

CPSC 304 Project Cover Page

Milestone #: 4

Date: 2023.11.30

Group Number: 92

Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
Jialu Jin	24403594	a2f3b	xyxxjinjialu@163.com
Camilla Ren	93534105	d5k5m	camillarr1002@gmail.com
Hao Jiang	58301110	o3f3l	a1181445408@126.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

Final project description

Our project is based on a comprehensive fitness and health management application. The application is a combined platform for fitness enthusiasts and trainers that facilitates monitoring, organizing, and collaborating on different aspects of an individual's health progress. Its basic features include saving member details, logging workout routines, and recording body data. This enables trainers to provide customized advice and plans, as well as help optimize gym resources by efficiently managing room deployment and equipment usage.

There are three dashboards in our Fitness and Wellness Management System: Member Profile, Workout History, AND Course and Trainer Info. All dashboards allow you to reset and print the table. Following are the lists of functionalities that each table has:

Member Profile:

- Insert a new member profile into the table
- Update a member's fitness goal
- Display members who have purchased all group courses
- Calculate the overall average metabolic rate across different age groups

Workout History

- Delete a workout history record
- Find workout history in a specific room
- Show total workouts and average workout duration

Course and Trainer Info

- Display a list of courses along with assigned trainers
- Find the course that has the number of members less than the threshold
- Find the price of a course
- Find the trainer of a course

Project Repository

https://github.students.cs.ubc.ca/CPSC304-2023W-T1/project_a2f3b_d5k5m_o3f3l.git

Webpage:

https://www.students.cs.ubc.ca/~cm1002/cpsc304/project_a2f3b_d5k5m_o3f3l/dashboard.php

SQL script:

We include this file (gym.sql) in our GitHub repository.

Changes of Schema

1. Changed the referenced table of MemberID attribute in the “make” from the “Members” to the “assessed_bodyAnalysisRecord4”.
2. Repeated the first change to the same attribute in the “evaluate”.
3. Added the missing attribute of Price in the ER Diagram of the “Course” entity.
4. Deleted the Price attribute in the “Purchase” relation.
5. Deleted the constraint of “HistoryID is not null” in the relation schema of “Room”.
6. Deleted the constraint of “Date&Time is not null” in the relation schema of “WorkoutHistory” since it is still possible that two members may have been recorded at the same time.
7. Deleted the candidate keys of ExerciseID, MemberID, and RoomID in the relation schema of “WorkoutHistory” since it does not make sense to make them unique in our design.
8. Deleted the candidate key MemberID in the relation schema “DietPlan” with the same reason in 7.
9. Deleted the candidate key RoomID in the relation schema “Equipment” with the same reason in 7.

Schema & Data

After the SQL initialization script has been run, we got the following tables:

WorkoutHistory2(ExerciseID, Duration, CaloriesBurned)

ExerciseID	Duration	CaloriesBurned
101	30	150
102	45	220
103	60	300
104	55	270
105	40	190

Room(RoomID, Name, MaxCapacit)

RoomID	Name	MaxCapacity
1	Room A'	50
2	Room B'	40
3	Room C'	60
4	Room D'	45
5	Room E'	55

Member(MemberID, Name, DateJoined, FitnessGoal)

University of British Columbia, Vancouver

Department of Computer Science

MemberID	Name	DateJoined	FitnessGoal
1	'John Doe'	'2023-01-15'	'Weight Loss'
2	'Jane Smith'	'2023-03-20'	'Muscle Gain'
3	'Alice Johnson'	'2023-05-10'	'Fitness Maintenance'
4	'Bob Brown'	'2023-07-02'	'Cardiovascular Health'
5	'Eve Wilson'	'2023-09-05'	'Strength Training'
11	'Eva son'	'2023-10-05'	'Strength Training'

WorkoutHistory1(HistoryID, DateTime, **ExerciseID**, **Duration**, **RoomID**, **MemberID**)

HistoryID	DateTime	ExerciseID	Duration	RoomID	MemberID
1	'2023-10-20 08:00:00'	101	30	1	1
2	'2023-10-20 09:00:00'	102	45	2	2
3	'2023-10-20 10:00:00'	103	60	3	3
4	'2023-10-20 11:00:00'	104	55	4	4
5	'2023-10-20 12:00:00'	105	40	5	5

