```
-- Tim Henderson
-- Table Creation for diplomacy

START TRANSACTION;
--DROP DATABASE IF EXISTS diplomacy;
--CREATE DATABASE diplomacy DEFAULT CHARACTER SET ascii COLLATE ascii_general_ci;
USE diplomacy;
-------------------------------------  Schema  -------------------------------------
--  users (usr_id : varchar(64), name : varchar(256), email : varchar(256),
--         screen_name : varchar(128), pass_hash : varchar(64), salt : varchar(64),
--         last_login : datetime, creation : datetime, status : varchar(500))
--
--  session (session_id : varchar(64), sig_id : varchar(64), msg_sig : varchar(64),
--           usr_id : varchar(64), last_update : datetime)
--
--  message (msg_id : int(11), from_usr : varchar(64), to_usr : varchar(64),
--           time_sent : datetime, subject : varchar(256), msg : varchar(10000)
--           read : tinyint(1))
--
--  map (map_id : int(11), world_name : varchar(128), pic : varchar(64), keep : tinyint(1))
--
--  game (gam_id : int(11), map_id : int(11), pic : varchar(64),
--        gam_season : enum('spring', 'fall'),  gam_year : year(4), turn_start : datetime,
--        turn_length : time, turn_stage : int(11),  ended : tinyint(1))
--
--  game_membership (usr_id : varchar(64), gam_id : int(11), orders_given : tinyint(1))
--
--  turn_stages (trs_id : int(11), name : varchar(64), description : varchar(256),
--               fall : tinyint(1))
--
--  country (cty_id : int(11), usr_id : varchar(64), name : varchar(128), color : varchar(7))
--
--  territory (ter_id : int(11), map_id : int(11), name : varchar(128), abbrev : varchar(4),
--             piece_x : int(11), piece_y : int(11), label_x : int(11), label_y : int(11),
--             ter_type : enum('land', 'sea'), supply : tinyint(1), coastal : tinyint(1))
--
--  adjacent (ter_id : int(11), adj_ter_id : int(11))
--
--  supplier (ter_id : int(11), cty_id : int(11))
--
--  triangle (tri_id : int(11), ter_id : int(11), x1 : int(11), y1 : int(11), x2 : int(11),
--            y2 : int(11), x3 : int(11), y3 : int(11))
--
--  line (ln_id : int(11), x1 : int(11), y1 : int(11), x2 : int(11), y2 : int(11))
--
--  ter_ln_relation (ter_id : int(11), ln_id : int(11))
--
--  piece (pce_id : int(11), cty_id : int(11), ter_id : int(11),
--         pce_type : enum('fleet', 'army'))
--
--  order_type (odt_id : int(11), order_text : varchar(128))
--
--  orders (ord_id : int(11), cty_id : int(11), pce_id : int(11),
--          season : enum('spring', 'fall'), year : year(4), order_type : int(11),
--          destination : int(11), executed : tinyint(1))
--
--  operands (opr_id : int(11), ord_id : int(11), ter_id : int(11))
-------------------------------------  Schema  -------------------------------------

DROP TABLE IF EXISTS users;
CREATE TABLE users
(
    usr_id varchar(64) NOT NULL,
    name varchar(256) NOT NULL,
    email varchar(256) NOT NULL,
    screen_name varchar(128) NOT NULL,
    pass_hash varchar(64) NOT NULL,
    salt varchar(64) NOT NULL,
```

```
    last_login datetime NOT NULL,
    creation datetime NOT NULL,
    status varchar(500),
    CONSTRAINT pk_users PRIMARY KEY (usr_id),
    CONSTRAINT uq_email UNIQUE (email),
    CONSTRAINT uq_screen_name UNIQUE (screen_name)
);

DROP TABLE IF EXISTS sessions;
CREATE TABLE sessions
(
    session_id varchar(64) NOT NULL,
    sig_id varchar(64) NOT NULL,
    msg_sig varchar(64) NOT NULL,
    usr_id varchar(64) NOT NULL,
