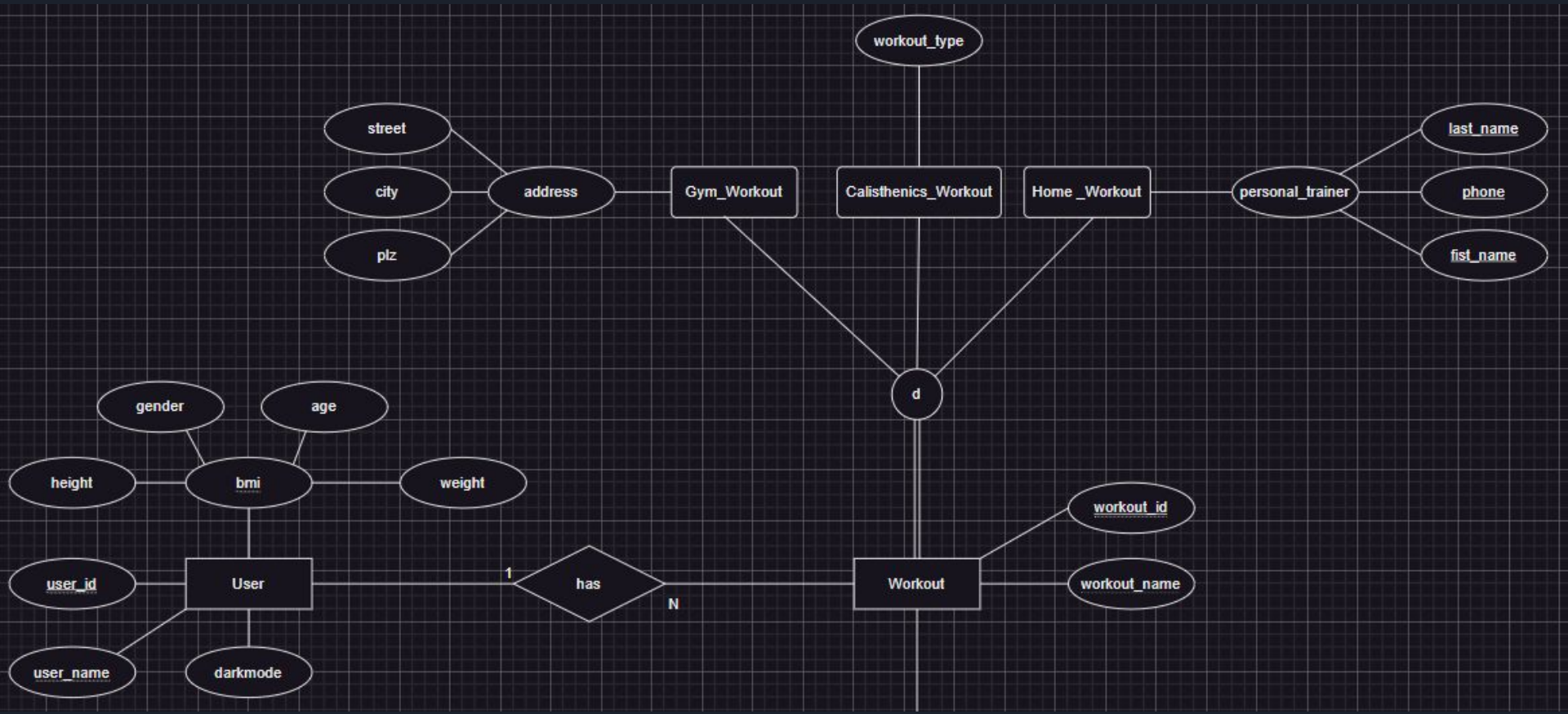
Database for Fitness-App

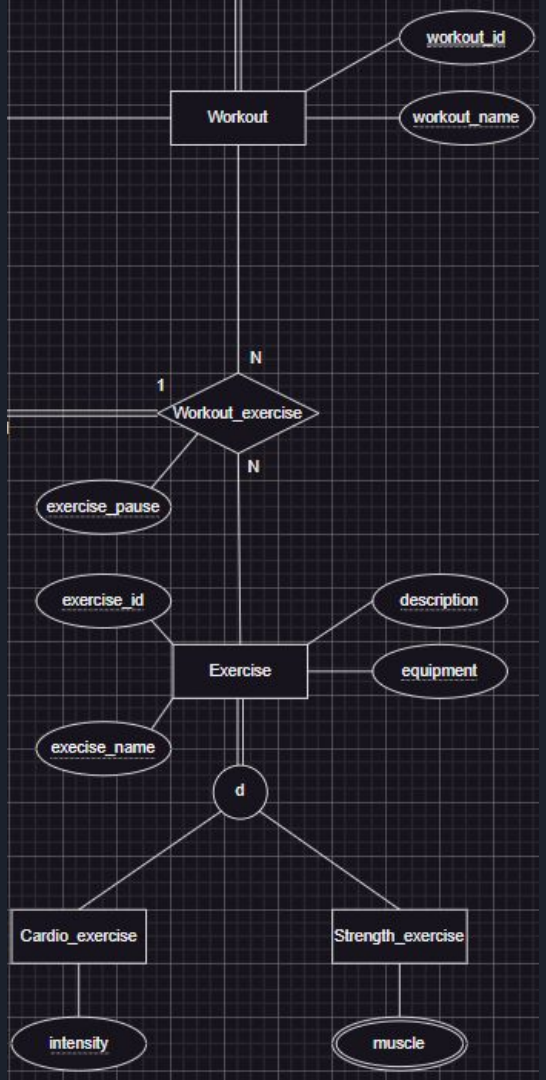
by Nick, Thomas, Willi, Niko, Johannes

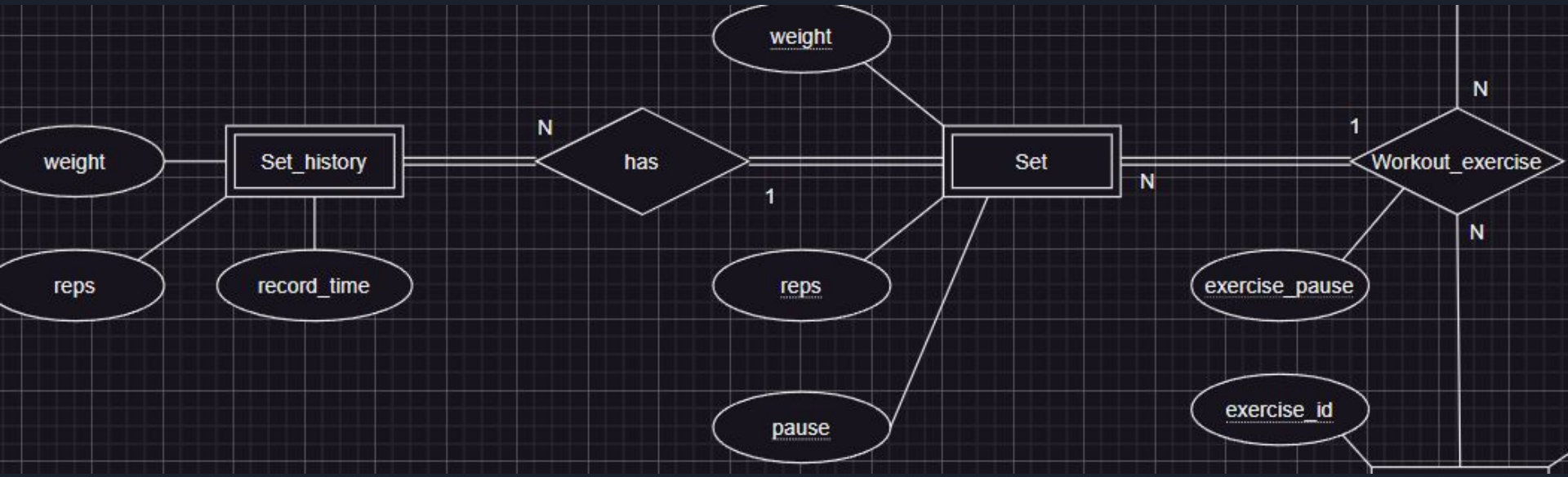
Our Application

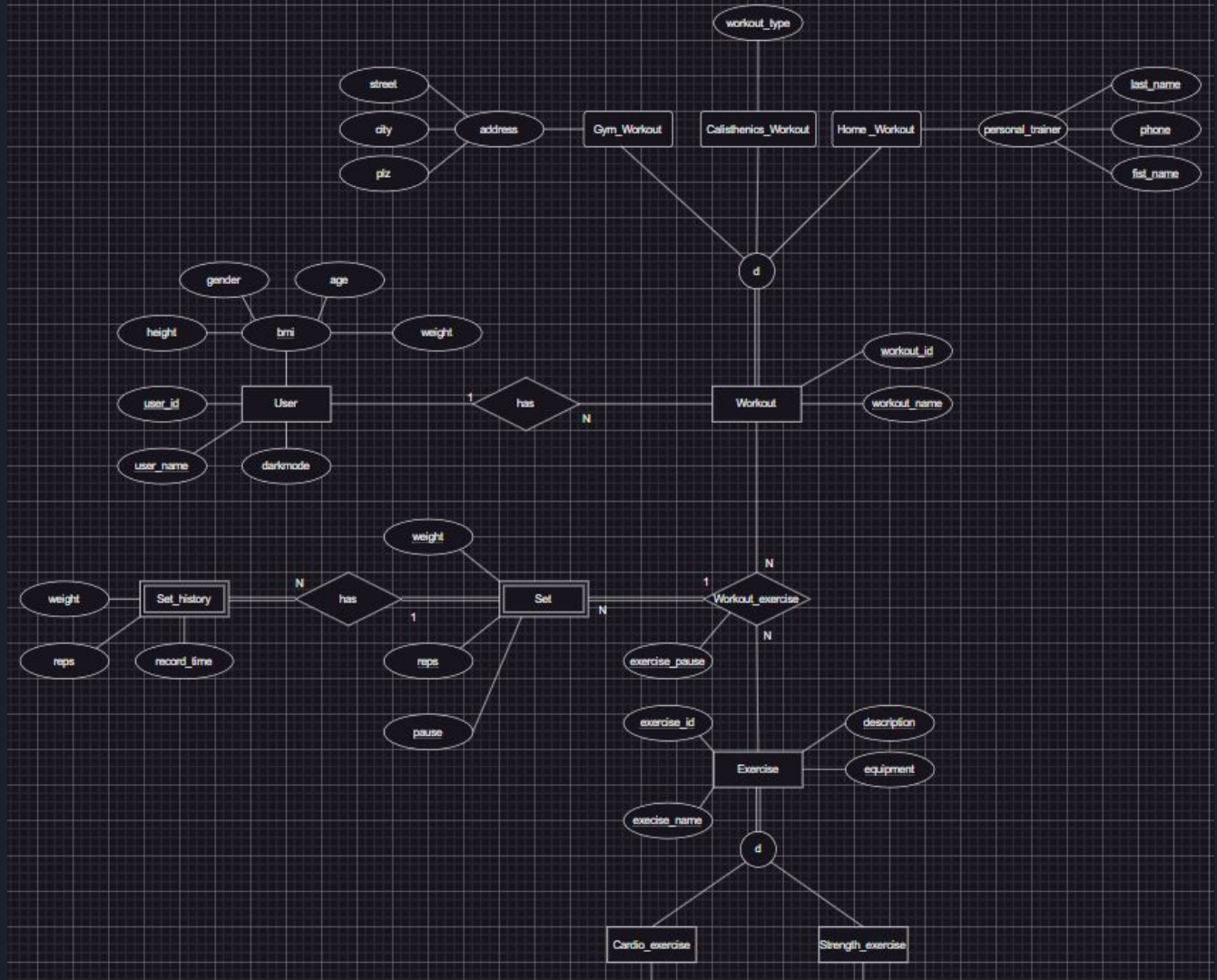
- users can enter their body data to get their bmi
- users create workouts and/or exercises
- each workout consists of multiple exercises
- each exercises can be done for multiple set
- in each set reps, weight and pause time can be specified
- after performing a set the actual amount of reps and weight that has been done can be entered and stored as a history to track progress over time

EER-Model


workout_type
street
city
plz
address
Gym_Workout
Calisthenics_Workout
Home _Workout
personal_trainer
last_name
phone
fist_name
d
gender
age
height
bmi
weight
user_id
User
1
has
N
Workout
workout_id
workout_name
user_name
darkmode


workout_id
Workout
workout_name
N
1
Workout_exercise
N
exercise_pause
exercise_id
description
Exercise
equipment
execise_name
d
Cardio_exercise
Strength_exercise
intensity
muscle

