University of British Columbia, Vancouver
Department of Computer Science

CPSC 304 Project Cover Page

Milestone #: 2

Date: October 6th, 2023

Group Number: 18

Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
Michele Mai	26575373	x8c3b	michele8231@gmail.com
Sean Dhanda	38290656	c9j3b	sdhanda4862@gmail.com
Ted Lee	25438789	L7E3B	johnj.lee2016@gmail.com

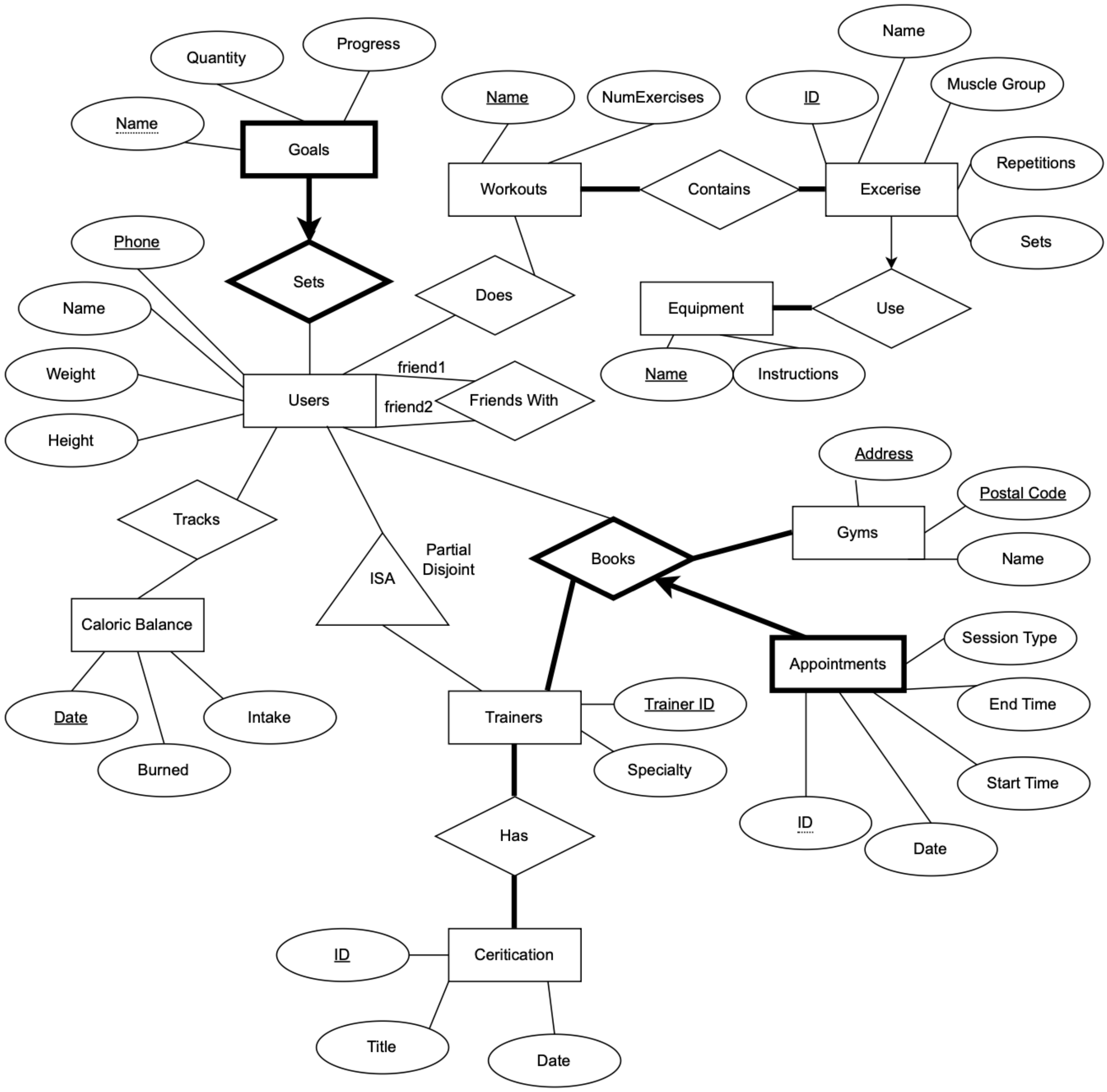
By typing our names and student numbers in the above table, we certify that the work in the attached assignment was performed solely by those whose names and student IDs are included above. (In the case of Project Milestone 0, the main purpose of this page is for you to let us know your e-mail address, and then let us assign you to a TA for your project supervisor.)

