University of British Columbia, Vancouver
Department of Computer Science

2. Summary

Oftentimes, parents come to pick up their children at the childcare center and ask what they have done for the day. The database models what the child does in the childcare, such as what activity they did or what food they had for lunch. It also helps the childcare keep track of the families that it services and the caretakers that it hires.

Changes to the ER Diagram

- Family Member Entity: Added Member ID attribute (key), changed Primary Key from First Name, Last Name to Member ID
- Child Entity: Removed Child ID attribute (key) and changed it to Age (non-key) instead

Explanation: We made these two changes above based on the feedback from Milestone 1. This way, our "'Family Member" entity will not only consist of the primary key attributes. Also, our sub-entity no longer specifies its own primary key.

- Made the relationship between child and activity to be many to many

Explanation: We thought it made more sense for the relationship to be many-to-many compared to one-to-one, which is what it was before. This way, the "Does" relationship contains all the times that the child did some activity.

- Added entity Schedule and attributes Schedule Id (key), Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, Sunday, Start Time, End Time. Then, made the relationship Has to connect Child and Schedule.

Explanation: To increase the functionality and to make up for deleted entities

- Deleted entity childcare Company and its associated attributes, along with the relationship Owns

Explanation: We thought having many childcare Companies each capable of owning multiple childcare branches was too broad for our database, so we are deleting it to focus on just the childcare branches instead.

- Changed the relationship Host between entities activity and caretakers from one-to-one to many-to-one

Explanation: Decided to have caretakers host specific activities rather than random pairings between activity and caretaker

- Added the Date and Time of the Does relationship between child and activity to be part of the many to many relationship table key

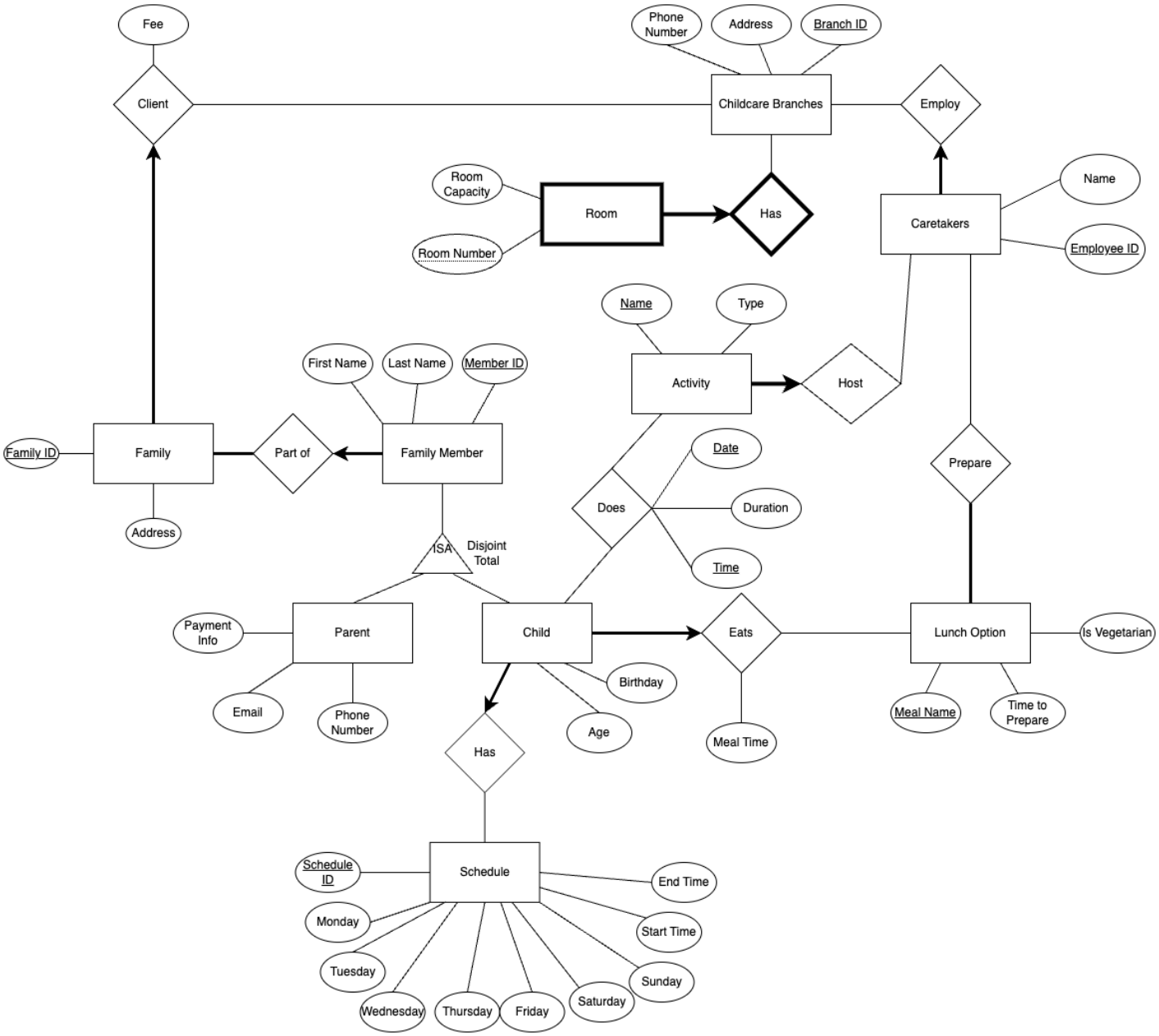
Explanation: This is so that we can identify specific activities that the child completed each day