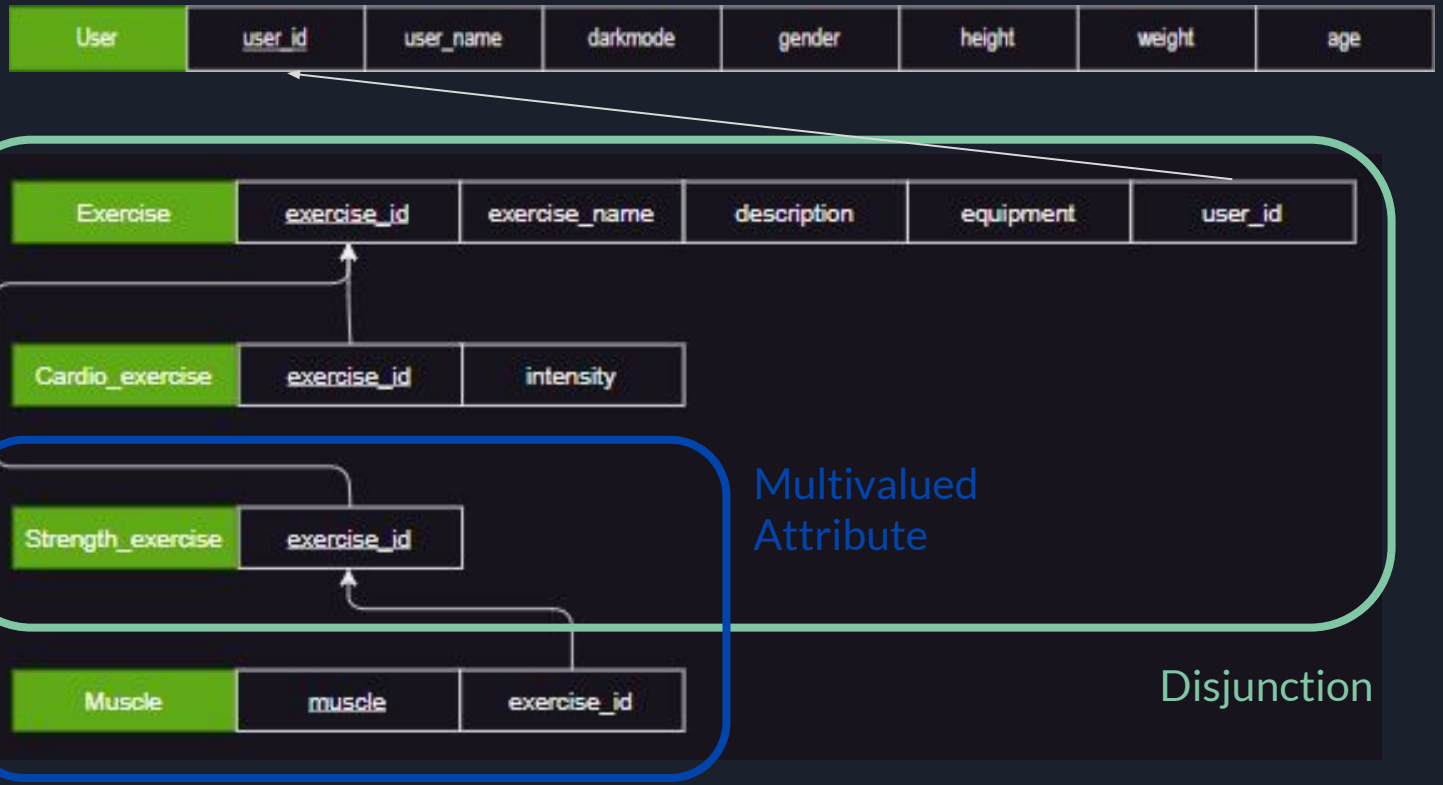

weight
weight
Set_history
N
has
1
Set
N
1
Workout_exercise
N
N
reps
record_time
reps
exercise_pause
pause
exercise_id


workout_type
street
city
plz
address
Gym_Workout
Calisthenics_Workout
Home_Workout
personal_trainer
last_name
phone
fisl_name
d
gender
age
height
bmi
weight
user_id
User
1
has
N
Workout
workout_id
workout_name
user_name
darkmode
weight
N
1
weight
Set_history
N
has
Set
1
Workout_exercise
N
N
reps
record_time
reps
exercise_pause
pause
exercise_id
description
Exercise
equipment
execise_name
d
Cardio_exercise
Strength_exercise

