus='good', ptrainexp='2023-12-03', cenid='SB'),

PermStaff(pid='ScYoung', pfirst='Scarlett', plast='Young', pemail='s.youngofficial@gmail.com', pphonenumber='773-432-1098', ptrainstatus='failed', ptrainexp=None, cenid='SB'),

PermStaff(pid='HeZimmerman', pfirst='Henry', plast='Zimmerman', pemail='h.zimmerman\_789@yahoo.com', pphonenumber='773-432-1234', ptrainstatus='good', ptrainexp='2025-11-27', cenid='SB'),

PermStaff(pid='GrHarrison', pfirst='Grace', plast='Harrison', pemail='graceharrisonofficial@gmail.com', pphonenumber='773-890-1234', ptrainstatus='initiated', ptrainexp=None, cenid='SL'),

PermStaff(pid='AlDixon', pfirst='Alexander', plast='Dixon', pemail='alexander\_dixon22@hotmail.com', pphonenumber='815-123-4567', ptrainstatus='good', ptrainexp='2025-08-09', cenid='SL'),

PermStaff(pid='LiMason', pfirst='Lily', plast='Mason', pemail='lily.mason\_456@yahoo.com', pphonenumber='773-432-1098', ptrainstatus='flag', ptrainexp='2023-09-08', cenid='SL'),

PermStaff(pid='SaPorter', pfirst='Samuel', plast='Porter', pemail='samuel.porterofficial@gmail.com', pphonenumber='773-234-5678', ptrainstatus='good', ptrainexp='2024-02-18', cenid='SL'),

PermStaff(pid='ChReed', pfirst='Chloe', plast='Reed', pemail='chloe.reed\_22@yahoo.com', pphonenumber='815-345-6789', ptrainstatus='initiated', ptrainexp=None, cenid='SL'),

PermStaff(pid='DaSanders', pfirst='Daniel', plast='Sanders', pemail='daniel.sanders\_123@hotmail.com', pphonenumber='773-678-9012', ptrainstatus='good', ptrainexp='2026-06-20', cenid='SL'),

PermStaff(pid='ZoWarren', pfirst='Zoey', plast='Warren', pemail='zoey.warrenofficial@gmail.com', pphonenumber='815-789-0123', ptrainstatus='good', ptrainexp='2025-07-22', cenid='SL'),

PermStaff(pid='CaHughes', pfirst='Caleb', plast='Hughes', pemail='caleb.hughes\_567@yahoo.com', pphonenumber='773-876-5432', ptrainstatus='initiated', ptrainexp=None, cenid='SL'),

PermStaff(pid='ElRoss', pfirst='Ella', plast='Ross', pemail='ella\_ross22@hotmail.com', pphonenumber='815-987-6543', ptrainstatus='good', ptrainexp='2026-04-11', cenid='SH'),

PermStaff(pid='MaPowell', pfirst='Mason', plast='Powell', pemail='mason.powellofficial@gmail.com', pphonenumber='773-432-1234', ptrainstatus='flag', ptrainexp='2023-05-28', cenid='SH'),

PermStaff(pid='SoBennett', pfirst='Sofia', plast='Bennett', pemail='sofia.bennett\_321@yahoo.com', pphonenumber='815-789-0123', ptrainstatus='good', ptrainexp='2023-12-12', cenid='SH')

]

# Volunteer

sample\_volunteers = [

Volunteer(vid='V01', vfirst='Jane', vlast='Angel', vphonenumber='333-444-5567', vemail='janeangel1@gmail.com', vtrainstatus='good', vtrainexp='2024-10-16'),

Volunteer(vid='V02', vfirst='David', vlast='Smith', vphonenumber='111-224-3378', vemail='davidsmith@outlook.com', vtrainstatus='good', vtrainexp='2024-11-23'),

Volunteer(vid='V03', vfirst='Angela', vlast='Precious', vphonenumber='208-447-5583', vemail='angelaprecious3@hotmail.com', vtrainstatus='good', vtrainexp='2024-06-04'),

Volunteer(vid='V04', vfirst='Leroy', vlast='Prescot', vphonenumber='305-456-7890', vemail='leroyprescot14@gmail.com', vtrainstatus='good', vtrainexp='2025-02-24'),

Volunteer(vid='V05', vfirst='Adam', vlast='Jake', vphonenumber='224-527-8765', vemail='adamjake@msn.com', vtrainstatus='initiated', vtrainexp=None),

Volunteer(vid='V06', vfirst='Logan', vlast='Peter', vphonenumber='220-657-7890', vemail='loganpeter@outlook.com', vtrainstatus='good', vtrainexp='2024-08-19'),

Volunteer(vid='V07', vfirst='Toby', vlast='Tyler', vphonenumber='451-567-6578', vemail='tobytyler@gmail.com', vtrainstatus='initiated', vtrainexp=None),

Volunteer(vid='V08', vfirst='Tyson', vlast='Chandler', vphonenumber='312-876-4521', vemail='tysonchandler@msn.com', vtrainstatus='flag', vtrainexp='2024-09-14'),

Volunteer(vid='V09', vfirst='Mary', vlast='Lovely', vphonenumber='512-489-0081', vemail='marylovely@gmail.com', vtrainstatus='good', vtrainexp='2025-04-16'),

Volunteer(vid='V10', vfirst='James', vlast='Richard', vphonenumber='444-012-6890', vemail='jamesrichard12@outlook.com', vtrainstatus='good', vtrainexp='2024-07-15')

]

# Event

sample\_events = [

Event(eid='093022ms\_wksp', elocation='Grove', ename='Goal Setting', edate='2022-09-30', eaudience='ms', ecategory='workshop'),

Event(eid='012023ms\_wksp', elocation='Grove', ename='Study Techniques', edate='2023-01-20', eaudience='ms', ecategory='workshop'),

Event(eid='121622ms\_service', elocation='St. Anne''s', ename='Food Pantry', edate='2022-12-16', eaudience='ms', ecategory='service'),

Event(eid='093022hs\_workshop', elocation='Sherlake', ename='Optimal Work', edate='2022-09-30', eaudience='hs', ecategory='workshop'),

Event(eid='100722hs\_workshop', elocation='Grove', ename='Leadership', edate='2022-10-07', eaudience='hs', ecategory='workshop'),

