

Date	Test information	The file name	The object for test	Expecting output	Test error
20/05/21	create a class that has readable name	calorie_recorder.rb	<pre> name = "saman" calorie = Calorierecorder.new(name) # this code is to check if the user get prompt to enter name it "display the user to enter their name" do expect(calorie.name).to eq(name) end </pre>	<p>Calorierecorder should have readable name</p> <p>Finished in 0.0025 seconds (files took 0.20508 seconds to load) 1 example, 0 failures</p>	<p>An error occurred while loading . ./spec/calorie_tracker_spec.rb. Failure/Error: describe Calorierecorder do name = "saman" calorie = Calorierecorder.new(name) # this code is to check if the user get prompt to enter name it "display the user to enter their name" do expect(calorie.name).to eq(name) end # this test is for to read the record of calorie # it "should read the total cal" do # total_cal = [{food: "chicken", calorie: 300, time: "morning"}] # end end NameError: uninitialized constant Calorierecorder</p>
20/05/21	create a welcome method	calorie_recorder.rb	<pre> def welcome puts "Welcome #{@name} to my calorie recorder app".colorize(:blue) puts "In this app you will be able to recorde your daily food intake with calorie and time of the day.".colorize(:blue) puts "Please just follow the instruction and you will be able to use this app easily.".colorize(:blue) puts "would you like to continue?(Y/N)".colorize(:blue) end </pre>	<p>.Welcome to my calorie recorder app Finished in 0.00653 seconds (files took 0.19183 seconds to load) 2 examples, 0 failures</p>	<p>NoMethodError: undefined method `welcome' for #<Calorierecorder:0x00007fffd66fb1d8 @name="saman", @record=[]> # ./spec/calorie_tracker_spec.rb:15:in `block (2 levels) in <top (required)>' Finished in 0.00675 seconds (files took 0.19513 seconds to load) 2 examples, 1 failure</p>
20/05/21	create a option method	calorie_recorder.rb	<pre> def option puts "#{name}:".colorize(:cyan) puts "would you like to record your calorie intake, calculate your BMI or check the calorie history?".colorize(:cyan) puts "please choose from the selection".colorize(:cyan) puts "(calorie,bmi, history)".colorize(:cyan) end </pre>	<p>would you like to record your calorie intake, calculate your BMI or check the calonie history? please choose from the selection (calorie,bmi, history) (calorie, bmi, history) .</p> <p>Finished in 0.00697 seconds (files took 0.25121 seconds to load) 3 examples, 0 failures</p>	<p>1) Calorierecorder should display set of option to the user Failure/Error: expect(user.option).to eq(puts "(calorie, bmi, history)") NoMethodError: undefined method `option' for #<Calorierecorder:0x00007fffd0bec800 @name="saman", @record=[]> # ./spec/calorie_tracker_spec.rb:19:in `block (2 levels) in <top (required)>' Finished in 0.00802 seconds (files took 0.26721 seconds to load) 3 examples, 1 failure Failed examples: rspec ./spec/calorie_tracker_spec.rb:18 # Calorierecorder should display set of option to the user</p>
20/05/21	create a get_data method	calorie_recorder.rb	<pre> def get_data(type, nmuber_of_cal, time_of_day) total = {food: type, calorie: nmuber_of_cal, time: time_of_day} @record << total end </pre>	<p>should display set of option to the user should get the type of food, number of calories, time of the day</p> <p>Finished in 0.00361 seconds (files took 0.26398 seconds to load) 4 examples, 0 failures</p>	<p>1) Calorierecorder should read the total cal Failure/Error: expect(calorie.get_calorei("chicken", 300, "morning")).to eq(total_cal) NameError: undefined local variable or method `calorie' for #<RSpec::ExampleGroups::Calorierecorder "should read the total cal" (./spec/calorie_tracker_spec.rb:22)> Did you mean? caller # ./spec/calorie_tracker_spec.rb:24:in `block (2 levels) in <top (required)>' Finished in 0.00682 seconds (files took 0.25344 seconds to load) 4 examples, 1 failure Failed examples: rspec ./spec/calorie_tracker_spec.rb:22 # Calorierecorder should read the total cal</p>

