Innlevering 2 - Gruppe 40

TDT4145 - Datamodellering og databasesystemer

Johannes Tomren Røsvik, Dennis Jianbin Liang, Pål Fossnes, Fredrik Jenssen

18.03.18

https://github.com/rosvik/ntnu-tdt4145-database-project

Installation

How to add test-data to database

1. Open MySQL Workbench
2. Find the file named "workoutdiary.sql" locally on your computer
3. Open the file in MySQL Workbench
4. Execute the file
5. Make sure that the workoutdiary database has appeared

Update database credentials

1. Open the file named config.conf
2. Replace 'username' on line 1 and 'password' on line 2 with the credentials to your database
3. If your database's location isn't local, replace the url with an appropriate url

Running

MacOS or Linux:

1. Open Terminal.app
2. Navigate to the project folder `cd location/of/workoutdiary/project`
3. Run `./workout-diary [data]` where [data] is replaced by one of the examples below.
4. Run `./workout-diary help` to get a list of possible commands

Windows

1. Open cmd.exe
2. Navigate to the project folder `cd location/of/workoutdiary/project`
3. Run `workout-diary.bat [data]` where [data] is replaced by one of the examples below.
4. Run `workout-diary.bat help` to get a list of possible commands

Examples

Add any of these examples to the commands explained above. `./workout-diary [example]` on Mac or Linux and `workout-diary.bat [example]` on Windows.

List

List the 5 latest workouts

```
list workout 5
```

List all exercises from 2012 to 2020

```
list exercise 2012-01-01 2020-01-01
```

List all exercises in the Bulking group

```
list group "Bulking"
```

List exercises that have an equipment

```
list equipment 2012-04-25 2018-04-27
```

Add

Add a workout with in April 2018 at 11:45AM with shape 8 and description "Fin tur i skogen"

```
add workout 2018-04-25 11:45:23 8 "Fin tur i skogen"
```

Add an exercise named Fjelltur with description "Gå en tur i fjellet"

```
add exercise "Fjelltur" "Gå en tur i fjellet"
```

Add an equipment named Spaserstokk with the description "Stav som gir stabilitet"

```
add equipment "Spaserstokk" "Stav som gir stabilitet"
```

Make a new group with name "Rolig beilmuskulaturvedlikehold"

```
add group "Rolig beilmuskulaturvedlikehold"
```

Connect

Add exercise with id 1 to workout with id 3 with a duration of 23 minutes and 9 performance

```
connect exercise 3 1 23 9
```

Add equipment with id 1 to exercise id 1 that is 50 kilos and has 10 sets

```
connect equipment 1 1 50 10
```

Add equipment with id 3 to exercise with id 2

```
connect group 2 3
```

Overview of classes:

Our classes are: - ConnectionConfiguration - ConnectionCredentials - MainController - Add - Connect - List - ConnectionApp

MainController

Handles command line input, finds the appropriate function for given parameters, passes information on to the other classes, and displays help information to the user.

ConnectionCredentials

Reads the contents of the config.conf file, and finds the password, username and url for the database.

ConnectionConfiguration

Establishes a connection to a database with the given credentials.

Add

Handles all cases where information needs to be added to the database. This class satisfies requirement number 1, and parts of 4. The class is able to add workouts, exercises, equipment and groups.

Connect

Handles all cases where there needs to be created a relation between two entities. This includes - Adding an exercise to a workout - Adding an exercise to a exercise group - Adding equipment to an exercise

List

Handles all cases where there needs to be displayed a list to the user. This class satisfies the requirement 2,3,5 and the remaining parts of 4. The class is able to display: - The n latest workouts - Workouts with a given exercise within a time period - All exercises that are part of a group - All exercises that uses an equipment

ConnectionApp

Used for testing the application during development

Use cases:

1. Registrere apparater, øvelser og treningsøkter med tilhørende data.
   - Løses ved å kjøre addEquipment(), addExercise(), addWorkout(), som legger inn en ny rad i databasen, med den informasjonen fremstilt av brukeren
2. Få opp informasjon om et antall n sist gjennomførte treningsøkter med notater, der n spesifiseres av brukeren.
   - Løses ved å kjøre listWorkouts(), som gjennomfører spørringer som filtrerer ut ønsket data, og viser denne til brukere.
3. For hver enkelt øvelse skal det være mulig å se en resultatlogg i et gitt tidsintervall spesifisert av brukeren.
   - Løses ved å kjøre listExercises(), som gjennomfører spørringer som filtrerer ut ønsket data gitt ved tidsintervall spesifisert av brukeren, og viser dette.
4. Lage øvelsegrupper og finne øvelser som er i samme gruppe.
   - Løses ved å kjøre addExerciseGroup() og listExerciseInGroup(), som lager en ny excercise gruppe og viser alle exercises som er klassifisert i en gruppe
5. Et valgfritt use case som dere selv bestemmer.
   - Løses ved å kjøre listExercisesWithEquipment(). Denne metoden gjennomfører spørring som viser hvilke exercise som bruker hvilke equipment, i et gitt tidsintervall.