In addition, we indicate that we are fully aware of the rules and consequences of plagiarism, as set forth by the Department of Computer Science and the University of British Columbia

Deliverable 2: Project Summary

Our project centers around developing a health and fitness/progress tracking application. In this domain, we are creating a comprehensive fitness app that empowers users to design workouts, monitor goal achievements, and schedule sessions with gym trainers. The database encompasses user profiles, workout plans, exercise details, progress tracking, trainer appointments, and facilitates social interaction, enabling users to share their fitness statistics and achievements with friends.

Deliverable 3: ER Diagram & Note on Changes



Notes on Changes:

- Removed Weight attribute from Exercise, to allow users to have freedom over the weight they use for a given exercise, without having to find a specific exercise ID with the exact weight they want.
- Removed Type attribute from Goal, as having both name and type were redundant for our desired functionality.
- Unary relationship labels with Users table renamed from “Friend and Friend” to “Friend1 and friend2” to mitigate confusion in naming convention.
- Created a 7th entity named “Caloric Balance” that consists of three attributes: “Intake, Burned, and Date.” Intake is the caloric intake on that day, and Burned is the calories burned on that day. Date is the primary key.
- Caloric Balance and User have the 7th relationship “Tracks”.

Deliverables 4 - 8: Schema, FDs, Normalization, SQL DDL, Inserts

Primary Keys are is underlined, **Foreign Keys are bolded**

Table 1: Users

Users(Phone: char(10), Name: varchar, Weight: integer, Height: integer)

FDs:

- Phone → Name
- Phone → Weight
- Phone → Height

Decomposition to BCNF: (Already in BCNF)

```
CREATE TABLE Users
     (Phone CHAR(10) PRIMARY KEY,
     Name VARCHAR,
     Weight INTEGER,
     Height INTEGER);


INSERT
INTO       Users (Phone, Name, Weight, Height)
VALUES     (‘7785734567’, ‘Alice Doe’, ‘134’, 178’)

INSERT
INTO       Users (Phone, Name, Weight, Height)
VALUES     (‘7783334987’, ‘Bob Woo’, ‘156’, 198’)

INSERT
INTO       Users (Phone, Name, Weight, Height)
VALUES     (‘7786879098’, ‘Sally Marsh’, ‘115’, 149’)
```

```
INSERT
INTO       Users (Phone, Name, Weight, Height)
VALUES     ('7784758890', 'Gordon Ramsey', '178', 185')

INSERT
INTO       Users (Phone, Name, Weight, Height)
VALUES     ('7781234098', 'Lizzie Smith', '124', 169')
```

Table 2: FriendsWith

FriendsWith(**friend1_Phone: char(10)**, **friend2_Phone: char(10)**)

FDs:

- friend1_Phone, friend2_Phone → friend1_Name, friend1_Weight, friend1_Height, friend2_Name, friend2_Weight, friend2_Height

Decomposition to BCNF: (Already in BCNF)

```
CREATE TABLE FriendsWith
     (friend1_Phone CHAR(10),
     friend2_Phone CHAR(10),
     PRIMARY KEY(friend1_Phone, friend2_Phone),
     FOREIGN KEY(friend1_Phone)
          REFERENCES Users(Phone)
          ON DELETE CASCADE
     FOREIGN KEY(friend2_Phone)
          REFERENCES Users(Phone)
          ON DELETE CASCADE);


INSERT
INTO       FriendsWith (friend1_Phone, friend2_Phone)
VALUES     ('7781234098', '7784758890')

INSERT
INTO       FriendsWith (friend1_Phone, friend2_Phone)
VALUES     ('7785734567', '7784758890')

INSERT
INTO       FriendsWith (friend1_Phone, friend2_Phone)
VALUES     ('7785734567', '7783334987')

INSERT
INTO       FriendsWith (friend1_Phone, friend2_Phone)
VALUES     ('7785734567', '7783334987')

INSERT
INTO       FriendsWith (friend1_Phone, friend2_Phone)
```

```
VALUES     ('7786879098', '7783334987')
```