Mapping to Relational Model

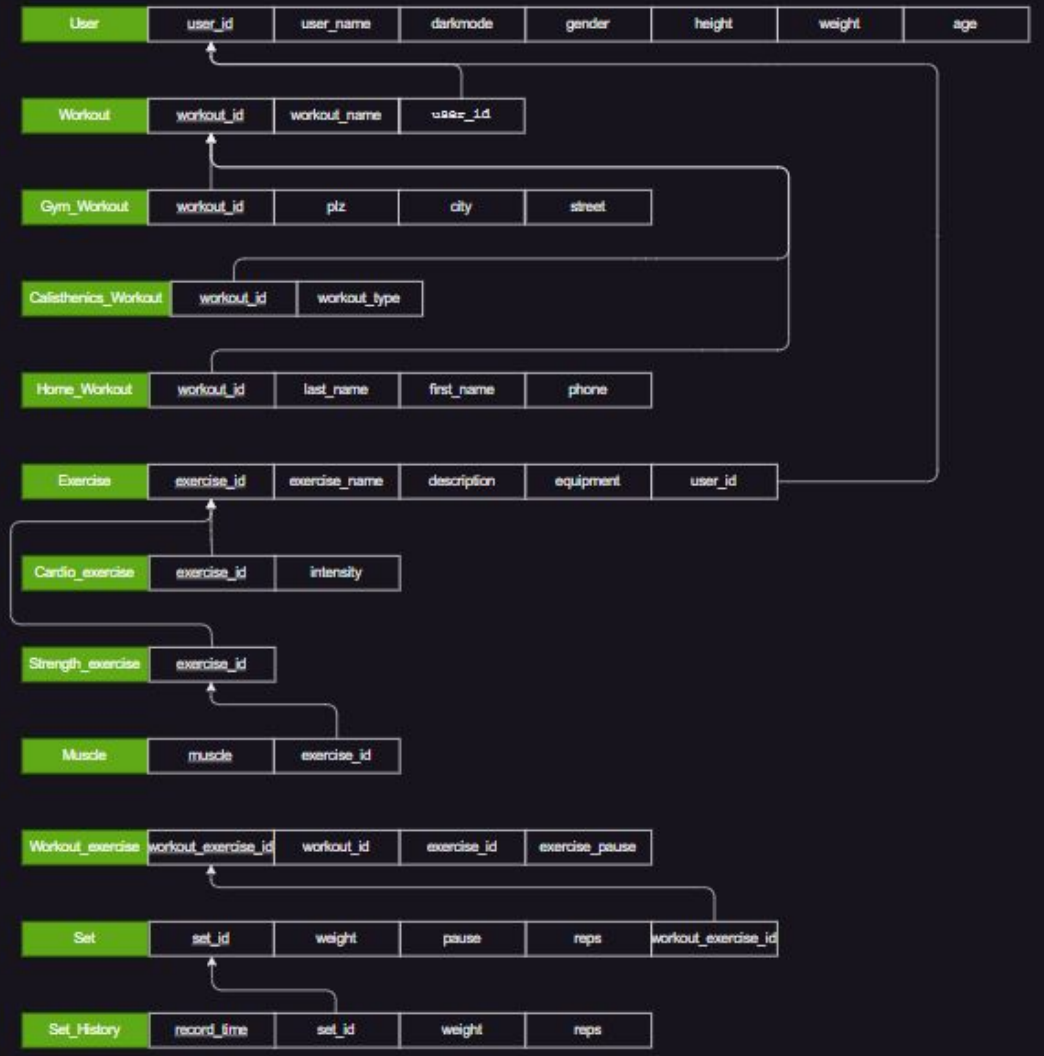
User & Workout

Exercise


User
user_id
user_name
darkmode
gender
height
weight
age
Exercise
exercise_id
exercise_name
description
equipment
user_id
Cardio_exercise
exercise_id
intensity
Strength_exercise
exercise_id
Muscle
muscle
exercise_id
Multivalued Attribute
Disjunction

Workout Exercise & Set

Overview


User
user_id
user_name
darkmode
gender
height
weight
age
Workout
workout_id
workout_name
user_id
Gym_Workout
workout_id
plz
city
street
Calisthenics_Workout
workout_id
workout_type
Home_Workout
workout_id
last_name
first_name
phone
Exercise
exercise_id
exercise_name
description
equipment
user_id
Cardio_exercise
exercise_id
intensity
Strength_exercise
exercise_id
Muscle
muscle
exercise_id
Workout_exercise
workout_exercise_id
workout_id
exercise_id
exercise_pause
Set
set_id
weight
pause
reps
workout_exercise_id
Set_History
record_time
set_id
weight
reps

