Fitness tracker

Purpose:

It is a Python-based fitness tracker that helps users calculate Information workouts, swimming and running Sessions. It tracks the start and end time, number of calories burned, and the type of workout you performed.

Structures of the program:

1. Parent Class: Workout
   This is a class used to model a generic workout.

- **Attributes**:
  - Start and end times of the session (using dateutil.parser for flexible input)
  - Calories burned (either given or auto-calculated)
  - Type of workout (default: Workout)
- **Key Functions**:
  - Getter and setter functions
    - get_calories(): Calculates calories if not provided.
    - get_duration(): Time between start and end.
    - get_start, get_end, and get_kind
    - set_start, set_end, and set_calories
  - __str__(): Nicely formats the session info when printed.
  - __eq__(): Allows comparison between two workout sessions.

2. Child Class: Running, Swimming
   These two classes **inherit from Workout**, and change only what's needed.
   Running
   - Calorie burn rate: **600 per hour**
   - kind: "Running"

   Swimming
   - Calorie burn rate: **500 per hour**
   - kind: "Swimming"

3. Calories calculations
   if (calories == None) then it return - duration in hours *calories_per_hour
   otherwise it return calories

4. Example:

   w1 = Workout("2021-01-01 3:00 PM", "2021-01-01 4:00 PM")

   r1 = Running("2021-01-02 6:00 AM", "2021-01-02 7:30 AM")

```
s1 = Swimming("2021-01-03 8:00 AM", "2021-01-03 9:15 AM")


all_sessions = [w1, r1, s1]
for session in all_sessions:
   print(session)
```

through this code we get the output this:

```
Workout 1:00:00
                200 Calories
Running  1:30:00
                300 Calories
Swimming  1:15:00
                  250 Calories
```

5. <u>Advantages:</u>
   It is flexible, which means that you can easily add other child classes like Badminton, Aerobic.
   Output is readable.
   It includes inheritance, encapsulation, and method overriding.