

```
use teengschema;
```

```
CREATE TABLE UserStatuses (  
    Id int auto_increment NOT NULL,  
    StatusName varchar(15) NOT NULL,  
    EntryDate datetime NOT NULL,  
    UpdateDate datetime NOT NULL,  
    PRIMARY KEY (Id)  
);
```

```
CREATE TABLE UserTypes (  
    Id int auto_increment NOT NULL,  
    TypeName varchar(20) NOT NULL,  
    TypeDescription varchar(1000) NOT NULL,  
    EntryDate datetime NOT NULL,  
    UpdateDate datetime NOT NULL,  
    PRIMARY KEY (Id)  
);
```

```
INSERT INTO UserStatuses (StatusName, EntryDate, UpdateDate) VALUES ('Pending', NOW(), NOW());  
INSERT INTO UserStatuses (StatusName, EntryDate, UpdateDate) VALUES ('Active', NOW(), NOW());  
INSERT INTO UserStatuses (StatusName, EntryDate, UpdateDate) VALUES ('Blocked', NOW(), NOW());
```

```
INSERT INTO UserTypes (TypeName, TypeDescription, EntryDate, UpdateDate)  
VALUES ('Standard', 'This is the standard user of the application', NOW(), NOW());
```

```
INSERT INTO UserTypes (TypeName, TypeDescription, EntryDate, UpdateDate)  
VALUES ('Administrative', 'These are the administrators of the application with full power, muah ha ha!', NOW(), NOW());
```

```
CREATE TABLE Users (  
    Id int auto_increment NOT NULL,  
    UserId int NOT NULL,  
    UserStatusId int NOT NULL,  
    Username varchar(50) NOT NULL,  
    HashedPassword varchar(100) NOT NULL,  
    FirstName varchar(50) NOT NULL,  
    LastName varchar(50) NOT NULL,  
    EmailAddress varchar(150) NOT NULL,  
    HomeAddressStreet varchar(150),  
    HomeAddressCity varchar(100),  
    HomeAddressState varchar(25),  
    HomeAddressZip char(9),
```

```

EntryDate datetime NOT NULL,
UpdateDate datetime NOT NULL,
PRIMARY KEY (Id),
FOREIGN KEY (UserId) REFERENCES UserTypes(Id),
FOREIGN KEY (UserStatusId) REFERENCES UserStatuses(Id)
);

```

```

Create table InterestTypes (
    Id int auto_increment not null,
    TypeName nvarchar(25) not null,
    EntryDate datetime not null,
    UpdateDate datetime not null,
    primary key(Id)
);

```

```

Create table PersonalInterests (
    Id int auto_increment not null,
    UserId int not null,
    InterestTypeId Int not null,
    InterestDescription varchar(1000) not null,
    EntryDate datetime not null,
    UpdateDate datetime not null,
    primary key(Id),
    foreign key (UserId) references Users(Id),
    foreign key (InterestTypeId) references InterestTypes(Id)
);

```

```

Create Table ConnectionLevels (
    Id int auto_increment not null,
    TypeName varchar (25) not null,
    EntryDate datetime not null,
    UpdateDate datetime not null,
    Primary Key (Id)
);

```

```

INSERT INTO ConnectionLevels (TypeName, EntryDate, UpdateDate) VALUES ('Public',NOW(), NOW());
INSERT INTO ConnectionLevels (TypeName, EntryDate, UpdateDate) VALUES ('Acquaintances',NOW(), NOW());
INSERT INTO ConnectionLevels (TypeName, EntryDate, UpdateDate) VALUES ('Co-Workers',NOW(), NOW());
INSERT INTO ConnectionLevels (TypeName, EntryDate, UpdateDate) VALUES ('Friends',NOW(), NOW());
INSERT INTO ConnectionLevels (TypeName, EntryDate, UpdateDate) VALUES ('Family',NOW(), NOW());
INSERT INTO ConnectionLevels (TypeName, EntryDate, UpdateDate) VALUES ('Intimates',NOW(), NOW());