Equipment(EquipmentID, AvailabilityStatus, Type, **RoomID**)

EquipmentID	AvailabilityStatus	Type	RoomID
101	'Available'	'Treadmill'	1
102	'In Use'	'Elliptical'	2
103	'Available'	'Dumbbells'	3
104	'In Use'	'Exercise Bike'	4
105	'Available'	'Rowing Machine'	5

Course(CourseID, Start_Date, Price, Duration)

CourseID	Start_Date	Price	Duration
6	'2023-10-20'	50	60
7	'2023-10-21'	40	45
8	'2023-10-22'	60	75
9	'2023-10-23'	55	90
10	'2023-10-24'	70	70

PrivateSession(CourseID)

CourseID
1
2
3
4
5

GroupSession(CourseID, MaxMembers)

University of British Columbia, Vancouver

Department of Computer Science

CourseID	MaxMembers
6	10
7	15
8	12
9	8
10	20

DietPlan2(DietPlanID, Recipes, MemberID)

DietPlanID	Recipes	MemberID
1	'Balanced Diet'	1
2	'Keto Diet'	2
3	'Vegan Diet'	3
4	'Paleo Diet'	4
5	'Low-Carb Diet'	5

DietPlan3(Carbohydrates, Proteins, Fats, Calories)

Carbohydrates	Proteins	Fats	Calories
100	50	30	1200
80	60	40	1400
60	70	50	1500
120	40	35	1300
90	55	45	1350

DietPlan4(Recipes, Carbohydrates, Proteins, Fats)

Recipes	Carbohydrates	Proteins	Fats
'Balanced Diet'	80	60	40
'Keto Diet'	100	50	30
'Vegan Diet'	120	40	35
'Paleo Diet'	90	55	45
'Low-Carb Diet'	60	70	50

Trainer(TrainerID, Name, Expertise, AvailableHoursDaily)

TrainerID	Name	Expertise	AvailableHoursDaily
201	'Trainer 1'	'Strength Training'	2
202	'Trainer 2'	'Yoga'	3
203	'Trainer 3'	'Cardiovascular Health'	2
204	'Trainer 4'	'CrossFit'	1
205	'Trainer 5'	'Pilates'	3

assessed_BodyAnalysisRecord1(Age, Weight, Height, MetabolicRate)

Age	Weight	Height	MetabolicRat
25	70.5	175	1500
30	68.2	170.5	1400
35	80	180	1600
28	65.5	160	1450
40	75	170	1550

University of British Columbia, Vancouver

Department of Computer Science

assessed_BodyAnalysisRecord3(Age, Weight, Height, Gender, BodyFatPercentage)

Age	Weight	Height	Gender	BodyFatPercentage
25	70.5	175	'Male'	18.5
30	68.2	170.5	'Female'	22
35	80	180	'Male'	15.2
28	65.5	160	'Male'	20.1
40	75	170	'Female'	19.8

assessed_BodyAnalysisRecord4(RecordID, Age, Weight, Height, Gender, Assess_Date, MuscleMass, MemberID)

RecordID	Age	Weight	Height	Gender	Assess_Date	MuscleMass	MemberID
1	25	70.5	175	'Male'	'2023-01-15'	65	1
2	30	68.2	170.5	'Female'	'2023-02-20'	60.5	2
3	35	80	180	'Male'	'2023-03-10'	70	3
4	28	65.5	160	'Male'	'2023-04-05'	58	4
5	40	75	170	'Female'	'2023-05-12'	68.5	5

purchase(MemberID, CourseID)

MemberID	CourseID
1	1
2	2
3	3
4	4
5	5
5	7
5	6
5	8
5	9
5	10

make(RecordID, MemberID, DietPlanID)

RecordID	MemberID	DietPlanID
1	1	1
2	2	2
3	3	3
4	4	4
5	5	5

evaluate(RecordID, MemberID, TrainerID)

RecordID	MemberID	TrainerID
1	1	201
2	2	202
3	3	203
4	4	204
5	5	205

teach(TrainerID, CourseID)