Table 3: Sets_Goals

Sets_Goals(**Phone: char(10)**, Goal_Name: varchar, Goal_Quantity: varchar, Goal_Progress: varchar)

FDs:

- Goal_Name → Goal_Quantity
- Goal_Name → Goal_Progress

Decomposition to BCNF: (Already in BCNF)

```
CREATE TABLE Sets_Goals
      (Phone CHAR(10),
      Goal_Name VARCHAR,
      Goal_Quantity VARCHAR,
      Goal_Progress VARCHAR,
      PRIMARY KEY(Phone, Goal_Name),
      FOREIGN KEY(Phone) REFERENCES
           Users(Phone),
           ON DELETE CASCADE);



INSERT
INTO       Sets_Goals (Phone, Goal_Name, Goal_Quantity,
Goal_Progress)
VALUES     ('7785734567', 'Go jogging', '10km', '0km')

INSERT
INTO       Sets_Goals (Phone, Goal_Name, Goal_Quantity,
Goal_Progress)
VALUES     ('7781112222', 'Go to the gym', '10 times', '2
times')

INSERT
INTO       Sets_Goals (Phone, Goal_Name, Goal_Quantity,
Goal_Progress)
VALUES     ('7781113333', 'Bench Press 120lbs', '120lbs',
'110lbs')

INSERT
INTO       Sets_Goals (Phone, Goal_Name, Goal_Quantity,
Goal_Progress)
VALUES     ('7781113333', 'Do 100lbs Barbell Squats', '20
times', '5 times')

INSERT
```

```
INTO      Sets_Goals (Phone, Goal_Name, Goal_Quantity,
Goal_Progress)
VALUES    ('7781113333', 'Go hiking', '5 times', '1 time')
```

<u>Table 4: CaloricBalance</u>

CaloricBalance(<u>Date</u>: char(10), Intake: integer, Burned: integer)

FDs:

- Date → Intake
- Date → Burned

Decomposition to BCNF: (Already in BCNF)

```
CREATE TABLE CaloricBalance
     (Date CHAR(10) PRIMARY KEY,
     Intake INTEGER,
     Burned INTEGER);


INSERT
INTO      CaloricBalance (Date, Intake, Burned)
VALUES    ('01 01 2023', '2500', '1700')

INSERT
INTO      CaloricBalance (Date, Intake, Burned)
VALUES    ('02 01 2023', NULL, NULL)

INSERT
INTO      CaloricBalance (Date, Intake, Burned)
VALUES    ('03 01 2023', '2250', '2400')

INSERT
INTO      CaloricBalance (Date, Intake, Burned)
VALUES    ('14 01 2023', '1700', '1300')

INSERT
INTO      CaloricBalance (Date, Intake, Burned)
VALUES    ('15 01 2023', '2500', '1700')
```

<u>Table 5: Tracks</u>

Tracks(**<u>Phone</u>: char(10)**, **<u>CaloricBalance_Date</u>: char(10)**)

FDs:

- Phone, CaloricBalance_Date → Name, Weight, Height, Intake, Burned

Decomposition to BCNF: (Already in BCNF)

```
CREATE TABLE Tracks
```

```
      (Phone CHAR(10),
      CaloricBalance_Date CHAR(10),
      PRIMARY KEY(Phone, CaloricBalance_Date),
      FOREIGN KEY(Phone)
           REFERENCES Users(Phone)
           ON DELETE CASCADE,
      FOREIGN KEY(CaloricBalance_Date)
           REFERENCES CaloricBalance(Date)
           ON DELETE CASCADE);


INSERT
INTO       Tracks (Phone, CaloricBalance_Date)
VALUES     ('7781234567', '01 01 2023')

INSERT
INTO       Tracks (Phone, CaloricBalance_Date)
VALUES     ('7781234567', '02 01 2023')

INSERT
INTO       Tracks (Phone, CaloricBalance_Date)
VALUES     ('7781112222', '13 07 2020')

INSERT
INTO       Tracks (Phone, CaloricBalance_Date)
VALUES     ('7780003333', '14 07 2030')

INSERT
INTO       Tracks (Phone, CaloricBalance_Date)
VALUES     ('7781114444', '03 02 2024')
```