- Removed Date from the Eats relationship and changed Time attribute to Meal Time

Explanation: This is to keep the relationship consistent with the many to one constraint. The meal_time would represent a fixed time every day that the child would eat instead of a varying time for each day.

University of British Columbia, Vancouver
Department of Computer Science

ER Diagram



University of British Columbia, Vancouver
Department of Computer Science

ER Diagram

https://drive.google.com/file/d/1A6hAn39V-miedwG-R5kN6wAe98gD9nZz/view?usp=sharing

4. Relational Schema

Legend:

- Underline: Primary Key
- **Bold**: Foreign Key
- Highlighted: Naming changed from the ER Diagram for reasons of reducing duplication and better understanding

family_member(first_name: VARCHAR(40), last_name: VARCHAR(40), member_id: VARCHAR(20), **family_id: VARCHAR(20)**)

- member_id cannot be null and is unique
- family_id cannot be null

parent(**member_id: VARCHAR(20)**, payment_info: VARCHAR(20), email: VARCHAR(250), phone_number: VARCHAR(20))

- member_id cannot be null and is unique
- payment_info cannot be null

child(**member_id: VARCHAR(20)**, age: INT, birthday: DATE, **schedule_id: VARCHAR(20)**, **meal_name: VARCHAR(100)**, meal_time: TIME)

- member_id cannot be null and is unique
- age > 0
- birthday follows format: YYYY-MM-DD
- meal_time follows format hh:mm:ss
- schedule_id cannot be null

family(family_id: VARCHAR(20), address: VARCHAR(250), **branch_id: VARCHAR(20)**, Fee: INT)

- family_id cannot be null and is unique
- branch_id cannot be null
- Fee cannot be null
- address cannot be null and is unique
- Candidate Key: {address}

childcare_branch_room(room_capacity: INT, room_number: INT, **branch_id: VARCHAR(20)**)

- (room_number, branch_id) is unique
- room_number cannot be null

- branch_id cannot be null

childcare_branches(phone_number: VARCHAR(20), address: VARCHAR(250), branch_id: VARCHAR(20))
- branch_id cannot be null and is unique
- address cannot be null and is unique
- Candidate Key: {address}

child_does_activity(date: DATE, duration:VARCHAR(250), time: TIME, **child_member_id: VARCHAR(20)**, **activity_name**: VARCHAR(250))
- (child_member_id, activity_name, date, time) is unique
- child_member_id is not null
- activity_name is not null
- date is not null
- time is not null
- date follows format: YYYY-MM-DD
- time follows format hh:mm:ss