Event(eid='041322hs\_workshop', elocation='Lindell', ename='Building Character', edate='2022-04-13', eaudience='hs', ecategory='workshop'),

Event(eid='031622hs\_workshop', elocation='Sherlake', ename='Leadership', edate='2022-03-16', eaudience='hs', ecategory='workshop'),

Event(eid='100622hs\_workshop', elocation='Sherlake', ename='Building Habits', edate='2022-10-06', eaudience='hs', ecategory='workshop'),

Event(eid='111822hs\_service', elocation='St. Anne''s', ename='Food Pantry', edate='2022-11-18', eaudience='hs', ecategory='service'),

Event(eid='051823hs\_service', elocation='Arbor Care', ename='Visit to Elderly', edate='2023-05-18', eaudience='hs', ecategory='service'),

Event(eid='110422hs\_retreat', elocation='Shellbourne', ename='HS Fall Retreat', edate='2022-11-04', eaudience='hs', ecategory='retreat'),

Event(eid='032323hs\_retreat', elocation='Shellbourne', ename='HS Spring Retreat', edate='2023-03-23', eaudience='hs', ecategory='retreat'),

Event(eid='121722hs\_other', elocation='Sherlake', ename='Christmas Party', edate='2022-12-17', eaudience='hs', ecategory='other'),

Event(eid='102822co\_workshop', elocation='Elms', ename='Optimal Work', edate='2022-10-28', eaudience='co', ecategory='workshop'),

Event(eid='120922co\_service', elocation='Elms', ename='Visit the Homeless', edate='2022-12-09', eaudience='co', ecategory='service'),

Event(eid='021023co\_service', elocation='Southold', ename='Visit the Homeless', edate='2023-02-10', eaudience='co', ecategory='service'),

Event(eid='032323co\_service', elocation='St. Anne''s', ename='Food Pantry', edate='2023-03-23', eaudience='co', ecategory='service'),

Event(eid='011323co\_retreat', elocation='Shellbourne', ename='College Winter Retreat', edate='2023-01-13', eaudience='co', ecategory='retreat'),

Event(eid='042023co\_retreat', elocation='Shellbourne', ename='College Spring Retreat', edate='2023-04-20', eaudience='co', ecategory='retreat'),

Event(eid='110622co\_other', elocation='Southold', ename='Dinner', edate='2022-11-06', eaudience='co', ecategory='other'),

Event(eid='032623co\_other', elocation='Southold', ename='Dinner', edate='2023-03-26', eaudience='co', ecategory='other'),

Event(eid='042823co\_other', elocation='Elms', ename='Movie Night', edate='2023-04-28', eaudience='co', ecategory='other')

]