    last_update datetime NOT NULL,
    CONSTRAINT pk_session PRIMARY KEY (session_id),
    CONSTRAINT fk_usr_id FOREIGN KEY (usr_id)
        REFERENCES users(usr_id) ON DELETE RESTRICT
);

DROP TABLE IF EXISTS message;
CREATE TABLE message
(
    msg_id int(11) AUTO_INCREMENT,
    from_usr varchar(64) NOT NULL,
    to_usr varchar(64) NOT NULL,
    time_sent datetime NOT NULL,
    subject varchar(256) NOT NULL,
    msg varchar(10000) NOT NULL,
    have_read tinyint(1) DEFAULT 0,
    CONSTRAINT pk_message PRIMARY KEY (msg_id),
    CONSTRAINT fk_from_usr FOREIGN KEY (from_usr)
        REFERENCES users(usr_id) ON DELETE RESTRICT,
    CONSTRAINT fk_to_usr FOREIGN KEY (to_usr)
        REFERENCES users(usr_id) ON DELETE RESTRICT
);

DROP TABLE IF EXISTS map;
CREATE TABLE map
(
    map_id int(11) AUTO_INCREMENT,
    world_name varchar(128),
    pic varchar(64) NOT NULL,
    CONSTRAINT pk_map PRIMARY KEY (map_id)
);

DROP TABLE IF EXISTS turn_stages;
CREATE TABLE turn_stages
(
    trs_id int(11) AUTO_INCREMENT,
    name varchar(64),
    description varchar(256),
    fall tinyint(1) DEFAULT 0,
    CONSTRAINT pk_turn_stages PRIMARY KEY (trs_id)
);

DROP TABLE IF EXISTS game;
CREATE TABLE game
(
    gam_id int(11) AUTO_INCREMENT,
    map_id int(11),
    pic varchar(64),
    gam_season enum('spring', 'fall') DEFAULT 'fall',
    gam_year year(4) DEFAULT 1999,
    turn_start datetime NULL,
    turn_length time DEFAULT '24:00:00',
    turn_stage int(11) DEFAULT 0,
    ended tinyint(1) DEFAULT 0,
```

```
    CONSTRAINT pk_game PRIMARY KEY (gam_id),
    CONSTRAINT fk_map_id FOREIGN KEY (map_id)
        REFERENCES map(map_id) ON DELETE RESTRICT,
    CONSTRAINT fk_turn_stage FOREIGN KEY (turn_stage)
        REFERENCES turn_stages(trs_id) ON DELETE RESTRICT
);

DROP TABLE IF EXISTS game_membership;
CREATE TABLE game_membership
(
    usr_id varchar(64) NOT NULL,
    gam_id int(11) NOT NULL,
    orders_given tinyint(1) DEFAULT 0,
    CONSTRAINT pk_game_membership PRIMARY KEY (usr_id, gam_id),
    CONSTRAINT fk_usr_id FOREIGN KEY (usr_id)
        REFERENCES users(usr_id) ON DELETE RESTRICT,
    CONSTRAINT fk_gam_id FOREIGN KEY (gam_id)
        REFERENCES game(gam_id) ON DELETE RESTRICT
);

DROP TABLE IF EXISTS country;
CREATE TABLE country
(
    cty_id int(11) AUTO_INCREMENT,
    usr_id varchar(64),
    name varchar(128),
    color varchar(7) DEFAULT '#ffffff',
    CONSTRAINT pk_country PRIMARY KEY (cty_id),
    CONSTRAINT fk_usr_id FOREIGN KEY (usr_id)
        REFERENCES users(usr_id) ON DELETE RESTRICT
);

DROP TABLE IF EXISTS territory;
CREATE TABLE territory
(
    ter_id int(11) AUTO_INCREMENT,
    map_id int(11) NOT NULL,
    name varchar(128),
    abbrev varchar(4),
    piece_x int(11) NOT NULL,
    piece_y int(11) NOT NULL,
    label_x int(11) NOT NULL,
    label_y int(11) NOT NULL,
    ter_type enum('land', 'sea') NOT NULL,
    supply tinyint(1) NOT NULL,
    coastal tinyint(1) NOT NULL,
    CONSTRAINT pk_territory PRIMARY KEY (ter_id),
    CONSTRAINT fk_map_id FOREIGN KEY (map_id)
        REFERENCES map(map_id) ON DELETE RESTRICT
);

DROP TABLE IF EXISTS adjacent;
CREATE TABLE adjacent
(
    ter_id int(11) NOT NULL,
    adj_ter_id int(11) NOT NULL,
    CONSTRAINT pk_adjacent PRIMARY KEY (ter_id, adj_ter_id),
    CONSTRAINT fk_ter_id FOREIGN KEY (ter_id)
        REFERENCES territory(ter_id) ON DELETE RESTRICT,
    CONSTRAINT fk_adj_ter_id FOREIGN KEY (adj_ter_id)