Table 6: Trainers

Trainers(**Phone: char(10)**, TrainerID: integer, Speciality: varchar)

FDs:

- TrainerID → Specialty

Decomposition to BCNF: (Already in BCNF)

```
CREATE TABLE Trainers
      (Phone CHAR(10),
      TrainerID INTEGER,
      Specialty VARCHAR,
      PRIMARY KEY(Phone, TrainerID),
      FOREIGN KEY(Phone)
           REFERENCES Users(Phone)
           ON DELETE CASCADE);
```

```
INSERT
INTO       Trainers (Phone, TrainerID, Specialty)
VALUES     (‘7782314567’, ‘23458970’, ‘Strength and
Conditioning’)

INSERT
INTO       Trainers (Phone, TrainerID, Specialty)
VALUES     (‘7786188911’, ‘23458971’, ‘HITT’)

INSERT
INTO       Trainers (Phone, TrainerID, Specialty)
VALUES     (‘7786188919’, ‘23458973’, ‘Bodybuilding’)

INSERT
INTO       Trainers (Phone, TrainerID, Specialty)
VALUES     (‘7786188912’, ‘23458972’, ‘Weight Loss’)

INSERT
INTO       Trainers (Phone, TrainerID, Specialty)
VALUES     (‘7786188914’, ‘23458978’, ‘Functional Training’)
```

Table 7: Certification

Certification(ID: integer, Title: varchar, Date: char(10))

FDs:

- ID → Title
- ID → Date

Decomposition to BCNF: (Already in BCNF)

```
CREATE TABLE Certification
     (ID INTEGER PRIMARY KEY,
     Title VARCHAR,
     Date CHAR(10));


INSERT
INTO       Certification (ID, Title, Date)
VALUES     (‘9012’, ‘BCRPA Personal Trainer’, ‘12 08 2014’)

INSERT
INTO       Certification (ID, Title, Date)
VALUES     (‘9013’, ‘ISSA Certification’, ‘15 08 2022’)

INSERT
INTO       Certification (ID, Title, Date)
VALUES     (‘9014’, ‘ISSA Certification’, ‘15 08 2022’)
```

```
INSERT
INTO       Certification (ID, Title, Date)
VALUES     (‘9018’, ‘NSCA-Certified Personal Trainer’, ‘21 04
2015’)

INSERT
INTO       Certification (ID, Title, Date)
VALUES     (‘9010’, ‘ACE Certified Personal Trainer’, ‘17 05
2019’)
```

Table 8: Has

Has(**TrainerID: integer**, **Certification_ID: integer**)

FDs:

- TrainerID, Ceritification_ID → Specialty, Title, Date

Decomposition to BCNF: (Already in BCNF)

```
CREATE TABLE Has
      (TrainerID INTEGER,
      Certification_ID INTEGER,
      PRIMARY KEY(TrainerID, Certification_ID),
      FOREIGN KEY(TrainerID)
            REFERENCES Trainers(TrainerID),
            ON DELETE CASCADE
      FOREIGN KEY(Certification_ID)
            REFERENCES Certification(ID)
            ON DELETE CASCADE);


INSERT
INTO       Has (TrainerID, Certification_ID)
VALUES     (‘23458970’, ‘0918’)

INSERT
INTO       Has (TrainerID, Certification_ID)
VALUES     (‘23458971’, ‘0918’)

INSERT
INTO       Has (TrainerID, Certification_ID)
VALUES     (‘23458973’, ‘0915’)

INSERT
INTO       Has (TrainerID, Certification_ID)
VALUES     (‘23458972’, ‘0918’)

INSERT
```

```
INTO       Has (TrainerID, Certification_ID)
VALUES     (‘23458978’, ‘0913’)
```