# Takes

sample\_takes = [

Takes(cid='ELco101122', sid='LeKoch'),

Takes(cid='ELco101122', sid='RyWilliams'),

Takes(cid='ELco101122', sid='DaMartinez'),

Takes(cid='ELco110822', sid='OlSmith'),

Takes(cid='ELco110822', sid='LeKoch'),

Takes(cid='ELco110822', sid='SaJohnson'),

Takes(cid='ELco110822', sid='MaTaylor'),

Takes(cid='ELco011023', sid='AvBrown'),

Takes(cid='ELco011023', sid='OlSmith'),

Takes(cid='ELco021423', sid='LeKoch'),

Takes(cid='ELco021423', sid='RyWilliams'),

Takes(cid='ELco021423', sid='EmDavis'),

Takes(cid='ELco021423', sid='SaJohnson'),

Takes(cid='ELco031423', sid='DaMartinez'),

Takes(cid='ELco031423', sid='MaTaylor'),

Takes(cid='ELco031423', sid='OlSmith'),

Takes(cid='ELco041123', sid='LeKoch'),

Takes(cid='ELco041123', sid='RyWilliams'),

Takes(cid='ELco050923', sid='LeKoch'),

Takes(cid='ELco050923', sid='MaTaylor'),

Takes(cid='ELco050923', sid='SaJohnson'),

Takes(cid='GRhs091522', sid='ChAnderson'),

Takes(cid='GRhs091522', sid='HeAdams'),

Takes(cid='GRhs100622', sid='ChHall'),

Takes(cid='GRhs100622', sid='ChAnderson'),

Takes(cid='GRhs100622', sid='MiClark'),

Takes(cid='GRhs100622', sid='SoParker'),

Takes(cid='GRhs102022', sid='LiGreen'),

Takes(cid='GRhs102022', sid='AmRodriguez'),

Takes(cid='GRhs102022', sid='NoMitchell'),

Takes(cid='GRhs110322', sid='ChAnderson'),

Takes(cid='GRhs110322', sid='BeWilson'),

Takes(cid='GRhs110322', sid='SoParker'),

Takes(cid='GRhs111722', sid='MiClark'),

Takes(cid='GRhs111722', sid='SaWhite'),

Takes(cid='GRhs111722', sid='BeWilson'),

Takes(cid='GRhs111722', sid='NoMitchell'),

Takes(cid='GRhs120122', sid='BeWilson'),

Takes(cid='GRhs120122', sid='SoParker'),

Takes(cid='GRhs011923', sid='SaWhite'),

Takes(cid='GRhs011923', sid='MiClark'),

Takes(cid='GRhs011923', sid='AmRodriguez'),

Takes(cid='GRhs020223', sid='NoMitchell'),

Takes(cid='GRhs020223', sid='LiGreen'),

Takes(cid='GRhs030223', sid='BeWilson'),

Takes(cid='GRhs030223', sid='MiClark'),

Takes(cid='GRhs030223', sid='SaWhite'),

Takes(cid='GRhs031623', sid='AmRodriguez'),

Takes(cid='GRhs031623', sid='SoParker'),

Takes(cid='GRhs031623', sid='ChHall'),

Takes(cid='GRhs040623', sid='AmRodriguez'),

Takes(cid='GRhs040623', sid='MiClark'),

Takes(cid='GRhs040623', sid='SaWhite'),

Takes(cid='GRhs042023', sid='NoMitchell'),

Takes(cid='GRhs042023', sid='SoParker'),

Takes(cid='GRhs042023', sid='MiClark'),

Takes(cid='GRhs050423', sid='BeWilson'),

Takes(cid='GRhs050423', sid='MiClark'),

Takes(cid='GRhs050423', sid='AmRodriguez'),

Takes(cid='GRhs050423', sid='SoParker'),

Takes(cid='GRms091922', sid='IsRodriguez'),

Takes(cid='GRms091922', sid='JaThomas'),

Takes(cid='GRms101722', sid='AvAnderson'),

Takes(cid='GRms101722', sid='IsRodriguez'),

Takes(cid='GRms112122', sid='IsRodriguez'),

Takes(cid='GRms112122', sid='JaThomas'),

Takes(cid='GRms112122', sid='OlBrown'),

Takes(cid='GRms121922', sid='AvAnderson'),

Takes(cid='GRms121922', sid='IsRodriguez'),

Takes(cid='GRms011623', sid='MiSmith'),

Takes(cid='GRms011623', sid='IsRodriguez'),

Takes(cid='GRms011623', sid='AvAnderson'),

Takes(cid='GRms022023', sid='WiHarris'),

Takes(cid='GRms022023', sid='LiWilson'),

Takes(cid='GRms041023', sid='AvAnderson'),

Takes(cid='GRms041023', sid='JaThomas'),

Takes(cid='PWhs091522', sid='ZoCooper'),

Takes(cid='PWhs091522', sid='LoWard'),

Takes(cid='PWhs102722', sid='ZoCooper'),

Takes(cid='PWhs102722', sid='DaCarter'),

Takes(cid='PWhs102722', sid='CaMitchell'),

Takes(cid='PWhs102722', sid='EmRivera'),

Takes(cid='PWhs110322', sid='EmRivera'),

Takes(cid='PWhs110322', sid='ZoCooper'),

Takes(cid='PWhs111722', sid='ZoCooper'),

Takes(cid='PWhs111722', sid='EmRivera'),

Takes(cid='PWhs120122', sid='ZoCooper'),

Takes(cid='PWhs120122', sid='DaCarter'),

Takes(cid='PWhs020223', sid='LiScott'),

Takes(cid='PWhs020223', sid='LoWard'),

Takes(cid='PWhs021623', sid='ZoCooper'),

Takes(cid='PWhs021623', sid='DaCarter'),

Takes(cid='PWhs030223', sid='EmRivera'),

Takes(cid='PWhs030223', sid='ZoCooper'),

Takes(cid='PWhs031623', sid='JaStewart'),

Takes(cid='PWhs031623', sid='DaCarter'),