Normalization

Normalization

- 1NF: Primary key, atomic values, no repeating tuples


User
user_id
user_name
darkmode
gender
height
weight
age
Workout
workout_id
workout_name
user_id
Gym_Workout
workout_id
plz
city
street
Calisthenics_Workout
workout_id
workout_type
Home_Workout
workout_id
last_name
first_name
phone
Exercise
exercise_id
exercise_name
description
equipment
user_id
Cardio_exercise
exercise_id
intensity
Strength_exercise
exercise_id
Muscle
muscle
exercise_id
Workout_exercise
workout_exercise_id
workout_id
exercise_id
exercise_pause
Set
set_id
weight
pause
reps
workout_exercise_id
Set_History
record_time
set_id
weight
reps

Normalization

- 1NF: Primary key, atomic values, no repeating tuples
- 2NF: Full functional dependency to the primary key

User	user_id	user_name	darkmode	gender	height	weight	age

FD: user_id → user_name, darkmode, gender, height, weight, age

Workout	workout_id	workout_name	user_id

FD: workout_id → workout_name, user_id

Gym_Workout	workout_id	plz	city	street

FD: workout_id → plz, city, street

Calisthenics_Workout	workout_id	workout_type

FD: workout_id → workout_type

Home_Workout	workout_id	last_name	first_name	phone

FD: workout_id → last_name, first_name, phone

Exercise	exercise_id	exercise_name	description	equipment	user_id

FD: exercise_id → exercise_name, description, equipment, user_id

Cardio_exercise	exercise_id	intensity

FD: exercise_id → intensity

Strength_exercise	exercise_id

FD: exercise_id → No additional functional dependencies

Muscle	muscle	exercise_id

FD: muscle → exercise_id

Workout_exercise	workout_exercise_id	workout_id	exercise_id	exercise_pause

FD: workout_exercise_id → workout_id, exercise_id, exericse_pause

Set	set_id	weight	pause	reps	workout_exercise_id

FD: set_id → weight, pause, reps, workout_exericse_id

Set_History	record_time	set_id	weight	reps

FD: record_time → set_id, weight, reps

Normalization

- 1NF: Primary key, atomic values, no repeating tuples
- 2NF: Full functional dependency to the primary key
- 3NF: No transitive dependencies

User	user_id	user_name	darkmode	gender	height	weight	age

FD: user_id → user_name, darkmode, gender, height, weight, age

Workout	workout_id	workout_name	user_id

FD: workout_id → workout_name, user_id

Gym_Workout	workout_id	plz	city	street

FD1: workout_id → plz, city, street **FD2: plz→ city**

Calisthenics_Workout	workout_id	workout_type

FD: workout_id → workout_type

Home_Workout	workout_id	last_name	first_name	phone

FD1: workout_id → last_name, first_name, phone **FD2: phone → first_name, last_name**

Exercise	exercise_id	exercise_name	description	equipment	user_id

FD: exercise_id → exercise_name, description, equipment, user_id

Cardio_exercise	exercise_id	intensity

FD: exercise_id → intensity

Strength_exercise	exercise_id

FD: exercise_id → No additional functional dependencies

Muscle	muscle	exercise_id

FD: muscle → exercise_id

Workout_exercise	workout_exercise_id	workout_id	exercise_id	exercise_pause

FD: workout_exercise_id → workout_id, exercise_id, exericse_pause

Set	set_id	weight	pause	reps	workout_exercise_id

FD: set_id → weight, pause, reps, workout_exericse_id

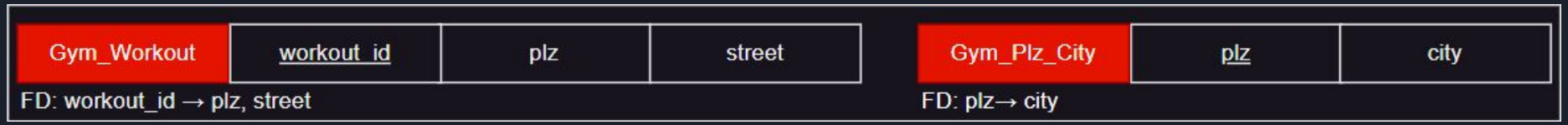
Set_History	record_time	set_id	weight	reps