University of British Columbia, Vancouver

Department of Computer Science

TrainerID	CourseID
201	1
202	2
203	3
204	4
205	5

take_in(**CourseID**, **RoomID**)

CourseID	DoomID
1	1
2	2
3	3
4	4
5	5

Schema

WorkoutHistory1(HistoryID: integer, DateTime: TIMESTAMP, **Duration**: integer, **ExerciseID**: integer, **MemberID**: integer, **RoomID**: integer)

- PK: HistoryID
- CK: HistoryID
- FK:
 - (**ExerciseID**, **Duration**) REFERENCES WorkoutHistory2(ExerciseID, Duration)
 - RoomID** REFERENCES Room(RoomID)
 - MemberID** REFERENCES Member(MemberID)

WorkoutHistory2(ExerciseID: integer, Duration: integer, CaloriesBurned: integer)

- PK: ExerciseID, Duration
- CK: ExerciseID, Duration

Room(RoomID: integer, Name: char[20], MaxCapacity: integer)

- PK: RoomID
- CK: RoomID

Equipment(EquipmentID: integer, AvailabilityStatus: char[10], Type: char[20], **RoomID**: integer)

- PK: EquipmentID
- CK: EquipmentID
- FK: **RoomID** REFERENCES Room(RoomID)

Course(CourseID: integer, Start_Date: DATE, Price: float, Duration: integer)

- PK: CourseID
- CK: CourseID

PrivateSession(**CourseID**: integer)

University of British Columbia, Vancouver

Department of Computer Science

- PK: **CourseID**
- CK: **CourseID**
- FK: **CourseID** REFERENCES Course(CourseID)

GroupSession(**CourseID**: integer, MaxMembers: integer)

- PK: **CourseID**
- CK: **CourseID**
- FK: **CourseID** REFERENCES Course(CourseID)

Member(**MemberID**: integer, Name: char[30], DateJoined: DATE, FitnessGoal: char[30])

- PK: **MemberID**
- CK: **MemberID**

Trainer(**TrainerID**: integer, Name: char[30], Expertise: char[30], AvailableHoursDaily: integer)

- PK: **TrainerID**
- CK: **TrainerID**

DietPlan2(**DietPlanID**: integer, Recipes: char[100], **MemberID**: integer)

- PK: **DietPlanID**
- CK: **DietPlanID**
- FK: **MemberID** REFERENCES Member(MemberID)

DietPlan3(**Carbohydrates**: integer, **Proteins**: integer, **Fats**: integer, Calories: integer)

- PK: **Carbohydrates, Proteins, Fats**
- CK: **Carbohydrates, Proteins, Fats**

DietPlan4(**Recipes**: char[100], **Carbohydrates**: integer, **Proteins**: integer, **Fats**: integer)

- PK: **Recipes**
- CK: **Recipes**
- FK: (**Carbohydrates, Proteins, Fats**) REFERENCES DietPlan3(Carbohydrates, Proteins, Fats)

assessed_BodyAnalysisRecord1(**Age**: integer, **Weight**: float, **Height**: float, MetabolicRate: float)

- PK: **Weight, Height, Age**
- CK: **Weight, Height, Age**

assessed_BodyAnalysisRecord3(**Age**: integer, **Weight**: float, **Height**: float, **Gender**: char[10],

BodyFatPercentage: float)

- PK: **Weight, Height, Age, Gender**
- CK: **Weight, Height, Age, Gender**
- FK:
 (**Age, Weight, Height**) REFERENCES assessed_BodyAnalysisRecord1(**Weight, Height, Age**)

assessed_BodyAnalysisRecord4(**RecordID**: integer, **Age**: integer, **Weight**: float, **Height**: float, **Gender**: char[10], Assess_Date: DATE, MuscleMass: float, **MemberID**: integer)

University of British Columbia, Vancouver

Department of Computer Science

- PK: RecordID, MemberID
- CK: RecordID, MemberID
- FK:
MemberID REFERENCES Member(MemberID)
(Age, Weight, Height, Gender) REFERENCES assessed_BodyAnalysisRecord3(Age, Weight, Height, Gender)

purchase(**MemberID**: integer, **CourseID**: integer)