```

```

Create Table ConnectionStatuses (
    Id int auto_increment not null,
    StatusType varchar (15) not null,
    EntryDate datetime not null,
    UpdateDate datetime not null,
    primary key (Id)
);

Insert into ConnectionStatuses (StatusType, EntryDate, UpdateDate) values ('Pending', now(), now());
Insert into ConnectionStatuses (StatusType, EntryDate, UpdateDate) values ('Blocked', now(), now());
Insert into ConnectionStatuses (StatusType, EntryDate, UpdateDate) values ('Active', now(), now());
Insert into ConnectionStatuses (StatusType, EntryDate, UpdateDate) values ('Rejected', now(), now());

Create Table Connections (
    Id int auto_increment not null,
    UserId int not null,
    ConnectionUserId int not null,
    ConnectionLevelId int not null,
    ConnectionStatusId int not null,
    EntryDate datetime not null,
    UpdateDate datetime not null,
    Primary key (Id),
    foreign key (UserId) references Users (Id),
    foreign key (ConnectionUserId) references Users (Id),
    foreign key (ConnectionLevelId) references ConnectionLevels (Id),
    foreign key (ConnectionStatusId) references ConnectionStatuses (Id)
);

Create Table EventTypes (
    Id int auto_increment not null,
    EventType varchar (25) not null,
    EventDescription varchar (500) not null,
    EntryDate datetime not null,
    UpdateDate datetime not null,
    primary key (Id)
);

Create Table UserEvents (
    Id int auto_increment not null,
    UserId int not null,
    EventTypeId int not null,
    EventDate datetime not null,

```

```

    IsRecurring bit,
    EntryDate datetime not null,
    UpdateDate datetime not null,
    primary key (Id),
    foreign key (EventTypeId) references EventTypes (Id),
    foreign key (UserId) references Users (Id)
);

CREATE TABLE RegistryStatuses(
    Id int auto_increment NOT NULL
    ,StatusType varchar(15) NOT NULL
    ,EntryDate datetime NOT NULL
    ,UpdateDate datetime NOT NULL
    ,PRIMARY KEY (Id)
);

CREATE TABLE Registries (
    Id int AUTO_INCREMENT NOT NULL,
    UserId int NOT NULL,
    RegistryStatusId int NOT NULL,
    RegistryName varchar(50) NOT NULL,
    EntryDate datetime NOT NULL,
    UpdateDate datetime NOT NULL,
    PRIMARY KEY (Id),
    FOREIGN KEY (UserId) REFERENCES Users(Id),
    FOREIGN KEY (RegistryStatusId) REFERENCES RegistryStatuses(Id)
);

CREATE TABLE RegistryEvents (
    Id int AUTO_INCREMENT NOT NULL,
    RegistryId int NOT NULL,
    EventId int NOT NULL,
    EntryDate datetime NOT NULL,
    UpdateDate datetime NOT NULL,
    PRIMARY KEY (Id),
    FOREIGN KEY (RegistryId) REFERENCES Registries (Id),
    FOREIGN KEY (EventId) REFERENCES UserEvents (Id)
);

CREATE TABLE ItemTypes (
    Id int AUTO_INCREMENT NOT NULL,
    TypeName varchar(50) NOT NULL,