Takes(cid='PWhs031623', sid='AnRoss'),

Takes(cid='PWhs042023', sid='ZoCooper'),

Takes(cid='PWhs042023', sid='ElYoung'),

Takes(cid='PWhs050423', sid='ZoCooper'),

Takes(cid='PWhs050423', sid='EmRivera'),

Takes(cid='PWhs051823', sid='CaMitchell'),

Takes(cid='PWhs051823', sid='ZoCooper'),

Takes(cid='SBhs093022', sid='EmSullivan'),

Takes(cid='SBhs093022', sid='MaHernandez'),

Takes(cid='SBhs102822', sid='LuWilson'),

Takes(cid='SBhs111822', sid='ZaParker'),

Takes(cid='SBhs111822', sid='MiStewart'),

Takes(cid='SBhs111822', sid='EmSullivan'),

Takes(cid='SBhs121622', sid='AbMorris'),

Takes(cid='SBhs012723', sid='EmSullivan'),

Takes(cid='SBhs012723', sid='MaHernandez'),

Takes(cid='SBhs012723', sid='OwCarter'),

Takes(cid='SBhs022423', sid='EmSullivan'),

Takes(cid='SBhs022423', sid='ZaParker'),

Takes(cid='SBhs033123', sid='AbMorris'),

Takes(cid='SBhs033123', sid='JoPeterson'),

Takes(cid='SBhs042823', sid='JoPeterson'),

Takes(cid='SBhs042823', sid='EmSullivan'),

Takes(cid='SBhs042823', sid='ZaParker'),

Takes(cid='SBhs052623', sid='NiJohnson'),

Takes(cid='SBhs052623', sid='ZaParker'),

Takes(cid='SLhs101222', sid='EmSchneider'),

Takes(cid='SLhs101222', sid='AnEllis'),

Takes(cid='SLhs102622', sid='ClHammond'),

Takes(cid='SLhs102622', sid='LeYates'),

Takes(cid='SLhs102622', sid='CeCollier'),

Takes(cid='SLhs110922', sid='CaFlores'),

Takes(cid='SLhs110922', sid='TaOsborne'),

Takes(cid='SLhs110922', sid='EmSchneider'),

Takes(cid='SLhs112322', sid='EmSchneider'),

Takes(cid='SLhs112322', sid='LeMorris'),

Takes(cid='SLhs121422', sid='ClHammond'),

Takes(cid='SLhs011123', sid='LoCunningham'),

Takes(cid='SLhs011123', sid='CeCollier'),

Takes(cid='SLhs011123', sid='EmSchneider'),

Takes(cid='SLhs012523', sid='JoJohnson'),

Takes(cid='SLhs012523', sid='LoCunningham'),

Takes(cid='SLhs020823', sid='EmSchneider'),

Takes(cid='SLhs020823', sid='LeMorris'),

Takes(cid='SLhs022223', sid='LeYates'),

Takes(cid='SLhs022223', sid='PaWhite'),

Takes(cid='SLhs022223', sid='CaFlores'),

Takes(cid='SLhs030823', sid='EmSchneider'),

Takes(cid='SLhs032223', sid='EmSchneider'),

Takes(cid='SLhs032223', sid='LoCunningham'),

Takes(cid='SLhs041223', sid='AnEllis'),

Takes(cid='SLhs041923', sid='EmSchneider'),

Takes(cid='SLhs041923', sid='ClHammond'),

Takes(cid='SLms092622', sid='NoClark'),

Takes(cid='SLms092622', sid='LiNyugen'),

Takes(cid='SLms102422', sid='StGuzman'),

Takes(cid='SLms102422', sid='StHoffman'),

Takes(cid='SLms102422', sid='NoClark'),

Takes(cid='SLms112822', sid='DaMontgomery'),

Takes(cid='SLms112822', sid='NoClark'),

Takes(cid='SLms121922', sid='BrMitchell'),

Takes(cid='SLms121922', sid='JaLee'),

Takes(cid='SLms121922', sid='NoClark'),

Takes(cid='SLms013023', sid='StGuzman'),

Takes(cid='SLms013023', sid='SaGardner'),

Takes(cid='SLms013023', sid='StHoffman'),

Takes(cid='SLms013023', sid='NoClark'),

Takes(cid='SLms022723', sid='NoClark'),

Takes(cid='SLms022723', sid='JaLee'),

Takes(cid='SLms032723', sid='JaLee'),

Takes(cid='SLms032723', sid='StGuzman'),

Takes(cid='SLms042423', sid='SaGardner'),

Takes(cid='SLms042423', sid='NoClark'),

Takes(cid='SHhs090722', sid='NoWarner'),

Takes(cid='SHhs090722', sid='LeRuiz'),

Takes(cid='SHhs090722', sid='FaCasey'),

Takes(cid='SHhs110222', sid='NoWarner'),

Takes(cid='SHhs110222', sid='LeRuiz'),

Takes(cid='SHhs010423', sid='NoWarner'),

Takes(cid='SHhs010423', sid='LeRuiz'),

Takes(cid='SHhs010423', sid='FaCasey'),

Takes(cid='SHhs030723', sid='FaCasey'),

Takes(cid='SHhs050223', sid='NoWarner'),

Takes(cid='SHhs050223', sid='FaCasey'),

Takes(cid='SHco101322', sid='HaMcguire'),

Takes(cid='SHco101322', sid='MaRoberts'),

Takes(cid='SHco111022', sid='HaMcguire'),

Takes(cid='SHco120822', sid='KaFleming'),

Takes(cid='SHco120822', sid='HaMcguire'),

Takes(cid='SHco020223', sid='LuGomez'),

Takes(cid='SHco020223', sid='HaMcguire'),

Takes(cid='SHco030123', sid='KaFleming'),

Takes(cid='SHco030123', sid='LuGomez'),

Takes(cid='SHco040523', sid='HaMcguire'),

Takes(cid='SHco040523', sid='CaBlake'),

Takes(cid='SHco050323', sid='HaMcguire'),

Takes(cid='SHco050323', sid='MaHudson'),

]