20/05/21	create display_data method	calorie_recorder.rb	<pre> def display_data @record.each do item puts "Meal -> #{item[:food]}, #{item[:calorie]} -> calories, Time -> #{item[:time]}" end end </pre>	<p>Meal -> chicken, 300 -> calories, Time -> morning should display the data that of user input</p> <p>Finished in 0.0038 seconds (files took 0.26292 seconds to load) 5 examples, 0 failures</p>	<p>1) Calorierecorder should display the data that of user input Failure/Error: expect(user.display_data).to eq(puts "Meal => chicken, Calorie => 300, Time => Afternoon")</p> <p>NoMethodError: undefined method 'display_data' for #<Calorierecorder:0x00007ffec3f63a0 @name="saman", @record=[{:food=>"chicken", :calorie=>300, :time=>"morning"}]> Did you mean? display # ./spec/calorie_tracker_spec.rb:28:in `block (2 levels) in <top (required)>'</p> <p>Finished in 0.00377 seconds (files took 0.25406 seconds to load) 5 examples, 1 failure</p> <p>Failed examples:</p> <p>rspec ./spec/calorie_tracker_spec.rb:27 # Calorierecorder should display the data that of user input</p>
20/05/21	create a history method	calorie_recorder.rb	<pre> def history if @record.length == 0 puts "Your log is empty." elsif @record.length >= 1 @record.each_with_index do item, index puts "#{index += 1}. Meal: #{item[:food]}, #{item[:calorie]} Calories, #{item[:time]} " end puts "your meal intake are #{@record.length} times!". colorize(:green) end end </pre>	<p>1. Meal: chicken, 300 Calories, morning your meal intake are 1 times! your meal intake are [{:food=>"chicken", :calorie=>300, :time=>"morning"}] times! should display the history of data input to the user</p> <p>Finished in 0.00412 seconds (files took 0.25542 seconds to load) 6 examples, 0 failures</p>	<p>1) Calorierecorder should display the history of data input to the user Failure/Error: puts "your meal intake are #{@record.length} times!"</p> <p>NoMethodError: undefined method 'length' for nil:NilClass # ./spec/calorie_tracker_spec.rb:36:in `block (2 levels) in <top (required)>'</p> <p>Finished in 0.00372 seconds (files took 0.26204 seconds to load) 6 examples, 1 failure</p> <p>Failed examples:</p> <p>rspec ./spec/calorie_tracker_spec.rb:35 # Calorierecorder should display the history of data input to the user</p>
21/05/21	create a class Bmi and has to variable and it should be readable	bmi.rb	<pre> class Bmi attr_reader :weight, :height # in this method create two instance variables to get the weight and height of the user def initialize(weight, height) @weight = weight @height = height end end </pre>	<p>Bmi should have a readable weight</p> <p>Finished in 0.00726 seconds (files took 0.25776 seconds to load) 7 examples, 0 failures</p>	<p>describe Bmi do it "should have a readable weight and height" do weight = 80 height = 1.82 bmi = Bmi.new(weight, height) expect(bmi).to eq(weight, height) end end</p> <p>NameError: uninitialized constant Bmi # ./spec/calorie_tracker_spec.rb:45:in `<top (required)>'</p> <p>Finished in 0.00003 seconds (files took 0.32497 seconds to load) 0 examples, 0 failures, 1 error occurred outside of examples</p>

21/05/21	create a class Bmi and has to variable and it should be readable	bmi.rb	<pre> class Bmi attr_reader :weight, :height # in this method create two instance variables to get the weight and height of the user def initialize(weight, height) @weight = weight @height = height end end </pre>	<p>Bmi should have a readable weight should have a readable height</p> <p>Finished in 0.00608 seconds (files took 0.27647 seconds to load) 8 examples, 0 failures</p>	<p>2) Bmi should have a readable height Failure/Error: bmi = Bmi.new(weight, height)</p> <p>ArgumentError: wrong number of arguments (given 2, expected 0) # ./spec/calorie_tracker_spec.rb:55:in `initialize' # ./spec/calorie_tracker_spec.rb:55:in `new' # ./spec/calorie_tracker_spec.rb:55:in `block (2 levels) in <top (required)>'</p> <p>Finished in 0.00649 seconds (files took 0.25859 seconds to load) 8 examples, 2 failures</p> <p>Failed examples:</p> <p>rspec ./spec/calorie_tracker_spec.rb:46 # Bmi should have a readable weight rspec ./spec/calorie_tracker_spec.rb:52 # Bmi should have a readable height</p>
21/05/21	create a method to calculate the bmi	bmi.rb	<pre> def get_bmi(total_bmi) @total_bmi = total_bmi @total_bmi = @weight / (height * height) # puts @total_bmi end </pre>	<p>should calculate the bmi</p> <p>Finished in 0.0061 seconds (files took 0.25952 seconds to load) 9 examples, 0 failures</p>	<p>1) Bmi should calculate the bmi Failure/Error: expect(bmi.get_bmi(total_bmi)).to eq (total_bmi)</p> <p>NoMethodError: undefined method `get_bmi' for #<Bmi:0x00007ffed8a6c28 @weight=80, @height=1.82> # ./spec/calorie_tracker_spec.rb:61:in `block (2 levels) in <top (required)>'</p> <p>Finished in 0.0063 seconds (files took 0.26058 seconds to load) 9 examples, 1 failure</p> <p>Failed examples:</p> <p>rspec ./spec/calorie_tracker_spec.rb:59 # Bmi should calculate the bmi</p>
		bmi.rb	<pre> def display if get_bmi(@total_bmi).round(1) >= 18.4 && get_bmi(@total_bmi).round(1) <= 24.9 puts "Good news, your BMI is #{get_bmi(@total_bmi).round(1)}, you have healthy weight.".colorize(:green) elsif get_bmi(@total_bmi).round(1) >= 25 puts "Your BMI is a bit high #{get_bmi(@total_bmi).round(1)}, you are overweight".colorize(:red) puts "If you reduce your calorie intake by 500 calories a day, you will be able to lose (450g) in a week".colorize(:red) elsif get_bmi(@total_bmi).round(1) <= 18.3 puts "Your BMI is a bit low #{get_bmi(@total_bmi).round(1)}, you are underweight".colorize(:yellow) puts "If you increase your calorie intake by 500 calories a day, you will be able to gain (450g) in a week.".colorize(:yellow) end end </pre>	<p>Good news, your BMI is 24.2, you have healthy weight. #<RSpec::ExampleGroups::Bmi:0x00007ffcbac9b30> should display the bmi to the user</p> <p>Finished in 0.00658 seconds (files took 0.26707 seconds to load) 10 examples, 0 failures</p>	<p>1) Bmi should display the bmi to the user Failure/Error: expect(bmi.display).to eq()</p> <p>ArgumentError: wrong number of arguments (given 0, expected 1) # ./spec/calorie_tracker_spec.rb:65:in `block (2 levels) in <top (required)>'</p> <p>Finished in 0.0072 seconds (files took 0.27128 seconds to load) 10 examples, 1 failure</p> <p>Failed examples:</p>