activity(name: VARCHAR(250), type: VARCHAR(250), **employee_id: VARCHAR(20)**)
- name cannot be null and is unique
- employee_id cannot be null

caretaker(name: VARCHAR(250), employee_id: VARCHAR(20), **branch_id: VARCHAR(20)**)
- employee_id cannot be null and is unique
- branch_id: cannot be null

caretaker_prepare_lunch(employee_id: VARCHAR(20), meal_name: VARCHAR(100))
- (employee_id, meal_name) is unique
- employee_id is not null
- meal_name is not null

lunch_option(meal_name: VARCHAR(100), time_to_prepare: VARCHAR(250), is_vegetarian: BOOL)
- meal_name cannot be null and is unique

schedule(schedule_id: VARCHAR(20), monday: BOOL, tuesday: BOOL, wednesday: BOOL, thursday: BOOL, friday: BOOL, saturday: BOOL, sunday: BOOL, start_time: TIME, end_time: TIME)
- schedule_id cannot be null and is unique
- end_time > start_time
- start_time and end_time follows format hh:mm:ss

University of British Columbia, Vancouver
Department of Computer Science

5. Functional Dependencies

family_member(first_name: VARCHAR(40), last_name: VARCHAR(40), member_id: VARCHAR(20), **family_id: VARCHAR(20)**)

- member_id → first_name, last_name, family_id

parent(**member_id: VARCHAR(20)**, payment_info: VARCHAR(20), email: VARCHAR(250), phone_number: VARCHAR(20))

- member_id → payment_info, email, phone_number

child(**member_id: VARCHAR(20)**, age: INT, birthday: DATE, **schedule_id: VARCHAR(20)**, **meal_name: VARCHAR(100)**, meal_time: TIME)

- member_id → age, birthday, schedule_id, meal_name, meal_time
- birthday → age

family(family_id: VARCHAR(20), address: VARCHAR(250), **branch_id: VARCHAR(20)**, Fee: INT)

- family_id → address, branch_id, Fee
- address → family_id, branch_id, Fee

childcare_branch_room(room_capacity: INT, room_number: INT, **branch_id: VARCHAR(20)**)

- room_number, branch_id → Capacity

childcare_branches(phone_number: VARCHAR(20), address: VARCHAR(250), branch_id: VARCHAR(20))

- branch_id → address, phone_number
- address → branch_id, phone_number

child_does_activity(date: DATE, duration:VARCHAR(250), time: TIME, **child_member_id: VARCHAR(20)**, **activity_name**: VARCHAR(250))

- date, time, child_member_id, activity_name → duration

activity(name: VARCHAR(250), type: VARCHAR(250), **employee_id: VARCHAR(20)**)

- name → type, employee_id

caretaker(name: VARCHAR(250), employee_id: VARCHAR(20), **branch_id: VARCHAR(20)**)

- employee_id → branch_id, name

caretaker_prepare_lunch(employee_id: VARCHAR(20), meal_name: VARCHAR(100))