# Teaches

sample\_teaches = [

Teaches(cid='ELco101122', pid='LiBennett'),

Teaches(cid='ELco110822', pid='EmEvans'),

Teaches(cid='ELco121322', pid='MaSmith'),

Teaches(cid='ELco011023', pid='MiKeller'),

Teaches(cid='ELco021423', pid='JaReynolds'),

Teaches(cid='ELco031423', pid='WiWallace'),

Teaches(cid='ELco041123', pid='AlDixon'),

Teaches(cid='ELco050923', pid='ChReed'),

Teaches(cid='GRhs091522', pid='CaHughes'),

Teaches(cid='GRhs100622', pid='MaSmith'),

Teaches(cid='GRhs102022', pid='NoDavis'),

Teaches(cid='GRhs110322', pid='AvGreene'),

Teaches(cid='GRhs111722', pid='IsIngram'),

Teaches(cid='GRhs120122', pid='LuJensen'),

Teaches(cid='GRhs011923', pid='EtLawson'),

Teaches(cid='GRhs020223', pid='OlNelson'),

Teaches(cid='GRhs021623', pid='AmOwens'),

Teaches(cid='GRhs030223', pid='ElParker'),

Teaches(cid='GRhs031623', pid='AbStewart'),

Teaches(cid='GRhs040623', pid='WiWallace'),

Teaches(cid='GRhs042023', pid='HeZimmerman'),

Teaches(cid='GRhs050423', pid='GrHarrison'),

Teaches(cid='GRms091922', pid='AlDixon'),

Teaches(cid='GRms091922', pid='LiMason'),

Teaches(cid='GRms101722', pid='SaPorter'),

Teaches(cid='GRms101722', pid='DaSanders'),

Teaches(cid='GRms112122', pid='ZoWarren'),

Teaches(cid='GRms112122', pid='CaHughes'),

Teaches(cid='GRms112122', pid='ElRoss'),

Teaches(cid='GRms121922', pid='LiBennett'),

Teaches(cid='GRms121922', pid='AvGreene'),

Teaches(cid='GRms011623', pid='AiHayes'),

Teaches(cid='GRms011623', pid='IsIngram'),

Teaches(cid='GRms022023', pid='MaSmith1'),

Teaches(cid='GRms032023', pid='LuJensen'),

Teaches(cid='GRms032023', pid='MiKeller'),

Teaches(cid='GRms032023', pid='EtLawson'),

Teaches(cid='GRms041023', pid='ChVaughn'),

Teaches(cid='GRms041023', pid='HeZimmerman'),

Teaches(cid='GRms041023', pid='GrHarrison'),

Teaches(cid='PWhs091522', pid='AlDixon'),

Teaches(cid='PWhs100622', pid='LiMason'),

Teaches(cid='PWhs102722', pid='SaPorter'),

Teaches(cid='PWhs102722', pid='ChReed'),

Teaches(cid='PWhs111722', pid='CaHughes'),

Teaches(cid='PWhs111722', pid='ElRoss'),

Teaches(cid='PWhs120122', pid='SoBennett'),

Teaches(cid='PWhs120122', pid='OlNelson'),

Teaches(cid='PWhs120122', pid='AmOwens'),

Teaches(cid='PWhs020223', pid='BeTurner'),

Teaches(cid='PWhs021623', pid='ScYoung'),

Teaches(cid='PWhs021623', pid='HeZimmerman'),

Teaches(cid='PWhs021623', pid='AlDixon'),

Teaches(cid='PWhs030223', pid='LiMason'),

Teaches(cid='PWhs030223', pid='SaPorter'),

Teaches(cid='PWhs031623', pid='ChReed'),

Teaches(cid='PWhs031623', pid='ZoWarren'),

Teaches(cid='PWhs040623', pid='CaHughes'),

Teaches(cid='PWhs040623', pid='ElRoss'),

Teaches(cid='PWhs042023', pid='LiBennett'),

Teaches(cid='PWhs042023', pid='AvGreene'),

Teaches(cid='PWhs050423', pid='AiHayes'),

Teaches(cid='PWhs050423', pid='IsIngram'),

Teaches(cid='SBhs093022', pid='MaSmith1'),

Teaches(cid='SBhs093022', pid='LuJensen'),

Teaches(cid='SBhs102822', pid='MiKeller'),

Teaches(cid='SBhs102822', pid='EtLawson'),

Teaches(cid='SBhs111822', pid='ChVaughn'),

Teaches(cid='SBhs111822', pid='HeZimmerman'),

Teaches(cid='SBhs121622', pid='GrHarrison'),

Teaches(cid='SBhs121622', pid='LiMason'),

Teaches(cid='SBhs012723', pid='SaPorter'),

Teaches(cid='SBhs022423', pid='ChReed'),

Teaches(cid='SBhs022423', pid='DaSanders'),

Teaches(cid='SBhs033123', pid='OlNelson'),

Teaches(cid='SBhs033123', pid='BeTurner'),

Teaches(cid='SBhs042823', pid='ChVaughn'),

Teaches(cid='SBhs042823', pid='WiWallace'),

Teaches(cid='SBhs052623', pid='ScYoung'),

Teaches(cid='SBhs052623', pid='HeZimmerman'),

Teaches(cid='SLhs101222', pid='GrHarrison'),

Teaches(cid='SLhs101222', pid='LiMason'),

Teaches(cid='SLhs102622', pid='SaPorter'),

Teaches(cid='SLhs102622', pid='ChReed'),

Teaches(cid='SLhs110922', pid='DaSanders'),

Teaches(cid='SLhs110922', pid='WiWallace'),

Teaches(cid='SLhs112322', pid='HeZimmerman'),

Teaches(cid='SLhs112322', pid='GrHarrison'),

Teaches(cid='SLhs121422', pid='AlDixon'),

Teaches(cid='SLhs121422', pid='LiMason'),

Teaches(cid='SLhs011123', pid='SaPorter'),

Teaches(cid='SLhs011123', pid='ChReed'),

Teaches(cid='SLhs012523', pid='DaSanders'),

Teaches(cid='SLhs020823', pid='SaPorter'),

Teaches(cid='SLhs020823', pid='ChReed'),

Teaches(cid='SLhs020823', pid='DaSanders'),

Teaches(cid='SLhs022223', pid='GrHarrison'),

Teaches(cid='SLhs022223', pid='AlDixon'),

Teaches(cid='SLhs030823', pid='LiMason'),

Teaches(cid='SLhs030823', pid='WiWallace'),

Teaches(cid='SLms092622', pid='ScYoung'),

Teaches(cid='SLms092622', pid='HeZimmerman'),

Teaches(cid='SLms092622', pid='GrHarrison'),

Teaches(cid='SLms102422', pid='AlDixon'),

Teaches(cid='SLms102422', pid='LiMason'),

Teaches(cid='SLms102422', pid='SaPorter'),

Teaches(cid='SLms112822', pid='ChReed'),

Teaches(cid='SLms112822', pid='DaSanders'),

Teaches(cid='SLms112822', pid='ZoWarren'),

Teaches(cid='SLms121922', pid='CaHughes'),

Teaches(cid='SLms121922', pid='ElRoss'),

Teaches(cid='SLms121922', pid='MaPowell'),

Teaches(cid='SLms013023', pid='SoBennett'),

Teaches(cid='SLms013023', pid='AmOwens'),

Teaches(cid='SLms013023', pid='OlNelson'),

Teaches(cid='SLms022723', pid='BeTurner'),

Teaches(cid='SLms022723', pid='ChVaughn'),

Teaches(cid='SLms022723', pid='WiWallace'),

Teaches(cid='SHhs090722', pid='HeZimmerman'),

Teaches(cid='SHhs090722', pid='GrHarrison'),

Teaches(cid='SHhs090722', pid='LiMason'),

Teaches(cid='SHhs110222', pid='SaPorter'),

Teaches(cid='SHhs110222', pid='ChReed'),

Teaches(cid='SHhs110222', pid='DaSanders'),

Teaches(cid='SHhs010423', pid='WiWallace'),

Teaches(cid='SHhs010423', pid='ScYoung'),

Teaches(cid='SHhs010423', pid='HeZimmerman'),

Teaches(cid='SHhs030723', pid='GrHarrison'),

Teaches(cid='SHhs030723', pid='AlDixon'),

Teaches(cid='SHhs030723', pid='LiMason'),

Teaches(cid='SHhs050223', pid='SaPorter'),

Teaches(cid='SHhs050223', pid='ChReed'),

Teaches(cid='SHhs050223', pid='DaSanders'),

Teaches(cid='SHco101322', pid='SaPorter'),

Teaches(cid='SHco101322', pid='ChReed'),

Teaches(cid='SHco101322', pid='DaSanders'),

Teaches(cid='SHco111022', pid='GrHarrison'),

Teaches(cid='SHco111022', pid='AlDixon'),

Teaches(cid='SHco111022', pid='LiMason'),

Teaches(cid='SHco120822', pid='WiWallace'),

Teaches(cid='SHco120822', pid='ScYoung'),

Teaches(cid='SHco120822', pid='HeZimmerman'),

Teaches(cid='SHco020223', pid='GrHarrison'),

Teaches(cid='SHco020223', pid='AlDixon'),

Teaches(cid='SHco020223', pid='LiMason'),

Teaches(cid='SHco030123', pid='SaPorter'),

Teaches(cid='SHco030123', pid='ChReed'),

Teaches(cid='SHco030123', pid='DaSanders'),

Teaches(cid='SHco040523', pid='ZoWarren'),

Teaches(cid='SHco040523', pid='CaHughes'),

Teaches(cid='SHco050323', pid='ElRoss'),

]

