Class Name: Exercise

Attributes:

* name (string)
* description (string)
* muscle\_group (string)
* equipment (string)
* difficulty\_level (string)

Methods:

* getName(): returns the name of the exercise as a string
* getDescription(): returns the description of the exercise as a string
* getMuscleGroup(): returns the muscle group targeted by the exercise as a string
* getEquipment(): returns the equipment needed for the exercise as a string
* getDifficultyLevel(): returns the difficulty level of the exercise as a string
* setName(name: string): sets the name of the exercise
* setDescription(description: string): sets the description of the exercise
* setMuscleGroup(muscle\_group: string): sets the muscle group targeted by the exercise
* setEquipment(equipment: string): sets the equipment needed for the exercise
* setDifficultyLevel(difficulty\_level: string): sets the difficulty level of the exercise.

Class name: ExerciseLibrary

Properties:

* List<Exercise> exercises: A list of all available exercises in the library.

Methods:

* addExercise(Exercise exercise): Adds an exercise to the library.
* removeExercise(Exercise exercise): Removes an exercise from the library.
* searchExerciseByName(String name): Searches for an exercise in the library by name.
* searchExerciseByMuscleGroup(String muscleGroup): Searches for an exercise in the library by muscle group.
* getBeginnerWorkouts(): Returns a list of beginner workouts.
* getIntermediateWorkouts(): Returns a list of intermediate workouts.

Class name: User

Properties:

name

age

weight

gender

fitness

password

Methods:

* setName(name: string): void - sets the user's name
* setAge(age: int): void - sets the user's age
* setWeight(weight: float): void - sets the user's weight
* setHeight(height: float): void - sets the user's height
* setGender(gender: string): void - sets the user's gender
* setFitnessLevel(fitnessLevel: string): void - sets the user's fitness level
* getName(): string - returns the user's name
* getAge(): int - returns the user's age
* getWeight(): float - returns the user's weight
* getHeight(): float - returns the user's height
* getGender(): string - returns the user's gender
* getFitnessLevel(): string - returns the user's fitness level
* saveUserToFile(fileName: string): void - saves the user's information to a file
* loadUserFromFile(fileName: string): void - loads the user's information from a file

Class Name: WorkoutsCreation

Attributes:

* exerciseLibrary (list): a list of Exercise objects available for selection by the user

Methods:

* **init**(self, exercise\_library): Initializes the WorkoutsCreation object with a list of Exercise objects available for selection
* createWorkout(self, user): Allows the user to select exercises from the exercise library and create a custom workout. Returns a Workout object.
* addExerciseToLibrary(self, exercise): Adds a new Exercise object to the exercise library.
* removeExerciseFromLibrary(self, exercise): Removes an Exercise object from the exercise library.
* updateExerciseInLibrary(self, exercise): Updates an existing Exercise object in the exercise library.

Class Name: Workout

Attributes:

* exercises (list): a list of Exercise objects in the workout
* duration (int): duration of the workout in minutes
* difficulty (int): difficulty level of the workout on a scale of 1-10

Methods:

* **init**(self, exercises, duration, difficulty): Initializes the Workout object with a list of Exercise objects, duration, and difficulty level.
* add\_exercise(self, exercise): Adds an Exercise object to the workout.
* remove\_exercise(self, exercise): Removes an Exercise object from the workout.
* update\_exercise(self, exercise): Updates an existing Exercise object in the workout.
* calculate\_calories\_burned(self, user): Calculates the estimated number of calories burned during the workout based on the user's weight and the intensity level of the workout.