System Overview

Entity	Purpose	Function	End points	HTTP request verb	Input parameter	Return values	Authentication
Users	Manage User data, Attributes: Email, Password, Admin, username, mobile_number	print all users	@user.get("/")	http://localhost:5000/users	None	list of users	none
		print users of primary key	@user.get("/<int:id>")	http://localhost:5000/users/5	primary id	user	none
		Register new user	@user.post("/")	http://localhost:5000/users	None	created user	none
		Delete User (admin or user can delete himself)	@user.delete("/<string:email>")	http://localhost:5000/users/arni@email.com	user email	deleted user id and email	user or admin login
		Amend User (only by the user itself)	@user.put("/update")	http://localhost:5000/users/update	User details	updated uder details	user login
		login user	@user.post("/login")	http://localhost:5000/users/login	email, password	email, token, admin status, user_id	user or admin login
Workouts	Manage Workout data, Attributes: workout_name, rest time, rounds, body_region, level, progres, date, Foreign key: user_id	print all workouts of logged in user	@workout.get("/")	http://localhost:5000/workouts	none	all workout data sets	user or admin login
		print workout of a speific id	@workout.get("/<int:id>")	http://localhost:5000/workouts/22	primary id	workout data set	none
		Create new workout with the user id of logged in user	@workout.post("/")	http://localhost:5000/workouts	workout details	details of created workout	user or admin login
		Delete workout (only admin or user who created it can delete)	@workout.delete("/<string:workout_name>")	http://localhost:5000/workouts/core1	workout_name	user id and name of deleted workout	user or admin login
		Amend workout (only user who created it)	@workout.put("/update/<string:workout_name>")	http://localhost:5000/workouts/update/Core2	workout_name	all workout data	user or admin login
Progres	Manage Progres records, Attributes: progres_name, weight, mid_arm, waist, chest, hip, test_score_date Foreign key: user_id	print all progres records of logged in user	@progres.get("/")	http://localhost:5000/progresses	none	all progres data sets	user or admin login
		print progres record of a speific id	@progres.get("/<int:id>")	http://localhost:5000/progresses/22	primary id	progres data set	none
		Create new progres record with the user id of logged in user	@progres.post("/")	http://localhost:5000/progresses	progres details	details of created progres	user or admin login
		Delete progres record (only admin or user who created it can delete)	@progres.delete("/<string:progres_name>")	http://localhost:5000/progresses/Advanced	progres_name	user id and name of deleted progres	user or admin login
		Amend progres record (only user who created it)	@progres.put("/update/<string:progres_name>")	http://localhost:5000/progresses/update/Core2	progres_name	all progres data	user or admin login
Workout_exercises	Mapping table to connect workouts and exercises Attributes: id, Date, Secondary keys: Workout_id, Exercise_id	Display all exercises belonging to a workout	@workout_exercise.get("/<string:workout_name>")	http://localhost:5000/workout_exercises/Upper B	Workout name	workout data set and nested exercises	none
		Create 4 new workout_exercise entries, exercise random for an existing workout valid workout id required as secondary key	@workout_exercise.post("/<int:workout_id>")	http://localhost:5000/workout_exercises/2	workout_id	4 workout_exercise records	none
		Manually add exercises to workout	@workout_exercise.post("/<string:workout_name>")	http://localhost:5000/workout_exercises/Cardio3	workout_name, exercise_id	name of added exercise and workout	user or admin login
		Display workout exercise record	@workout_exercise.get("/<int:id>")	http://localhost:5000/workout_exercises/4	workout_exercise id	workout exercise record	none

System Overview

Entity	Purpouse	Function	End points	HTTP request verb	Input parameter	Return values	Authentication
Exercises		print all ecercises	@exercise.get("/")	http://localhost:5000/exercises	none	all exercise data sets	none
	Manage exercise data, Attributes:	print all exercises for a specific body region	@exercise.get("/<string:body_region>")	http://localhost:5000/exercises/core	body_region	exercise data sets for body region	none
	id, name, description, interval_time, repetitions,	Create new exercise (admin only)	@exercise.post("/")	http://localhost:5000/exercises	all exercise parameters	data set of created exercise	admin login
	body_region, level, weight	Delete exercise (only admin)	@exercise.delete("/<int:id>")	http://localhost:5000/exercises/4	exercise id	name and id of deleted exercise	admin login
		Amend exercise (only admin)	@exercise.put("/update")	http://localhost:5000/exercises/update	all exercise parameters	name, id and level of updated exercise	admin login