# Runs

sample\_runs = [

Runs(pid='MaSmith', eid='093022ms\_wksp'),

Runs(pid='LiBennett', eid='093022ms\_wksp'),

Runs(pid='OlCarter', eid='012023ms\_wksp'),

Runs(pid='IsIngram', eid='012023ms\_wksp'),

Runs(pid='NoDavis', eid='121622ms\_service'),

Runs(pid='MaSmith', eid='121622ms\_service'),

Runs(pid='EmEvans', eid='093022hs\_workshop'),

Runs(pid='JaFisher', eid='093022hs\_workshop'),

Runs(pid='LiMason', eid='093022hs\_workshop'),

Runs(pid='AvGreene', eid='100722hs\_workshop'),

Runs(pid='AiHayes', eid='100722hs\_workshop'),

Runs(pid='IsIngram', eid='041322hs\_workshop'),

Runs(pid='MaSmith1', eid='041322hs\_workshop'),

Runs(pid='LuJensen', eid='031622hs\_workshop'),

Runs(pid='IsIngram', eid='031622hs\_workshop'),

Runs(pid='AbStewart', eid='100622hs\_workshop'),

Runs(pid='MiKeller', eid='100622hs\_workshop'),

Runs(pid='EtLawson', eid='111822hs\_service'),

Runs(pid='HaMitchell', eid='111822hs\_service'),

Runs(pid='OlNelson', eid='111822hs\_service'),

Runs(pid='IsIngram', eid='051823hs\_service'),

Runs(pid='AmOwens', eid='051823hs\_service'),

Runs(pid='ElParker', eid='051823hs\_service'),

Runs(pid='WiWallace', eid='110422hs\_retreat'),

Runs(pid='EvQuinn', eid='110422hs\_retreat'),

Runs(pid='ElParker', eid='110422hs\_retreat'),

Runs(pid='JaReynolds', eid='110422hs\_retreat'),

Runs(pid='AbStewart', eid='110422hs\_retreat'),

Runs(pid='ScYoung', eid='121722hs\_other'),

Runs(pid='BeTurner', eid='032323hs\_retreat'),

Runs(pid='ChVaughn', eid='032323hs\_retreat'),

Runs(pid='WiWallace', eid='032323hs\_retreat'),

Runs(pid='HeZimmerman', eid='121722hs\_other'),

Runs(pid='GrHarrison', eid='102822co\_workshop'),

Runs(pid='AlDixon', eid='102822co\_workshop'),

Runs(pid='ScYoung', eid='120922co\_service'),

Runs(pid='LiMason', eid='120922co\_service'),

Runs(pid='SaPorter', eid='021023co\_service'),

Runs(pid='ScYoung', eid='021023co\_service'),

Runs(pid='EtLawson', eid='032323co\_service'),

Runs(pid='LiMason', eid='032323co\_service'),

Runs(pid='DaSanders', eid='011323co\_retreat'),

Runs(pid='ZoWarren', eid='011323co\_retreat'),

Runs(pid='HaMitchell', eid='042023co\_retreat'),

Runs(pid='CaHughes', eid='042023co\_retreat'),

Runs(pid='ElRoss', eid='110622co\_other'),

Runs(pid='DaSanders', eid='110622co\_other'),

Runs(pid='IsIngram', eid='032623co\_other'),

Runs(pid='ChVaughn', eid='032623co\_other'),

Runs(pid='JaReynolds', eid='042823co\_other'),

Runs(pid='ElParker', eid='042823co\_other'),

Runs(pid='HeZimmerman', eid='042823co\_other'),

]

# Attends

sample\_attends = [

Attends(sid='ChAnderson', eid='093022ms\_wksp'),

Attends(sid='AvAnderson', eid='012023ms\_wksp'),

Attends(sid='AnEllis', eid='121622ms\_service'),

Attends(sid='JaMorris', eid='093022hs\_workshop'),

Attends(sid='LoShaw', eid='100722hs\_workshop'),

Attends(sid='HaMcguire', eid='041322hs\_workshop'),

Attends(sid='AvBrown', eid='031622hs\_workshop'),

Attends(sid='KaFleming', eid='100622hs\_workshop'),

Attends(sid='EmDavis', eid='111822hs\_service'),

Attends(sid='MaHudson', eid='051823hs\_service'),

Attends(sid='LeYates', eid='110422hs\_retreat'),

Attends(sid='BrMitchell', eid='032323hs\_retreat'),

Attends(sid='JaLee', eid='121722hs\_other'),

Attends(sid='FaCasey', eid='102822co\_workshop'),

Attends(sid='FaCasey', eid='120922co\_service'),

Attends(sid='MaBecker', eid='021023co\_service'),

Attends(sid='BrMitchell', eid='032323co\_service'),

Attends(sid='LiWood', eid='011323co\_retreat'),

Attends(sid='DaCarter', eid='042023co\_retreat'),

Attends(sid='LoMiller', eid='110622co\_other'),

Attends(sid='ZaParker', eid='032623co\_other'),

Attends(sid='MaRoberts', eid='042823co\_other'),

Attends(sid='JaMorris', eid='093022ms\_wksp'),

Attends(sid='ChAnderson', eid='012023ms\_wksp'),

Attends(sid='EmDavis', eid='121622ms\_service'),

Attends(sid='ChAnderson', eid='093022hs\_workshop'),

Attends(sid='JaMoore', eid='100722hs\_workshop'),

Attends(sid='ClHammond', eid='041322hs\_workshop'),

Attends(sid='OlBrown', eid='031622hs\_workshop'),

Attends(sid='AnEllis', eid='100622hs\_workshop'),

Attends(sid='CeCollier', eid='111822hs\_service'),

Attends(sid='KaFleming', eid='051823hs\_service'),

Attends(sid='SoParker', eid='110422hs\_retreat'),

Attends(sid='SaGardner', eid='032323hs\_retreat'),

Attends(sid='ElYoung', eid='121722hs\_other'),

Attends(sid='CaFlores', eid='102822co\_workshop'),

Attends(sid='NoClark', eid='120922co\_service'),

Attends(sid='ZoCooper', eid='021023co\_service'),

Attends(sid='WiHarris', eid='032323co\_service'),

Attends(sid='SoMorgan', eid='011323co\_retreat'),

]

# Helps

sample\_helps = [

Helps(vid='V01', eid='093022hs\_workshop'),

Helps(vid='V05', eid='102822co\_workshop'),

Helps(vid='V10', eid='032323co\_service'),

Helps(vid='V08', eid='110622co\_other'),

Helps(vid='V03', eid='042823co\_other'),

Helps(vid='V01', eid='102822co\_workshop'),

Helps(vid='V10', eid='042823co\_other'),

Helps(vid='V08', eid='032323co\_service'),

Helps(vid='V08', eid='110422hs\_retreat'),

Helps(vid='V03', eid='100722hs\_workshop'),

]