```

```

    TypeDescription varchar(500) NOT NULL,
    EntryDate datetime NOT NULL,
    UpdateDate datetime NOT NULL,
    PRIMARY KEY (Id)
);

INSERT INTO ItemTypes (TypeName, TypeDescription, EntryDate, UpdateDate)
VALUES ('Specific and online', 'Specific product by a specific producer that can be found online. The user should be able to
provide a description and a URL', NOW(), NOW());
INSERT INTO ItemTypes (TypeName, TypeDescription, EntryDate, UpdateDate)
VALUES ('Specific and offline', 'Specific product by a specific producer that is generally found in stores. The user should
be able to provide a description and the name of the store. The physical address would be even better', NOW(), NOW());
INSERT INTO ItemTypes (TypeName, TypeDescription, EntryDate, UpdateDate)
VALUES ('Generic', 'A general class of product that the user can express an interest in without providing specific guidance
as to which one.', NOW(), NOW());
INSERT INTO ItemTypes (TypeName, TypeDescription, EntryDate, UpdateDate)
VALUES ('Curios and collectables', 'Specific products that the user desires but does not know where to get.', NOW(), NOW());
INSERT INTO ItemTypes (TypeName, TypeDescription, EntryDate, UpdateDate)
VALUES ('Services and experiences', 'Intangible gifts that are not physical goods that the user must physically be present
to experience', NOW(), NOW());
INSERT INTO ItemTypes (TypeName, TypeDescription, EntryDate, UpdateDate)
VALUES ('Cash and gift cards', 'Gifts that are directly convertible to money in some form. ', NOW(), NOW());
INSERT INTO ItemTypes (TypeName, TypeDescription, EntryDate, UpdateDate)
VALUES ('Charities', 'The user can indicate that instead of getting a gift directly that they would like their connections
to donate to a particular charity instead', NOW(), NOW());
INSERT INTO ItemTypes (TypeName, TypeDescription, EntryDate, UpdateDate)
VALUES ('Undetermined', 'The item is of an undetermined type', NOW(), NOW());

ALTER TABLE ItemTypes
ADD COLUMN IsIndicatable bit NOT NULL;

UPDATE ItemTypes SET IsIndicatable=1, UpdateDate=Now() WHERE Id IN (1,2,3,4,5,8);

CREATE TABLE ItemStatuses (
    Id int AUTO_INCREMENT NOT NULL,
    StatusName varchar(15) NOT NULL,
    EntryDate datetime NOT NULL,
    UpdateDate datetime NOT NULL,
    PRIMARY KEY (Id)
);

INSERT INTO ItemStatuses (StatusName, EntryDate, UpdateDate) VALUES ('Visible', NOW(), NOW());

```

```
INSERT INTO ItemStatuses (StatusName, EntryDate, UpdateDate) VALUES ('Hidden', NOW(), NOW());
INSERT INTO ItemStatuses (StatusName, EntryDate, UpdateDate) VALUES ('Received', NOW(), NOW());
```

```
CREATE TABLE Items (
    Id int AUTO_INCREMENT NOT NULL,
    ItemTypeId int NOT NULL,
    ItemStatusId int NOT NULL,
    ItemName varchar(50) NOT NULL,
    ItemDescription varchar(1000),
    ItemUrl varchar(1000),
    ItemAddress varchar(1000),
    EntryDate datetime NOT NULL,
    UpdateDate datetime NOT NULL,
    PRIMARY KEY (Id),
    FOREIGN KEY (ItemTypeId) REFERENCES ItemTypes (Id),
    FOREIGN KEY (ItemStatusId) REFERENCES ItemStatuses (Id)
);
```

```
CREATE TABLE ItemConnectionLevels (
    Id int AUTO_INCREMENT NOT NULL,
    ItemId int NOT NULL,
    ConnectionLevelId int NOT NULL,
    EntryDate datetime NOT NULL,
    UpdateDate datetime NOT NULL,
    PRIMARY KEY (Id),
    FOREIGN KEY (ItemId) REFERENCES Items (Id),
    FOREIGN KEY (ConnectionLevelId) REFERENCES ConnectionLevels (Id)
);
```

```
CREATE TABLE ItemIndications (
    Id int AUTO_INCREMENT NOT NULL,
    UserId int NOT NULL,
    ItemId int NOT NULL,
    Message varchar(1000),
    EntryDate datetime NOT NULL,
    UpdateDate datetime NOT NULL,
    PRIMARY KEY (Id),
    FOREIGN KEY (UserId) REFERENCES Users (Id),
    FOREIGN KEY (ItemId) REFERENCES Items (Id)
);
```

```
CREATE TABLE RegistryMessages (  
    Id int AUTO_INCREMENT NOT NULL,  
    RegistryId int NOT NULL,  
    Message varchar(1000) NOT NULL,  
    EntryDate datetime NOT NULL,  
    UpdateDate datetime NOT NULL,  
    PRIMARY KEY (Id),  
    FOREIGN KEY (RegistryId) REFERENCES Registries (Id)  
);
```