- (Primary Key Trivial FD) employee_id, meal_name → employee_id, meal_name

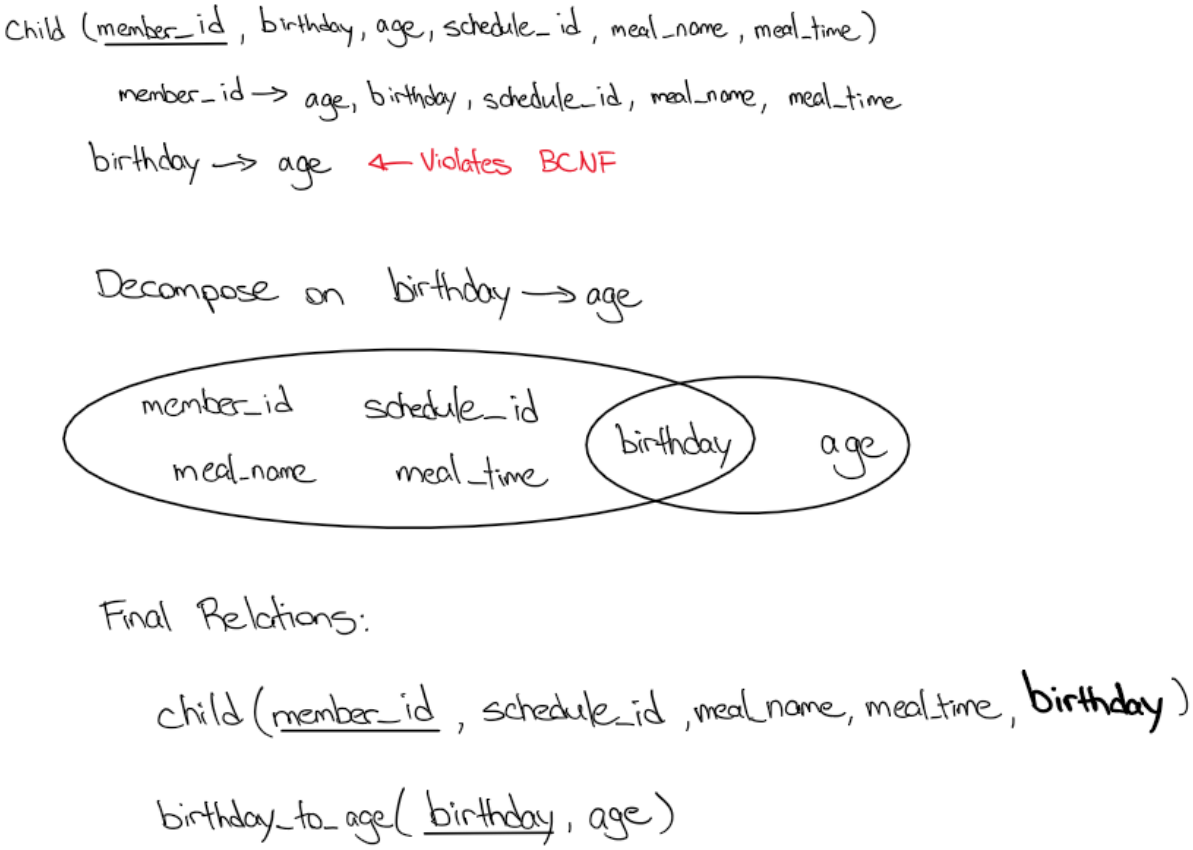
lunch_option(meal_name: VARCHAR(100), time_to_prepare: VARCHAR(250), is_vegetarian: BOOL)

- meal_name → time_to_prepare, is_vegetarian

schedule(schedule_id: VARCHAR(20), monday: BOOL, tuesday: BOOL, wednesday: BOOL, thursday: BOOL, friday: BOOL, saturday: BOOL, sunday: BOOL, start_time: TIME, end_time: TIME)

- schedule_id → monday, tuesday, wednesday, thursday, friday, saturday, sunday, start_time, end_time

6. Normalization



Normalized Relational Schema:

Legend:

- Underline: Primary Key
- **Bold**: Foreign Key

University of British Columbia, Vancouver
Department of Computer Science

family_member(first_name: VARCHAR(40), last_name: VARCHAR(40), member_id: VARCHAR(20), **family_id: VARCHAR(20)**)

- member_id cannot be null and is unique
- family_id cannot be null

parent(**member_id: VARCHAR(20)**, payment_info: VARCHAR(20), email: VARCHAR(250), phone_number: VARCHAR(20))

- member_id cannot be null and is unique
- payment_info cannot be null

child(**member_id: VARCHAR(20)**, **birthday: DATE**, **schedule_id: VARCHAR(20)**, **meal_name**: VARCHAR(20), meal_time: TIME)

- member_id cannot be null and is unique
- age > 0
- birthday follows format: YYYY-MM-DD
- meal_time time follows format hh:mm:ss
- schedule_id cannot be null

birthday_to_age(birthday: DATE, age: INT)