FD: record_time → set_id, weight, reps

from


Gym_Workout
workout_id
plz
city
street
FD1: workout_id → plz, city, street
FD2: plz→ city

to


Gym_Workout
workout_id
plz
street
FD: workout_id → plz, street
Gym_Plz_City
plz
city
FD: plz→ city

from

Home_Workout	workout_id	last_name	first_name	phone

FD1: workout_id → last_name, first_name, phone

FD2: phone → first_name, last_name

to

Home_Workout	workout_id	phone

FD: workout_id → phone

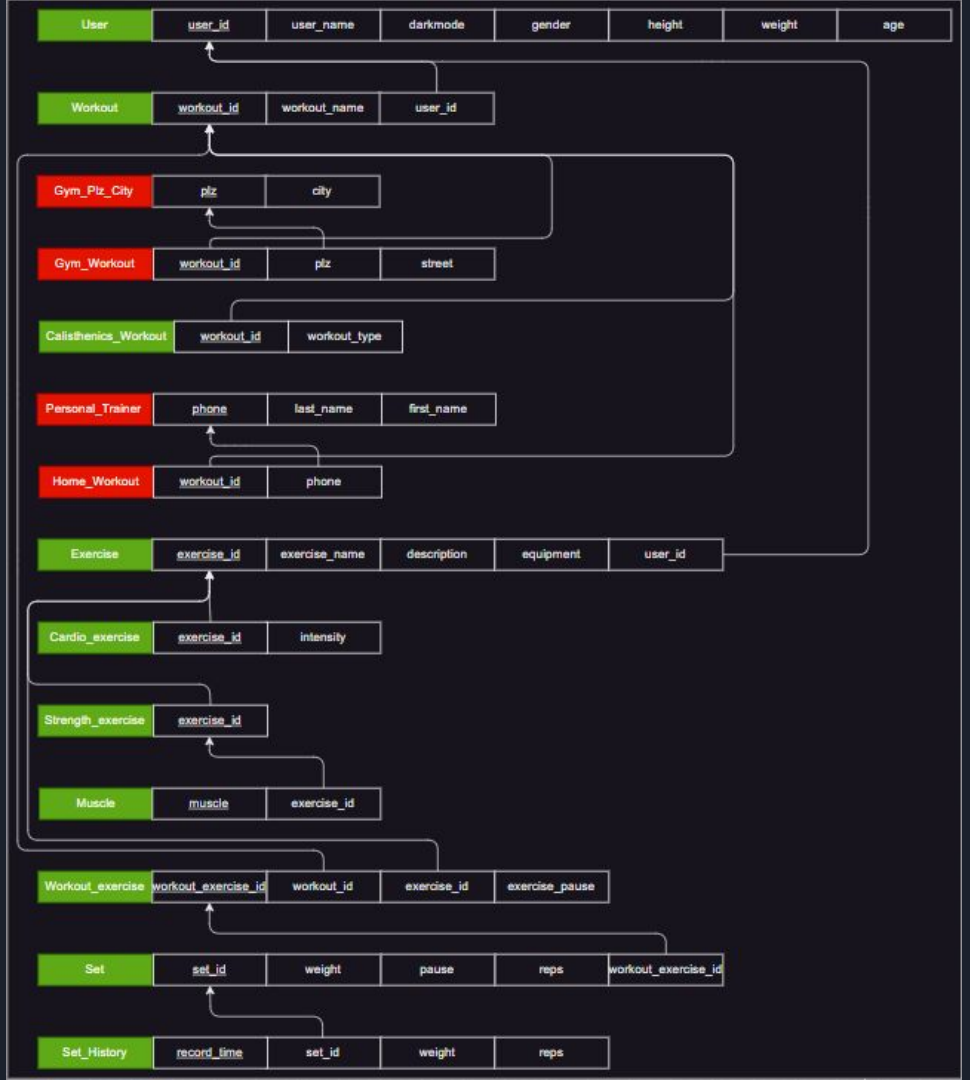
Personal_Trainer	phone	last_name	first_name