Table 9: Workouts

Workouts(Name: varchar, NumExercises: integer)

FDs:

- Name → NumExercises

Decomposition to BCNF: (Already in BCNF)

```
CREATE TABLE Workouts
      (Name VARCHAR PRIMARY KEY,
      NumExercises INTEGER);


INSERT
INTO       Workouts (Name, NumExercises)
VALUES     (‘HIT: Legs and Core’, ‘3’)

INSERT
INTO       Workouts (Name, NumExercises)
VALUES     (‘Intense Abs’, ‘3’)

INSERT
INTO       Workouts (Name, NumExercises)
VALUES     (‘Biceps and Triceps’, ‘3’)

INSERT
INTO       Workouts (Name, NumExercises)
VALUES     (‘Super Upper Body Strength’, ‘3’)

INSERT
INTO       Workouts (Name, NumExercises)
VALUES     (‘Endurance and Core Strength’, ‘3’)
```

Table 10: Does

Does(**Phone: char(10)**, **Workouts_Name: varchar**)

FDs:

- Phone, Workouts_Name → Name, Weight, Height

Decomposition to BCNF: (Already in BCNF)

```
CREATE TABLE Does
      (Phone CHAR(10),
      Workouts_Name VARCHAR,
      PRIMARY KEY(Phone, Workouts_Name),
      FOREIGN KEY(Phone)
           REFERENCES Users(Phone)
           ON DELETE CASCADE
      FOREIGN KEY(Workouts_Name)
           REFERENCES Workouts(Name)
           ON DELETE CASCADE);


INSERT
INTO       Does (Phone, Workouts_Name)
VALUES     ('7785734567', 'Super Upper Body Strength')

INSERT
INTO       Does (Phone, Workouts_Name)
VALUES     ('7783334987', 'Biceps and Triceps')

INSERT
INTO       Does (Phone, Workouts_Name)
VALUES     ('7786879098', 'Intense Abs')

INSERT
INTO       Does (Phone, Workouts_Name)
VALUES     ('7784758890', 'Super Upper Body Strength')

INSERT
INTO       Does (Phone, Workouts_Name)
VALUES     ('7781234098', 'HIT: Legs and Core')
```

Table 11: Exercise

Exercise(ID: integer, Exercise_Name: varchar, MuscleGroup: varchar, Repetitions: integer, Sets: integer, **unique(**Exercise_Name, MuscleGroup**,** Repetitions, Sets**)**)
Candidate Keys:

- {Exercise_Name, MuscleGroup, Repetitions, Sets}

FDs:

- ID → Name
- ID → MuscleGroup
- ID → Repetitions
- ID → Sets
- Name → MuscleGroup
- Name, MuscleGroup, Repetitions, Sets → ID

Decomposition to BCNF:

- Variables:
  A - ID
  B - Name

  C - MuscleGroup
  D - Repetitions
  E - Sets
- R(ABCDE)
- Closures:
  - A+ = ABCDE
  - B+ = BC
  - C+ = C
  - D+ = D
  - E+ = E
  - BCDE+ = ABCDE
- B → C violates BCNF
  - Decompose to R1(B,C), R2(ABDE)
- Look at A → B
  - A is a superkey for R2, so stop
- Final Answer: R1(BC), R2(ABDE)

After Normalization =
R1(Exercise_Name: varchar, MuscleGroup: varchar),
R2(ID: integer, Exercise_Name: varchar, Repetitions: integer, Sets: integer)
Candidate Key for R2: {Exercise_name, Repetitions, Sets)

Before Normalization:

```
CREATE TABLE Exercise
      (ID INTEGER PRIMARY KEY,
     Exercise_Name VARCHAR,
     MuscleGroup VARCHAR,
     Repetitions INTEGER,
     Sets INTEGER,
     UNIQUE (Exercise_Name, MuscleGroup, Repetitions, Sets));
```

