create table statement

// ON UPDATE CASCADE would allow you to change the primary key value and any tables that have foreign key references to the value will be changed accordingly.

// For "ON DELETE CASCADE", if a parent with an id is deleted, a record in child with parent\_id = parent.id will be automatically deleted.

// RESTRICT means that any attempt to delete and/or update the parent will fail throwing an error. This is the default behaviour in the event that a referential action is not explicitly specified.

CREATE TABLE Task (

task\_type varchar(50) NOT NULL,

task\_description varchar(255) NULL,

task\_id int NOT NULL,

task\_point int NOT NULL,

owned\_by varchar(50) NOT NULL,

finish\_date datetime NULL,

Primary Key (task\_id),

FOREIGN KEY (owned\_by) REFERENCES User(username) ON UPDATE CASCADE ON DELETE CASCADE);

CREATE TABLE Family (

family\_number int NOT NULL,

house\_name varchar(255) NOT NULL,

Primary Key (family\_number, house\_name),

FOREIGN KEY (house\_name) REFERENCES House(house\_name) ON UPDATE CASCADE ON DELETE CASCADE);

CREATE TABLE House (

house\_name varchar(255) NOT NULL,

total\_point int NOT NULL,

Primary Key (house\_name),

);

CREATE TABLE USER (

username varchar(50) NOT NULL UNIQUE,

last\_name varchar(255) NOT NULL,

first\_name varchar(255) NOT NULL,

phone\_num varchar(20),

email varchar(255),

pref\_communication varchar(10),

PRIMARY KEY(username));

CREATE TABLE Mentor (

username varchar(50) NOT NULL UNIQUE,

alias varchar(255),

opt\_in int,

approved boolean,

depth\_focus varchar(255) NOT NULL,

post\_grad\_plan varchar(255) NOT NULL,

post\_grad\_plan\_desc varchar(255),

expec\_graduation varchar(11) NOT NULL,

transfer\_from\_outside\_tech boolean NOT NULL,

institution\_name varchar(255),

transfer\_from\_within\_tech boolean NOT NULL,

prev\_major varchar(100),

international\_student boolean NOT NULL,

first\_gen\_college\_student boolean NOT NULL,

live\_before\_tech varchar(255) NOT NULL,

live\_on\_campus\_fall boolean NOT NULL,

live\_on\_campus\_spring boolean NOT NULL,

undergrad\_research boolean NOT NULL,

undergrad\_research\_desc varchar(255),

home\_country varchar(255) NOT NULL,

personal\_hobby varchar(255),

gender varchar(1),

family\_belongs int NOT NULL,

house\_blongs varchar(255) NOT NULL,

PRIMARY KEY(username),

FOREIGN KEY(username) REFERENCES User(username) ON UPDATE CASCADE ON DELETE RESTRICT,

FOREIGN KEY(family\_belongs) REFERENCES Family(family\_number) ON UPDATE CASCADE ON DELETE CASCADE,

FOREIGN KEY(house\_blongs) REFERENCES House(house\_name) ON UPDATE CASCADE ON DELETE CASCADE);

CREATE TABLE Mentee (

username varchar(50) NOT NULL UNIQUE,

mentor\_user varchar(50),

depth\_focus varchar(255),

post\_grad\_plan varchar(255),

post\_grad\_plan\_desc varchar(255),

transfer\_from\_outside boolean,

institution\_name varchar(255),

transfer\_from\_within boolean,

prev\_major varchar(100),

international\_student boolean,

first\_gen\_college\_student boolean,

family\_belongs int NOT NULL,

house\_blongs varchar(255) NOT NULL,

PRIMARY KEY(username),

FOREIGN KEY(username) REFERENCES User(username) ON UPDATE CASCADE ON DELETE RESTRICT,

FOREIGN KEY(mentor\_user) REFERENCES Mentor(username) ON UPDATE CASCADE ON DELETE CASCADE,

FOREIGN KEY(family\_belongs) REFERENCES Family(family\_number) ON UPDATE CASCADE ON DELETE CASCADE,

FOREIGN KEY(house\_blongs) REFERENCES House(house\_name) ON UPDATE CASCADE ON DELETE CASCADE);

SQL

**1. Get the points of a house**

SELECT total\_point

FROM House

WHERE house\_name = $house\_name;

**2. Breadth Track inside a house (username and email)**

SELECT username, email

FROM USER AS u

JOIN Mentor AS r ON u.username = r.username

JOIN Mentee AS e ON u.username = e.username

WHERE r.house\_belongs = $house\_name AND e.house\_belongs = $house\_belongs;

**3. Breadth Track inside a family**

SELECT username, email

FROM USER AS u

JOIN Mentor AS r ON u.username = r.username

JOIN Mentee AS e ON u.username = e.username

WHERE r.family\_belongs = $family\_number AND e.family\_belongs = $family\_number;

**4. Set house**

UPDATE Mentor

SET house\_belongs = $house\_name

WHERE username = $username;

UPDATE Mentee

SET house\_belongs = $house\_name

WHERE username = $username;

**5. Set family**

UPDATE Mentor

SET family\_belongs = $family\_number

WHERE username = $username;

UPDATE Mentee

SET family\_belongs = $family\_number

WHERE username = $username;

**6. Inset family into a house**

UPDATE Family

SET house\_name = $house\_name

WHERE family\_number = $family\_number;

**7. Submit tasks**

get task point first:

SELECT task\_point

FROM Task

WHERE task\_id = $task\_id;

Do add:

UPDATE House

SET total\_point = $total\_point

WHERE house\_name = $house\_name;