- birthday is not null and is unique
- age is not null
- age > 0
- birthday follows format MM/DD/YYYY, where MM is month, DD is day

family(family_id: VARCHAR(20), address: VARCHAR(250), **branch_id: VARCHAR(20)**, Fee: INT)

- family_id cannot be null and is unique
- branch_id cannot be null
- Fee cannot be null
- address cannot be null and is unique
- Candidate Key: {address}

childcare_branch_room(room_capacity: INT, room_number: INT, **branch_id: VARCHAR(20)**)

- room_number is not null
- branch_id is not null
- (room_number, branch_id) is unique

childcare_branches(phone_number: VARCHAR(20), address: VARCHAR(250), branch_id: VARCHAR(20))

- branch_id cannot be null and is unique
- address cannot be null and is unique

- Candidate Key: {address}

child_does_activity(date: DATE, duration:TIME, time: TIME, **child_member_id: VARCHAR(20)**, **activity_name**: VARCHAR(250))

- (child_member_id, activity_name, date, time) is unique
- child_member_id is not null
- activity_name is not null
- date is not null
- time is not null
- date follows format: YYYY-MM-DD
- time follows format hh:mm:ss

activity(name: VARCHAR(250), type: VARCHAR(250), **employee_id: VARCHAR(20)**)

- name cannot be null and is unique
- employee_id cannot be null

caretaker(name: VARCHAR(40), employee_id: VARCHAR(20), **branch_id: VARCHAR(20)**)

- employee_id cannot be null and is unique
- branch_id: cannot be null

caretaker_prepare_lunch(employee_id: VARCHAR(20), meal_name: VARCHAR(20))

- (employee_id, meal_name) is unique
- employee_id is not null
- meal_name is not null

lunch_option(meal_name: VARCHAR(100), time_to_prepare: TIME, is_vegetarian: BOOL)

- meal_name cannot be null and is unique

schedule(schedule_id: VARCHAR(20), monday: BOOL, tuesday: BOOL, wednesday: BOOL, thursday: BOOL, friday: BOOL, saturday: BOOL, sunday: BOOL, start_time: TIME, end_time: TIME)

- schedule_id cannot be null and is unique
- end_time > start_time
- start_time and end_time follows format hh:mm:ss