After Normalization:

```
CREATE TABLE ExerciseR1
      (Exercise_Name VARCHAR PRIMARY KEY,
     MuscleGroup VARCHAR);

CREATE TABLE ExerciseR2
      (ID INTEGER PRIMARY KEY,
     Exercise_Name VARCHAR,
     Repetitions INTEGER,
     Sets INTEGER
     UNIQUE (Exercise_Name, Repetitions, Sets));
```

Insert Statements Before the Normalization:

```
INSERT
INTO       Exercise (ID, Exercise_Name, MuscleGroup,
Repetitions, Sets)
VALUES     (‘435’, ‘DumbBell Bench Press’, ‘Upper Body’, ‘8’,
‘5’)

INSERT
INTO       Exercise (ID, Exercise_Name, MuscleGroup,
Repetitions, Sets)
VALUES     (‘532’, ‘DumbBell Bent Over Row’, ‘Upper Body’, ‘8’,
‘5’)

INSERT
INTO       Exercise (ID, Exercise_Name, MuscleGroup,
Repetitions, Sets)
VALUES     (‘135’, ‘Drop Squats’, ‘Lower Body’, ‘10’, ‘5’)

INSERT
INTO       Exercise (ID, Exercise_Name, MuscleGroup,
Repetitions, Sets)
VALUES     (‘357’, ‘Mountain Climbers’, ‘Lower Body’, ‘20’,
‘3’)

INSERT
INTO       Exercise (ID, Exercise_Name, MuscleGroup,
Repetitions, Sets)
VALUES     (‘96’, ‘Plank Walk’, ‘Core’, ‘15’, ‘2’)
```

Insert Statements After the Normalization:

```
INSERT
INTO       ExerciseR1 (Exercise_Name, MuscleGroup)
VALUES     (‘DumbBell Bench Press’, ‘Upper Body’)

INSERT
INTO       ExerciseR1 (Exercise_Name, MuscleGroup)
VALUES     (‘DumbBell Bent Over Row’, ‘Upper Body’)

INSERT
INTO       ExerciseR1 (Exercise_Name, MuscleGroup)
VALUES     (‘Drop Squats’, ‘Lower Body’)

INSERT
INTO       ExerciseR1 (Exercise_Name, MuscleGroup)
VALUES     (‘Mountain Climbers’, ‘Lower Body’)

INSERT
INTO       ExerciseR1 (Exercise_Name, MuscleGroup)
```

```
VALUES     (‘Plank Walk’, ‘Core’)
INSERT
INTO       ExerciseR2 (ID, Exercise_Name, Repetitions, Sets)
VALUES     (‘435’, ‘DumbBell Bench Press’, ‘8’, ‘5’)

INSERT
INTO       ExerciseR2 (ID, Exercise_Name, Repetitions, Sets)
VALUES     (‘532’, ‘DumbBell Bent Over Row’, ‘8’, ‘5’)

INSERT
INTO       ExerciseR2 (ID, Exercise_Name, Repetitions, Sets)
VALUES     (‘135’, ‘Drop Squats’, ‘10’, ‘5’)

INSERT
INTO       ExerciseR2 (ID, Exercise_Name, Repetitions, Sets)
VALUES     (‘357’, ‘Mountain Climbers’, ‘20’, ‘3’)

INSERT
INTO       ExerciseR2 (ID, Exercise_Name, Repetitions, Sets)
VALUES     (‘96’, ‘Plank Walk’, ‘15’, ‘2’)
```

Table 12: Contains

Contains(**Workouts_Name: varchar, Exercise_ID: integer**)

FDs:

- Workouts_Name, Exercise_ID → NumExercises, Exercise_Name, MuscleGroup, Repetitions, Sets