# Commit the changes to the database

session.add\_all(sample\_centers)

session.add\_all(sample\_students)

session.add\_all(sample\_volunteers)

session.add\_all(sample\_perm\_staff)

session.add\_all(sample\_classes)

session.add\_all(sample\_events)

session.add\_all(sample\_helps)

session.add\_all(sample\_runs)

session.add\_all(sample\_takes)

session.add\_all(sample\_teaches)

session.add\_all(sample\_attends)

session.commit()

from sqlalchemy.orm import sessionmaker

from sqlalchemy import func

# Create a session

Session = sessionmaker(bind=engine)

session = Session()

# Query: PopularTopics

# Student: Rachel

# Description: Top 10 most popular class topics by center and total number of students attending those classes. Ordered by number of students in descending order/

popularTopics = (

session.query(

Center.cenname.label("CenterName"),

Class.ctopic.label("ClassTopic"),

func.count(func.distinct(Student.sid)).label('num\_students')

)

.join(Class, Class.cenid == Center.cenid)

.join(Takes, Takes.cid == Class.cid)

.join(Student, Student.sid == Takes.sid)

.group\_by(Center.cenid, Center.cenname, Class.ctopic)

.order\_by(func.count(Student.sid).desc())

.limit(10)

)

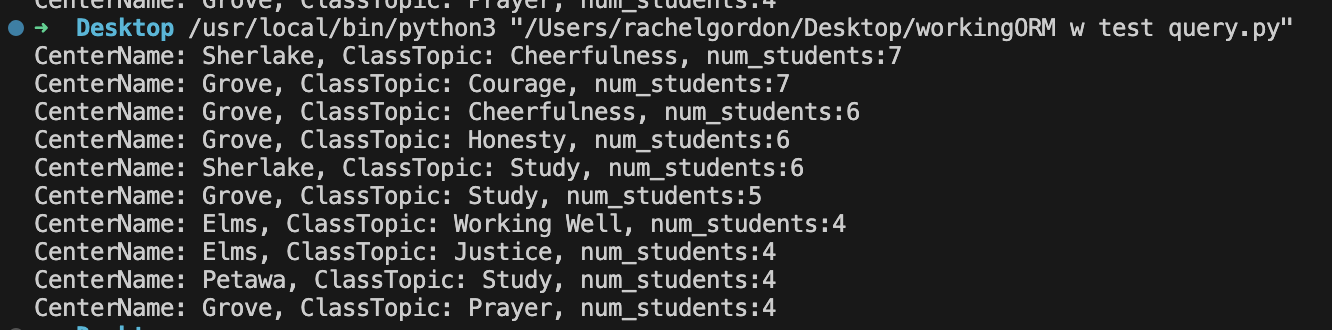
print('\n#Query: popularTopics (Rachel)#')

for row in popularTopics:

print(

f"CenterName: {row.CenterName}, ClassTopic: {row.ClassTopic}, num\_students:{row.num\_students}"

)



# Query: DistinctStudentsAttendingClasses

# Student: Taylor

# Description: For each center at each class level, find the number of distinct students taking classes.

# Order by descending student count, then center name.

distinctStudentsAttendingClasses = (

select(

Center.cenname,

Class.clevel,

func.count(func.distinct(Takes.sid)).label("student\_count")

)

.join(Class, Center.cenid == Class.cenid)

.join(Takes, Class.cid == Takes.cid)

.group\_by(Center.cenid, Center.cenname, Class.clevel)

#order by descending student\_count, then center name

.order\_by(desc(func.count(func.distinct(Takes.sid))), Center.cenname)

)

print('\n#Query: DistinctStudentsAttendingClasses (Taylor)#')

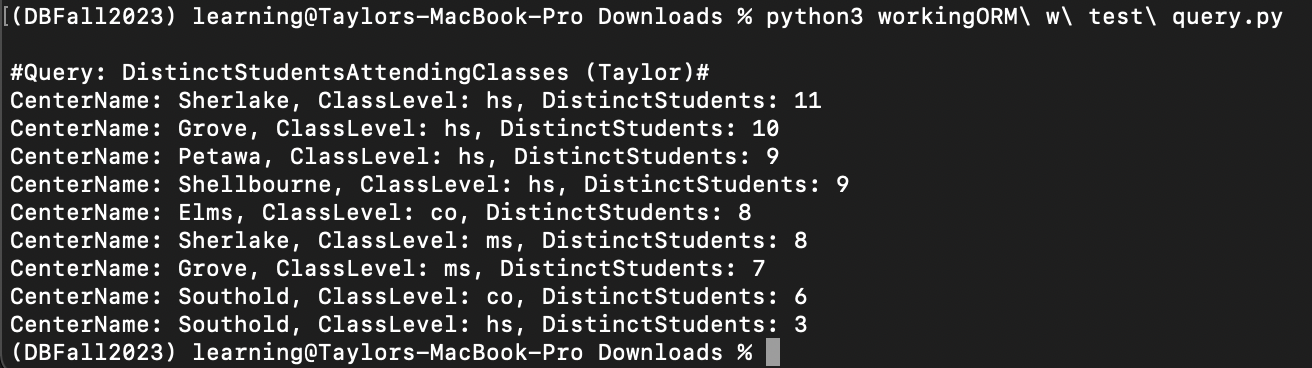
student\_counts = session.execute(distinctStudentsAttendingClasses)

for entry in student\_counts:

print(

f"CenterName: {entry.cenname}, ClassLevel: {entry.clevel}, DistinctStudents: {entry.student\_count}"

)



# Query: EventVolunteer

# Student: David

# Description: Extracts volunteer ID, first name, training status, and phone number for volunteers helping at each event and orders by first name in ascending order.

eventVolunteer = (

select(

Event.ename,

Volunteer.vid,

Volunteer.vfirst,

Volunteer.vtrainstatus,

Volunteer.vphonenumber,

)

.join(Helps, Volunteer.vid == Helps.vid)

.join(Event, Helps.eid == Event.eid)

.order\_by(Volunteer.vfirst.asc())