7. SQL DDL

```
CREATE TABLE family_member (
    member_id VARCHAR(20) PRIMARY KEY,
    first_name VARCHAR(40),
    last_name VARCHAR(40),
    family_id VARCHAR(20) NOT NULL,
    FOREIGN KEY (family_id) REFERENCES family(family_id)
        ON DELETE CASCADE
        ON UPDATE CASCADE,
    UNIQUE (member_id)
);

CREATE TABLE parent (
    member_id VARCHAR(20) PRIMARY KEY,
    payment_info VARCHAR(20) NOT NULL,
    email VARCHAR(250),
    phone_number VARCHAR(20),
    FOREIGN KEY (member_id) REFERENCES family_member(member_id)
        ON DELETE CASCADE
        ON UPDATE CASCADE
);

CREATE TABLE child (
    member_id VARCHAR(20) PRIMARY KEY,
    birthday DATE,
    schedule_id VARCHAR(20) NOT NULL,
    meal_name VARCHAR(20),
    meal_time TIME,
    FOREIGN KEY (member_id) REFERENCES family_member(member_id)
        ON DELETE CASCADE
        ON UPDATE CASCADE,
    FOREIGN KEY (schedule_id) REFERENCES schedule(schedule_id)
);
```

```
CREATE TABLE birthday_to_age (
    birthday DATE PRIMARY KEY,
    age INT NOT NULL,
    CHECK (age>=0)
);

CREATE TABLE family (
    family_id VARCHAR(20) PRIMARY KEY,
    family_address VARCHAR(250) UNIQUE NOT NULL,
    branch_id VARCHAR(20) NOT NULL,
    fee INT NOT NULL,
    FOREIGN KEY (branch_id) REFERENCES childcare_branches(branch_id)
        ON DELETE CASCADE
        ON UPDATE CASCADE
);

CREATE TABLE childcare_branch_room (
    room_number INT NOT NULL,
    room_capacity INT,
    branch_id VARCHAR(20) NOT NULL,
    PRIMARY KEY (room_number, branch_id),
    FOREIGN KEY (branch_id) REFERENCES childcare_branches(branch_id)
        ON DELETE CASCADE
        ON UPDATE CASCADE,
    UNIQUE (room_number, branch_id)
);

CREATE TABLE childcare_branches (
    phone_number VARCHAR(20),
    childcare_address VARCHAR(250) UNIQUE NOT NULL,
    branch_id VARCHAR(20) PRIMARY KEY
);
```

```
CREATE TABLE child_does_activity (
    date DATE,
    duration TIME,
    time TIME,
    child_member_id VARCHAR(20),
    activity_name VARCHAR (250),
    PRIMARY KEY (date, time, child_member_id, activity_name),
    FOREIGN KEY (child_member_id) REFERENCES child(member_id)
        ON DELETE CASCADE
        ON UPDATE CASCADE,
    FOREIGN KEY (activity_name) REFERENCES activity(name)
);

CREATE TABLE activity (
    name VARCHAR(250) PRIMARY KEY,
    type VARCHAR(250),
    employee_id VARCHAR(20) NOT NULL,
    FOREIGN KEY (employee_id) REFERENCES caretaker(employee_id)
        ON DELETE SET DEFAULT
        ON UPDATE CASCADE
);

CREATE TABLE caretaker (
    name VARCHAR(40),
    employee_id VARCHAR(20) PRIMARY KEY DEFAULT 'N/A',
    branch_id VARCHAR(20) NOT NULL,
    FOREIGN KEY (branch_id) REFERENCES childcare_branches(branch_id)
        ON DELETE CASCADE
        ON UPDATE CASCADE
);
```

```
CREATE TABLE caretaker_prepare_lunch (
    employee_id VARCHAR(20),
    meal_name VARCHAR(20),
    PRIMARY KEY (employee_id, meal_name),
    FOREIGN KEY (employee_id) REFERENCES caretaker(employee_id)
        ON DELETE CASCADE
        ON UPDATE CASCADE,
    FOREIGN KEY (meal_name) REFERENCES lunch_option(meal_name)
        ON DELETE CASCADE
        ON UPDATE CASCADE
);

CREATE TABLE lunch_option (
    meal_name VARCHAR(100) PRIMARY KEY,
    time_to_prepare TIME,
    is_vegetarian BOOL
);

CREATE TABLE schedule (
    schedule_id VARCHAR(20) PRIMARY KEY,
    monday BOOL,
    tuesday BOOL,
    wednesday BOOL,
    thursday BOOL,
    friday BOOL,
    saturday BOOL,
    sunday BOOL,
    start_time TIME,
    end_time TIME,
    CHECK (end_time>start_time)
);
```