FD: phone → first_name, last_name


User
user_id
user_name
darkmode
gender
height
weight
age
FD: user_id → user_name, darkmode, gender, height, weight, age
Workout
workout_id
workout_name
user_id
FD: workout_id → workout_name, user_id
Gym_Workout
workout_id
plz
street
FD: workout_id → plz, street
Gym_Plz_City
plz
city
FD: plz → city
Calisthenics_Workout
workout_id
workout_type
FD: workout_id → workout_type
Home_Workout
workout_id
phone
FD: workout_id → phone
Personal_Trainer
phone
last_name
first_name
FD: phone → first_name, last_name
Exercise
exercise_id
exercise_name
description
equipment
user_id
FD: exercise_id → exercise_name, description, equipment, user_id
Cardio_exercise
exercise_id
intensity
FD: exercise_id → intensity
Strength_exercise
exercise_id
FD: exercise_id → No additional functional dependencies
Muscle
muscle
exercise_id
FD: muscle → exercise_id
Workout_exercise
workout_exercise_id
workout_id
exercise_id
exercise_pause
FD: workout_exercise_id → workout_id, exercise_id, exericse_pause
Set
set_id
weight
pause
reps
workout_exercise_id
FD: set_id → weight, pause, reps, workout_exericse_id
Set_History
record_time
set_id
weight
reps
FD: record_time → set_id, weight, reps

Normalization

- 1NF: Primary key, atomic values, no repeating tuples
- 2NF: Full functional dependency to the primary key
- 3NF: No transitive dependencies
- BCNF: No non-trivial functional dependencies between non-key attributes
- 4NF: No non-trivial multivalued dependencies


User
user_id
user_name
darkmode
gender
height
weight
age
Workout
workout_id
workout_name
user_id
Gym_Plz_City
plz
city
Gym_Workout
workout_id
plz
street
Calisthenics_Workout
workout_id
workout_type
Personal_Trainer
phone
last_name
first_name
Home_Workout
workout_id
phone
Exercise
exercise_id
exercise_name
description
equipment
user_id
Cardio_exercise
exercise_id
intensity
Strength_exercise
exercise_id
Muscle
muscle
exercise_id
Workout_exercise
workout_exercise_id
workout_id
exercise_id
exercise_pause
Set
set_id
weight
pause
reps
workout_exercise_id
Set_History
record_time
set_id
weight
reps

Constraints

Domain Constraints

- Attribute age can accept only Int-Values → age can only be specified as an integer
- Attributes height and weight can accept only float-values → e.g. more precise calculation of BMI

Key Constraints

- Superkey: workout_exercise_id → no two tuples can have the same value

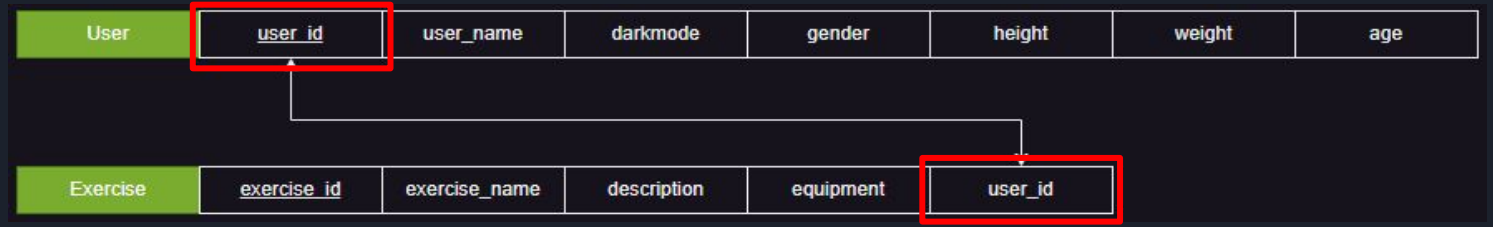
SK = {workout_exercise_id}, {workout_exercise_id, workout_id}, {workout_exercise_id, exercise_id},{workout_exercise_id, exercise_pause}, {workout_exercise_id, workout_id, exercise_id, exercise_pause}

- Every relation has at least one superkey
- The last superkey is a set of all attributes

Entity Integrity Constraints

- Primary key: workout_exercise_id → states that no primary key value can be NULL
- Values of primary key are used to identify the individual tuples

Referential Integrity Constraints



- Foreign key: user_id
  - Exercise → attribute (foreign key)
  - User → primary key
- Foreign key must satisfy the following conditions:
  - Same domain
  - $t_1[FK] = t_2[PK]$ → tuple with FK has the same value like tuple with PK
- Exercise → referencing relation
- User → referenced relation
- This referencing between this two relation disallows deleting of entries with user_id in User relation

Live Demo

Any questions?

Thank you for your attention!