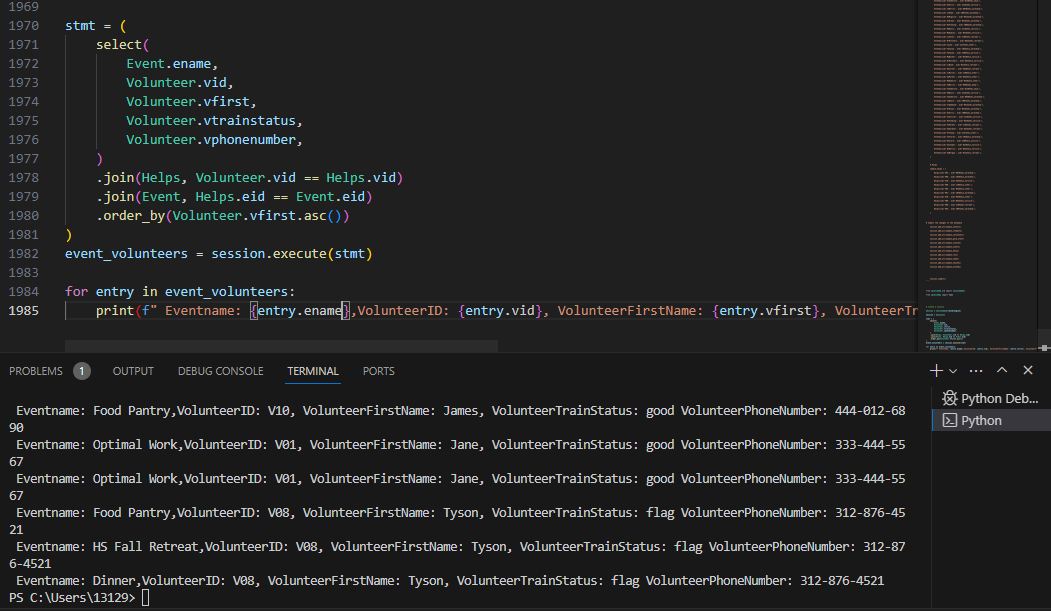
)

print('\n#Query: eventVolunteer (David)#')

event\_volunteers = session.execute(eventVolunteer)

for entry in event\_volunteers:

print(f"Eventname: {entry.ename},VolunteerID: {entry.vid}, VolunteerFirstName: {entry.vfirst}, VolunteerTrainStatus: {entry.vtrainstatus} VolunteerPhoneNumber: {entry.vphonenumber}")



#Query: NumClassesTaught

# Student: Kadin

# Description: he query counts the number of classes each permStaff teaches

permStaffTeaches = (

session.query(

PermStaff.pid,

PermStaff.pfirst,

PermStaff.plast,

func.count(Teaches.cid).label('count')

)

.join(Teaches, Teaches.pid == PermStaff.pid)

.filter(PermStaff.ptrainstatus == 'good')

.group\_by(PermStaff.pid, PermStaff.pfirst, PermStaff.plast)

.order\_by(func.count(Teaches.cid).desc())

.all()

)

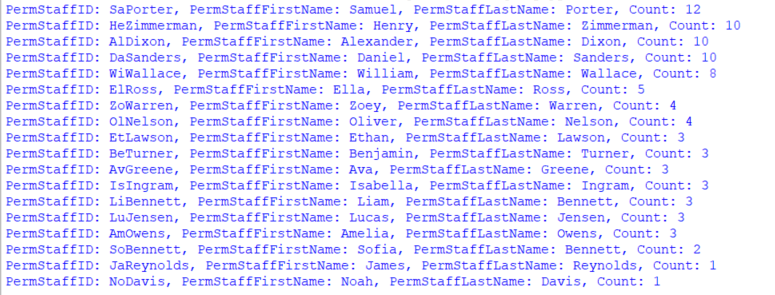
print('\n#Query: permStaffTeaches (Kadin)#')

for entry in permStaffTeaches:

print(f"PermStaffID: {entry.pid}, PermStaffFirstName: {entry.pfirst}, PermStaffLastName: {entry.plast}, "

f"Count: {entry.count}"

)

yh

# EventAttendanceAndStaff

# Student: Max Simmer

# Description: Pulls each event and the number of students that attend the event as well as the number of staff working the event.

eventAttendeesAndStaff = (

session.query(

Event.eid.label('EventID'),

Event.ename.label('EventName'),

func.count(func.distinct(Attends.sid)).label('numberOfAttendees'),

func.count(func.distinct(Runs.pid)).label('numberOfPermStaff')

)

.join(Attends, Attends.eid == Event.eid)

.join(Runs, Runs.eid == Event.eid)

.group\_by(Event.eid, Event.ename)

.order\_by(func.count(func.distinct(Attends.sid)).desc())

)

print('\n#Query: eventAttendeesAndStaff (Max)')

for row in eventAttendeesAndStaff:

print(

f"EventID: {row.EventID}, EventName: {row.EventName}, "

f"numberOfAttendees: {row.numberOfAttendees}, numberOfPermStaff: {row.numberOfPermStaff}"

)



session.close()

**GROUP STATUS REPORT**

**GROUP #:** 1 **GROUP NAME:** YOUTHPROGRAM **PHASE #: 3**

Dates & attendance at group meetings in this phase:

Tuesday, Nov 14 7:30-8:30PM All present

Wednesday, Nov 15 8:45-10pm All present

Monday, Nov 20 7:30-8:30pm All present

Monday, Nov 27 5:30-6:30pm All present

Phase 3 Deliverable is complete. The YOUTHPROGRAM Database is complete and functional. Listed below are the class and query responsibilities completed by each group member for the ORM implementation.

CONTRIBUTIONS OF GROUP MEMBERS

Leader: Rachel Gordon

1. Coordinated meeting times and deadlines for task completion
2. Kept report up to date
3. Helped debug queries
4. Class: PermStaff
5. QUERY IMPLEMENTATION: PopularTopics

Recorder: Kadin Suga

1. Class: Class + Teaches
2. Assisted various team members on running their code
3. Advised on how code should be structured
4. QUERY IMPLEMENTATION: NumClassesTaught

Phase Checker: Max Simmer

1. Combine code for all the classes.
2. Format and insert data.
3. Combine queries and final code.
4. Class: Event + Runs
5. QUERY IMPLEMENTATION: EventAttendanceAndStaff

Technical Advisor: David Sheleru

1. Helped implement code for ORM implementation

* Class: Volunteer + Helps

1. QUERY IMPLEMENTATION: EventVolunteer

Technical Advisor: Taylor Fischer

1. Completed the final editing and stitching together of relationships between classes
2. Class: Center & Student
3. QUERY IMPLEMENTATION: DistinctStudentsAttendingClasses