- PK: **MemberID**, **CourseID**
- CK: **MemberID**, **CourseID**
- FK: **MemberID** REFERENCES Member(MemberID), **CourseID** REFERENCES Course(CourseID)

make(**RecordID**: integer, **MemberID**: integer, **DietPlanID**: integer)

- PK: **RecordID**, **MemberID**, **DietPlanID**
- CK: **RecordID**, **MemberID**, **DietPlanID**
- FK: **RecordID**, **MemberID** REFERENCES BodyAnalysisRecord4(RecordID, MemberID),
DietPlanID REFERENCES DietPlan2(DietPlanID)

evaluate(**RecordID**: integer, **MemberID**: integer, **TrainerID**: integer)

- PK: **RecordID**, **MemberID**, **TrainerID**
- CK: **RecordID**, **MemberID**, **TrainerID**
- FK: **RecordID**, **MemberID** REFERENCES BodyAnalysisRecord4(RecordID, MemberID),
TrainerID REFERENCES Trainer(TrainerID)

teach(**TrainerID**: integer, **CourseID**: integer)

- PK: **CourseID**, **TrainerID**
- CK: **CourseID**, **TrainerID**
- FK: **TrainerID** REFERENCES Trainer(TrainerID), **CourseID** REFERENCES Course(CourseID)

take_in(**CourseID**: integer, **RoomID**: integer)

- PK: **CourseID**, **RoomID**
- CK: **CourseID**, **RoomID**
- FK: **TrainerID** REFERENCES Trainer(TrainerID), **RoomID** REFERENCES Room(RoomID)

SQL Queries

1. Insert (member.php - line 246)

```
INSERT INTO Member VALUES (:bind1, :bind2, :bind3, :bind4)
```

2. Delete (workoutHistoryTable.php - line 208)

University of British Columbia, Vancouver

Department of Computer Science

```
DELETE FROM WorkoutHistory1 WHERE HistoryID = '" . $historyID . "'
```

3. Update (member.php - line 392)

```
UPDATE Member SET FitnessGoal = '" . $new_fitness_goal . "' WHERE MemberID = '" . $memberID . "'
```

4. Selection (workoutHistoryTable.php - line 255)

```
SELECT * FROM WorkoutHistory1 WHERE RoomID = :roomID
```

5. Projection (courseTrainer.php - line 287)

```
SELECT c.CourseID, c.Price FROM Course c WHERE c.CourseID = $CourseID
```

6. Join (courseTrainer.php - line 216)

```
SELECT t.CourseID, tr.TrainerID, tr.Name  
      FROM Trainer tr, teach t  
     WHERE tr.TrainerID = t.TrainerID
```

7. Aggregation with Group By (workoutHistoryTable.php - line 304)

```
SELECT COUNT(*) AS TotalWorkouts, AVG(Duration) AS AverageDuration  
      FROM WorkoutHistory1  
     WHERE MemberID = $MemberID AND TO_CHAR(DateTime, 'YYYY-MM') = '$Month'
```

8. Aggregation with Having (courseTrainer.php - line 247)

```
SELECT CourseID, COUNT(MemberID) AS NumberOfMembers  
      FROM purchase  
     GROUP BY CourseID  
    HAVING COUNT(MemberID) < $threshold
```

9. Nested Aggregation with Group By (member.php - line 329)

```
SELECT  
      CASE  
        WHEN Age < 30 THEN 'Under 30'  
        WHEN Age BETWEEN 30 AND 40 THEN '30-40'  
        ELSE 'Over 40'  
      END AS AgeGroup,  
      AVG(MetabolicRate) AS AverageMetabolicRate  
    FROM assessed_BodyAnalysisRecord1
```

University of British Columbia, Vancouver

Department of Computer Science

```
GROUP BY
CASE
    WHEN Age < 30 THEN 'Under 30'
    WHEN Age BETWEEN 30 AND 40 THEN '30-40'
    ELSE 'Over 40'
END
```

10. Division (member.php - line 292)

```
SELECT DISTINCT m.MemberID, m.Name
FROM Member m
WHERE NOT EXISTS (
    (SELECT CourseID FROM GroupSession)
    MINUS
    (SELECT p.CourseID
    FROM Purchase p, GroupSession gs
    WHERE p.CourseID = gs.CourseID AND
        m.MemberID = p.MemberID)
)
```