8. Populate Tables

```
INSERT INTO family_member (member_id, first_name, last_name, family_id)
VALUES
('1', 'Frank', 'Song', 'F1'),
('2', 'Jake', 'Lake', 'F2'),
('3', 'Michelle', 'James', 'F1'),
('4', 'Edward', 'Wong', 'F3'),
('5', 'Donnie', 'Tello', 'F2');
('6', 'Frannie', 'Song', 'F1'),
('7', 'Jon', 'Lake', 'F2'),
('8', 'Mike', 'James', 'F1'),
('9', 'Emma', 'Wong', 'F3'),
('10', 'Leo', 'Nardo', 'F2');

INSERT INTO parent (member_id, payment_info, email, phone_number) VALUES
('1', '4724123412345678', 'franksong@cpsc.what', '123-456-7890'),
('2', '4580789453164528', 'jakelake@cpsc.what', '987-654-3210'),
('3', '4568123123512355', 'michelle.james@cpsc.what', '111-222-3333'),
('4', '8465213576197538', 'edward_wong@cpsc.what', '444-555-6666'),
('5', '4982159784531238', 'purpleninjaturtle@cpsc.what', '777-888-9999');



INSERT INTO child (member_id, birthday, schedule_id, meal_name, meal_time)
VALUES
('6', '2015-03-12', 'S1', 'Lunch', '12:00:00'),
('7', '2017-08-25', 'S2', 'Dinner', '18:30:00'),
('8', '2014-11-05', 'S1', 'Breakfast', '08:00:00'),
('9', '2019-02-20', 'S3', 'Snack', '15:30:00'),
('10', '2016-05-10', 'S2', 'Lunch', '12:30:00');

INSERT INTO birthday_to_age(birthday, age) VALUES
('2015-03-12', 8)
('2017-08-25', 6)
('2014-11-05', 8)
('2019-02-20', 4)
('2016-05-10', 7)
```

```
INSERT INTO family (family_id, family_address, branch_id, fee) VALUES
('F1', '1234 Road Place', 'B1', 200),
('F2', '4567 Marine Drive', 'B2', 400),
('F3', '7894 Oak St', 'B1', 400),
('F4', '1560 Pine St', 'B3', 550),
('F5', '4895 41st Ave', 'B2', 180);

INSERT INTO childcare_branch_room (room_number, room_capacity, branch_id)
VALUES
(101, 30, 'B1'),
(102, 25, 'B1'),
(201, 20, 'B2'),
(202, 15, 'B2'),
(301, 30, 'B3');

INSERT INTO childcare_branches (branch_id, childcare_address,
phone_number) VALUES
('B1', '9849 West Mall', '123-456-7890'),
('B2', '5461 Westbrook', '987-654-3210'),
('B3', '3812 Oak St', '111-222-3333'),
('B4', '1981 Pine St', '444-555-6666'),
('B5', '3645 16th Ave', '777-888-9999');

INSERT INTO child_does_activity (date, duration, time, child_member_id,
activity_name) VALUES
('2023-10-01', '01:00:00', '10:00:00', '6', 'Drawing'),
('2023-10-02', '00:45:00', '14:30:00', '7', 'Painting'),
('2023-10-03', '00:30:00', '11:00:00', '8', 'Sand Castles'),
('2023-10-04', '02:00:00', '15:00:00', '9', 'Jungle Gym'),
('2023-10-05', '01:30:00', '13:45:00', '10', 'Gymnastics');

INSERT INTO activity (name, type, employee_id) VALUES
('Drawing', 'Art', 'E1'),
('Painting', 'Art', 'E2'),
('Sand Castles', 'Games', 'E3'),
('Jungle Gym', 'Sports', 'E4'),
('Gymnastics', 'Sports', 'E5');
```

```
INSERT INTO caretaker (name, employee_id, branch_id) VALUES
('Sammy', 'E1', 'B1'),
('Daniel', 'E2', 'B2'),
('Leia', 'E3', 'B1'),
('Peter', 'E4', 'B3'),
('Samuel', 'E5', 'B2');

INSERT INTO caretaker_prepare_lunch (employee_id, meal_name) VALUES
('E1', 'Fruit Salad'),
('E2', 'Chicken Strips'),
('E3', 'Ham Sandwich'),
('E4', 'Congee'),
('E5', 'Egg Salad');

INSERT INTO lunch_option (meal_name, time_to_prepare, is_vegetarian)
VALUES
('Fruit Salad', '00:15:00', true),
('Chicken Strips', '00:30:00', false),
('Ham Sandwich', '00:15:00', false),
('Congee', '01:00:00', true),
('Egg Salad', '00:30:00', false);


INSERT INTO schedule (schedule_id, monday, tuesday, wednesday, thursday,
friday, saturday, sunday, start_time, end_time) VALUES
('S1', true, true, false, true, false, false, false, '08:00:00',
'16:00:00'),
('S2', false, false, true, true, false, false, false, '07:30:00',
'15:30:00'),
('S3', true, true, true, false, true, false, false, '09:00:00',
'17:00:00'),
('S4', false, true, false, true, false, true, false, '10:00:00',
'18:00:00'),
('S5', false, false, false, false, true, true, true, '11:30:00',
'19:30:00');
```