Decomposition to BCNF: (Already in BCNF)

```
CREATE TABLE Contains
     (Workouts_NameVARCHAR,
     Exercise_ID INTEGER,
     PRIMARY KEY(Workouts_Name, Exercise_ID),
     FOREIGN KEY(Workouts_Name)
          REFERENCES Workouts(Name),
          ON DELETE CASCADE
     FOREIGN KEY(Exercise_ID)
          REFERENCES Exercise(ID)
          ON DELETE CASCADE);


INSERT
INTO       Contains (Workouts_Name, Exercise_ID)
VALUES     (‘Intense Abs’, ‘4357’)
```

```
INSERT
INTO       Contains (Workouts_Name, Exercise_ID)
VALUES     (‘HIT: Legs and Core’, ‘4396’)

INSERT
INTO       Contains (Workouts_Name, Exercise_ID)
VALUES     (‘Super Upper Body Strength’, ‘4356’)

INSERT
INTO       Contains (Workouts_Name, Exercise_ID)
VALUES     (‘Biceps and Triceps’, ‘4321’)

INSERT
INTO       Contains (Workouts_Name, Exercise_ID)
VALUES     (‘Endurance and Core Strength’, ‘4356’)
```

Table 13: Equipment

Equipment(Name: varchar, Instructions: char(200))

FDs:

- Name → Instructions

Decomposition to BCNF: (Already in BCNF)

```
CREATE TABLE Equipment
      (Name VARCHAR PRIMARY KEY,
      Instructions CHAR(200))


INSERT
INTO       Equipment (Name, Instructions)
VALUES     (‘Yoga Mat’, ‘1. Unroll the yoga mat 2. Position the
mat right side up 3. Place hands and feet on the mat’)

INSERT
INTO       Equipment (Name, Instructions)
VALUES     (‘DumbBells’, ‘1. Hold one dumbbell in each hand 2.
Grip the dumbbell firmly)

INSERT
INTO       Equipment (Name, Instructions)
VALUES     (‘Leg Press Machine’, ‘1. Sit on the machine and
place your feet shoulder width apart on the platform 2. Extend
your legs without locking them 3. Bring your legs back and
repeat’)

INSERT
```

```
INTO       Equipment (Name, Instructions)
VALUES     (‘Lat Pulldown Machine’, ‘1. Sit on the seat and
grip the bar shoulder width apart 2. Pull the bar down to your
chest slowly 3. Release the bar back up and repeat’)

INSERT
INTO       Equipment (Name, Instructions)
VALUES     (‘Pull-Up Bar’, ‘1. Grip the bar with your arms
positions slightly wider than shoulder width apart 2. Pull
yourself up until your chin reaches the bar 3. Lower yourself
down and repeat’)
```

Table 14: Uses

Uses(**Exercise_ID: integer**, **Equipment_Name: char(20) not null**)

FDs:

- Exercise_ID, Equipment_Name → Exercise_Name, MuscleGroup, Repetitions, Sets, Instructions

Decomposition to BCNF: (Already in BCNF)

```
CREATE TABLE Uses
      (Exercise_ID INTEGER PRIMARY KEY,
      Equipment_Name CHAR(20) NOT NULL,
      FOREIGN KEY(Exercise_ID)
           REFERENCES Exercise(ID)
           ON DELETE CASCADE
      FOREIGN KEY(Equipment_Name)
           REFERENCES Equipment(Name)
           ON DELETE CASCADE);


INSERT
INTO       Uses (Exercise_ID, Equipment_Name)
VALUES     (‘4396’, ‘Yoga Mat’)

INSERT
INTO       Uses (Exercise_ID, Equipment_Name)
VALUES     (‘4357’, ‘Yoga Mat’)

INSERT
INTO       Uses (Exercise_ID, Equipment_Name)
VALUES     (‘4356’, ‘DumbBells’)

INSERT
INTO       Uses (Exercise_ID, Equipment_Name)
VALUES     (‘4321’, ‘DumbBell’)
```

```
INSERT
INTO       Uses (Exercise_ID, Equipment_Name)
VALUES     ('4356', 'Yoga Mat')
```

Table 15: Gyms

Gyms(Address: varchar, PostalCode: char(6), Name: varchar)

FDs:

- Address, PostalCode → Name

Decomposition to BCNF: (Already in BCNF)

```
CREATE TABLE Gyms
      (Address VARCHAR,
      PostalCode CHAR(6),
      Name VARCHAR,
      PRIMARY KEY(Address, PostalCode));
```

```
INSERT
INTO       Has (Address, PostalCode, Name)
VALUES     ('2155 Allison Rd, Vancouver, BC', 'V6T1T5', 'Gold's
Gym University MarketPlace')

INSERT
INTO       Has (Address, PostalCode, Name)
VALUES     ('6138 Student Union Blvd, Vancouver, BC', 'V6T1Z1',
'ARC @ UBC Life Building')

INSERT
INTO       Has (Address, PostalCode, Name)
VALUES     ('6000 Student Union Blvd, Vancouver, BC', 'V6T1T5',
'V6T1Z1', 'BirdCoop Fitness Centre')

INSERT
INTO       Has (Address, PostalCode, Name)
VALUES     ('6108 Thunderbird Blvd Unit 1, Vancouver, BC',
'V6T1Z3', 'UBC BodyWorks Fitness Centre')

INSERT
INTO       Has (Address, PostalCode, Name)
VALUES     ('5740 Toronto Rd #205, Vancouver, BC', 'V6T2H7',
'Little Rock Fitness')
```

<u>Table 16: Books_Appointment</u>

Books_Appointment(**<u>ID</u>: integer , <u>Phone</u>: char(10), <u>TrainerID</u>: integer, <u>Address</u>: varchar, <u>PostalCode</u>: char(6),** Date: char(10), StartTime: char(4), EndTime: char(4), SessionType: char(20))

FDs:

- ID → Date
- ID → StartTime
- ID → EndTime
- ID → SessionType

Decomposition to BCNF: (Already in BCNF)

```
CREATE TABLE Books_Appointment
      (Phone CHAR(10),
      TrainerID INTEGER,
      Address VARCHAR,
      PostalCode CHAR(6),
      ID INTEGER,
      Date CHAR(10),
      StartTime CHAR(4),
      EndTime CHAR(4),
      SessionType CHAR(20),
      PRIMARY KEY(ID, Phone, TrainerID, Address, PostalCode),
      FOREIGN KEY(Phone)
            REFERENCES Users(Phone)
            ON DELETE CASCADE
      FOREIGN KEY(TrainerID)
            REFERENCES Trainers(TrainerID)
            ON DELETE CASCADE
      FOREIGN KEY(Address, PostalCode)
            REFERENCES Gyms(Address, PostalCode)
            ON DELETE CASCADE);


INSERT
INTO      Books_Appointment (Phone, TrainerID, Address,
PostalCode, ID, Date, StartTime, EndTime, SessionType)
VALUES    (‘7785734567’, ‘23458978, ‘5740 Toronto Rd #205,
Vancouver, BC’, ‘V6T2H7’, ‘02 10 2023’,  ‘1400’, ‘1500’,
‘Upper Body’)

INSERT
INTO      Books_Appointment (Phone, TrainerID, Address,
PostalCode, ID, Date, StartTime, EndTime, SessionType)
```

```
VALUES     (‘7783334987’, ‘‘23458972’, ‘5740 Toronto Rd #205,
Vancouver, BC’, ‘V6T2H7’, ‘02 10 2023’, ‘1400’, ‘1500’, ‘Lower
Body’)

INSERT
INTO       Books_Appointment (Phone, TrainerID, Address,
PostalCode, ID, Date, StartTime, EndTime, SessionType)
VALUES     (‘7786879098’, ‘23458973’, ‘5740 Toronto Rd #205,
Vancouver, BC’, ‘V6T2H7’, ‘02 10 2023’,  ‘1400’, ‘1500’,
‘Cardio’)

INSERT
INTO       Books_Appointment (Phone, TrainerID, Address,
PostalCode, ID, Date, StartTime, EndTime, SessionType)
VALUES     (‘7784758890’, ‘23458971’, ‘5740 Toronto Rd #205,
Vancouver, BC’, ‘V6T2H7’, ‘02 10 2023’,  ‘1400’, ‘1500’,
‘Conditioning’)

INSERT
INTO       Books_Appointment (Phone, TrainerID, Address,
PostalCode, ID, Date, StartTime, EndTime, SessionType)
VALUES     (‘7781234098’, ‘23458970’, ‘5740 Toronto Rd #205,
Vancouver, BC’, ‘V6T2H7’, ‘02 10 2023’,  ‘1400’, ‘1500’,
‘Pilates’)
```