Screenshots of query result

1. Insert

Before:

Retrieved data from Member Table:			
Member ID	Date Joined	Fitness Goal	
1	15-JAN-23	Weight Loss	
2	20-MAR-23	Muscle Gain	
3	10-MAY-23	Fitness Maintenance	
4	02-JUL-23	Cardiovascular Health	
5	05-SEP-23	Strength Training	
11	05-OCT-23	Strength Training	

During:

Insert the new member

(Insert)

Press Insert button to add a member record

MemberID:

Name:

DateJoined:

(DateJoined in the YYYY-MM-DD format)

FitnessGoal:

After:

University of British Columbia, Vancouver

Department of Computer Science

The member has been successfully inserted.

Retrieved data from Member Table:

Member ID	Date Joined	Fitness Goal
1	15-JAN-23	Weight Loss
2	20-MAR-23	Muscle Gain
3	10-MAY-23	Fitness Maintenance
4	02-JUL-23	Cardiovascular Health
5	05-SEP-23	Strength Training
11	05-OCT-23	Strength Training
6	27-DEC-23	Weight Loss

2. Delete

Before:

Retrieved data from Workout History Table 1:

History ID	Date and Time	Exercise ID	Duration	Room ID	Member ID
1	20-OCT-23 08.00.00.000000 AM	101	30	1	1
2	20-OCT-23 09.00.00.000000 AM	102	45	2	2
3	20-OCT-23 10.00.00.000000 AM	103	60	3	3
4	20-OCT-23 11.00.00.000000 AM	104	55	4	4
5	20-OCT-23 12.00.00.000000 PM	105	40	5	5

During:

Delete a Workout History Record:

(Delete)

History ID to Delete:

Delete Record

After:

Delete operation successful.

Retrieved data from Workout History Table 1:

History ID	Date and Time	Exercise ID	Duration	Room ID	Member ID
1	20-OCT-23 08.00.00.000000 AM	101	30	1	1
2	20-OCT-23 09.00.00.000000 AM	102	45	2	2
4	20-OCT-23 11.00.00.000000 AM	104	55	4	4
5	20-OCT-23 12.00.00.000000 PM	105	40	5	5

3. Update

Before:

Retrieved data from Member Table:

Member ID	Date Joined	Fitness Goal
1	15-JAN-23	Weight Loss
2	20-MAR-23	Muscle Gain
3	10-MAY-23	Fitness Maintenance
4	02-JUL-23	Cardiovascular Health
5	05-SEP-23	Strength Training
11	05-OCT-23	Strength Training
6	27-DEC-23	Weight Loss

During:

University of British Columbia, Vancouver

Department of Computer Science

Update the Member's Fitness Goal

(Update)

Member ID:

New Goal:

After:

Update successful. 1 row(s) updated. New Fitness Goal is now:

Retrieved data from Member Table:

Member ID	Date Joined	Fitness Goal
1	15-JAN-23	Weight Loss
2	20-MAR-23	Muscle Gain
3	10-MAY-23	Fitness Maintenance
4	02-JUL-23	Cardiovascular Health
5	05-SEP-23	Strength Training
11	05-OCT-23	Weight Loss
6	27-DEC-23	Weight Loss

4. Selection

Before:

Retrieved data from Workout History Table 1:

History ID	Date and Time	Exercise ID	Duration	Room ID	Member ID
1	20-OCT-23 08.00.00.000000 AM	101	30	1	1
2	20-OCT-23 09.00.00.000000 AM	102	45	2	2
4	20-OCT-23 11.00.00.000000 AM	104	55	4	4
5	20-OCT-23 12.00.00.000000 PM	105	40	5	5

During:

Find Workout History in a Specific Room

(Selection)

Room ID:

After:

Workout History in the specified room:

HistoryID	DateTime	ExerciseID	Duration	RoomID	MemberID
2	20-OCT-23 09.00.00.000000 AM	102	45	2	2

5. Projection

Before:

University of British Columbia, Vancouver

Department of Computer Science

Retrieved data from Course Table:

Course ID	Start Date	Price	Duration
1	20-OCT-23	50	60
2	21-OCT-23	40	45
3	22-OCT-23	60	75
4	23-OCT-23	55	90
5	24-OCT-23	70	70
6	20-OCT-23	50	60
7	21-OCT-23	40	45
8	22-OCT-23	60	75
9	23-OCT-23	55	90
10	24-OCT-23	70	70

During:

Find Price of a Course

(Projection)

Course ID:

Find Price

After:

CourseID: 3, Price: 60

6. Join

Before:

Retrieved data from Trainer Table:

Trainer ID	Name	Expertise	Available Hours Daily
201	Trainer 1	Strength Training	2
202	Trainer 2	Yoga	3
203	Trainer 3	Cardiovascular Health	2
204	Trainer 4	CrossFit	1
205	Trainer 5	Pilates	3

Retrieved data from Course Table:

Course ID	Start Date	Price	Duration
1	20-OCT-23	50	60
2	21-OCT-23	40	45
3	22-OCT-23	60	75
4	23-OCT-23	55	90
5	24-OCT-23	70	70
6	20-OCT-23	50	60
7	21-OCT-23	40	45
8	22-OCT-23	60	75
9	23-OCT-23	55	90
10	24-OCT-23	70	70

During:

University of British Columbia, Vancouver

Department of Computer Science

Display a list of courses along with the assigned trainers

(Join)

Display

After:

Joined the Trainer and teach tables

CourseID	TrainerID	Name
1	201	Trainer 1
2	202	Trainer 2
3	203	Trainer 3
4	204	Trainer 4
5	205	Trainer 5

7. Aggregation with Group By

Before:

Retrieved data from Workout History Table 1:

History ID	Date and Time	Exercise ID	Duration	Room ID	Member ID
2	20-OCT-23 09.00.00.000000 AM	102	45	2	2
4	20-OCT-23 11.00.00.000000 AM	104	55	4	4
5	20-OCT-23 12.00.00.000000 PM	105	40	5	5

During:

Show Total Workouts and Average Workout Duration

(Aggregation with Group By)

Member ID:

Month (YYYY-MM):

Find workout details

After:

Total Workouts: 1, Average Duration: 55 minutes

8. Aggregation with Having

Before:

Retrieved data from Course Table:

Course ID	Start Date	Price	Duration
1	20-OCT-23	50	60
2	21-OCT-23	40	45
3	22-OCT-23	60	75
4	23-OCT-23	55	90
5	24-OCT-23	70	70
6	20-OCT-23	50	60
7	21-OCT-23	40	45
8	22-OCT-23	60	75
9	23-OCT-23	55	90
10	24-OCT-23	70	70

During:

University of British Columbia, Vancouver

Department of Computer Science

Find the course that has number of members less than the threshold

(Aggregation with Having)

Threshold:

After:

Courses with number of members below the threshold:

CourseID	NumberOfMembers
6	1
1	1
7	1
2	1
8	1
4	1
5	1
10	1
3	1
9	1

9. Nested Aggregation with Group By

Before:

Retrieved data from Body Analysis Record:

Age	Weight	Height	Metabolic Rate
25	70.5	175	1500
30	68.2	170.5	1400
35	80	180	1600
28	65.5	160	1450
40	75	170	1550

During:

Calculate Overall Average Metabolic Rate Across Different Age Groups

(Nested Aggregation with Group By)

After:

Overall Average Metabolic Rate Across Different Age Groups:

The overall average metabolic rate across age groups is around: 1495.83

10. Division

Before:

University of British Columbia, Vancouver

Department of Computer Science

Retrieved data from Member Table:

Member ID	Date Joined	Fitness Goal
1	15-JAN-23	Weight Loss
2	20-MAR-23	Muscle Gain
3	10-MAY-23	Fitness Maintenance
4	02-JUL-23	Cardiovascular Health
5	05-SEP-23	Strength Training
11	05-OCT-23	Strength Training

During:

Display members who have purchased all group courses:

(Division)

Display

After:

Retrieved members who have purchased all group courses:

MemberID	Name
5	Eve Wilson