        REFERENCES territory(ter_id) ON DELETE RESTRICT
);

DROP TABLE IF EXISTS supplier;
CREATE TABLE supplier
(
    ter_id int(11) NOT NULL,
    cty_id int(11) NOT NULL,
    CONSTRAINT pk_supplier PRIMARY KEY (ter_id, cty_id),
```

```
    CONSTRAINT fk_ter_id FOREIGN KEY (ter_id)
        REFERENCES territory(ter_id) ON DELETE RESTRICT,
    CONSTRAINT fk_cty_id FOREIGN KEY (cty_id)
        REFERENCES country(cty_id) ON DELETE RESTRICT
);

DROP TABLE IF EXISTS triangle;
CREATE TABLE triangle
(
    tri_id int(11) AUTO_INCREMENT,
    ter_id int(11) NOT NULL,
    x1 int(11) NOT NULL,
    y1 int(11) NOT NULL,
    x2 int(11) NOT NULL,
    y2 int(11) NOT NULL,
    x3 int(11) NOT NULL,
    y3 int(11) NOT NULL,
    CONSTRAINT pk_triangle PRIMARY KEY (tri_id),
    CONSTRAINT fk_ter_id FOREIGN KEY (ter_id)
        REFERENCES territory(ter_id) ON DELETE RESTRICT
);

DROP TABLE IF EXISTS line;
CREATE TABLE line
(
    ln_id int(11) AUTO_INCREMENT,
    x1 int(11) NOT NULL,
    y1 int(11) NOT NULL,
    x2 int(11) NOT NULL,
    y2 int(11) NOT NULL,
    CONSTRAINT pk_line PRIMARY KEY (ln_id)
);

DROP TABLE IF EXISTS ter_ln_relation;
CREATE TABLE ter_ln_relation
(
    ter_id int(11) NOT NULL,
    ln_id int(11) NOT NULL,
    CONSTRAINT pk_ter_ln_relation PRIMARY KEY (ter_id, ln_id),
    CONSTRAINT fk_ter_id FOREIGN KEY (ter_id)
        REFERENCES territory(ter_id) ON DELETE RESTRICT,
    CONSTRAINT fk_ln_id FOREIGN KEY (ln_id)
        REFERENCES line(ln_id) ON DELETE RESTRICT
);

DROP TABLE IF EXISTS piece;
CREATE TABLE piece
(
    pce_id int(11) AUTO_INCREMENT,
    cty_id int(11) NOT NULL,
    ter_id int(11) NOT NULL,
    pce_type enum('fleet', 'army'),
    CONSTRAINT pk_pce_id PRIMARY KEY (pce_id),
    CONSTRAINT fk_ter_id FOREIGN KEY (ter_id)
        REFERENCES territory(ter_id) ON DELETE RESTRICT,
    CONSTRAINT fk_cty_id FOREIGN KEY (cty_id)
        REFERENCES country(cty_id) ON DELETE RESTRICT
);

DROP TABLE IF EXISTS order_type;
CREATE TABLE order_type
(
    odt_id int(11) AUTO_INCREMENT,
    order_text varchar(128) NOT NULL,
    CONSTRAINT pk_order_type PRIMARY KEY (odt_id)
);

DROP TABLE IF EXISTS orders;
CREATE TABLE orders
```

```
(
    ord_id int(11) AUTO_INCREMENT,
    cty_id int(11) NOT NULL,
    pce_id int(11) NOT NULL,
    gam_season enum('spring', 'fall') NOT NULL,
    gam_year year(4) NOT NULL,
    order_type int(11) NOT NULL,
    destination int(11),
    executed tinyint(1) DEFAULT 0,
    CONSTRAINT pk_orders PRIMARY KEY (ord_id),
    CONSTRAINT uq_orders UNIQUE (cty_id, pce_id, gam_season, gam_year),
    CONSTRAINT fk_cty_id FOREIGN KEY (cty_id)
        REFERENCES country(cty_id) ON DELETE RESTRICT,
    CONSTRAINT fk_pce_id FOREIGN KEY (pce_id)
        REFERENCES piece(pce_id) ON DELETE RESTRICT,
    CONSTRAINT fk_order_type FOREIGN KEY (order_type)
        REFERENCES order_type(odt_id) ON DELETE RESTRICT,
    CONSTRAINT fk_destination FOREIGN KEY (destination)
        REFERENCES territory(ter_id) ON DELETE RESTRICT
);

DROP TABLE IF EXISTS operands;
CREATE TABLE operands
(
    opr_id int(11) AUTO_INCREMENT,
    ord_id int(11) NOT NULL,
    ter_id int(11) NOT NULL,
    CONSTRAINT pk_operands PRIMARY KEY (opr_id),
    CONSTRAINT fk_ord_id FOREIGN KEY (ord_id)
        REFERENCES orders(ord_id) ON DELETE RESTRICT,
    CONSTRAINT fk_ter_id FOREIGN KEY (ter_id)
        REFERENCES territory(ter_id) ON DELETE RESTRICT
);

COMMIT;
```