

ID	Feature ID	Test Case	Test Data	Expected Result	Error Output	Actual Result	Status
1	1	Display Welcome Message	Welcome message def welcome_message puts "Test welcome message" end	Fail. No method exists.	<p>Welcome should print the welcome message (FAILED - 1)</p> <p>Failures:</p> <p>1) Welcome should print the welcome message Failure/Error: expect(welcome. welcome_message).to eq(puts "Welcome")</p> <p>NoMethodError: undefined method `welcome_message' for #<Welcome:0x00007fffe4e89810> # ./spec/make_10_spec.rb:21:in `block (2 levels) in <top (required)>'</p> <p>Finished in 0.00359 seconds (files took 0.20021 seconds to load) 1 example, 1 failure</p> <p>Failed examples:</p> <p>rspec ./spec/make_10_spec.rb:20 # Welcome should print the welcome messageFailure/Error: require_relative "../index" NoMethodError: undefined method `welcome_message' for #<Welcome:0x00007fffb9dd7de8> # ./index.rb:6:in `<top (required)>' # ./spec/make_10_spec.rb:5:in `require_relative' # ./spec/make_10_spec.rb:5:in `<top (required)>' No examples found. Finished in 0.00019 seconds (files took 0.16583 seconds to load) 0 examples, 0 failures, 1 error occurred outside of examples</p>	<p>Welcome Test welcome message Welcome should print the welcome message</p> <p>Finished in 0.00646 seconds (files took 0.18918 seconds to load) 1 example, 0 failures</p>	passed

2	2	Displays Instructions	<pre>def instructions system("clear") puts ("Instructions"). colorize(:white).on_green puts "How the game works:" puts "Make 10 is played by taking four numbers, and combining different mathematical operations to solve for a total of 10. Numbers range from 1000 to 9999." puts " " puts "Rules:" puts "Players are free to use addition(+), subtraction(-), multiplication(*), division(/), parentheses() and exponentials(**)." puts "Players can't change the order of the numbers, and can't use decimals or square root functions." puts " " puts "Examples:" puts "`1234` can be solved as `1+2+3+4 = 10` and also `(1 * 2 * 3) + 4 = 10`" puts "But you cannot solve it as `4+3+2+1 = 10`" puts " " endpassword: 1</pre>	Fail. No method exists.	<p>Welcome should print the instructions (FAILED - 1)</p> <p>Failures:</p> <p>1) Welcome should print the instructions Failure/Error: expect(welcome.instructions).to eq (puts "Instructions")</p> <p>NoMethodError: undefined method `instructions' for #<Welcome: 0x00007ffff422fac8> # ./spec/make_10_spec.rb:13:in `block (2 levels) in <top (required)>'</p> <p>Finished in 0.00279 seconds (files took 0.37004 seconds to load) 1 example, 1 failure</p> <p>Failed examples:</p> <p>rspec ./spec/make_10_spec.rb:12 # Welcome should print the instructions</p>	<p>Instructions should print the instructions</p> <p>Finished in 0.0142 seconds (files took 0.24339 seconds to load) 1 example, 0 failures</p>	passed
---	---	-----------------------	---	-------------------------	---	--	--------

3	3	Display main menu	<pre>def ask_user_start quit_status = false until quit_status != false answer = @@prompt. select("What would you like to do?", %w(calculator instructions quit)) if answer == "calculator" system("clear") ask_for_digits elsif answer == "instructions" system("clear") instructions ask_user_start else answer == "quit" puts "Stay well. Come back next time you are stuck on a number!" quit_status = true exit end end end endpassword: 123456</pre>	Fail. No method exists.	<p>Welcome should display main menu (FAILED - 1)</p> <p>Failures:</p> <p>1) Welcome should display main menu Failure/Error: expect(welcome.ask_user_start). to eq(puts "Menu")</p> <p>NameError: undefined local variable or method `welcome' for #<RSpec::ExampleGroups::Welcome: 0x00007ffc27b3670> # ./spec/make_10_spec.rb:17:in `block (2 levels) in <top (required)>'</p> <p>Finished in 0.00361 seconds (files took 0.24111 seconds to load) 1 example, 1 failure</p> <p>Failed examples:</p> <p>rspec ./spec/make_10_spec.rb:16 # Welcome should display main menu</p>	<p>Finished in 2.35 seconds (files took 0.24873 seconds to load) 1 example, 0 failures</p>	passed
4	4	Sum two numbers	<pre>it "should return the result of a mathematical operation" do expect(operation.add (1,2)).to eq(3) end</pre>	Fail. No method exists.	<p>Operations should return the result of a mathematical operation (FAILED - 1)</p> <p>Failures:</p> <p>1) Operations should return the result of a mathematical operation Failure/Error: expect(operation.add(1,2)).to eq(3)</p> <p>NoMethodError: undefined method `add' for #<Operations: 0x00007ffe6dd6fb0> # ./spec/make_10_spec.rb:25:in `block (2 levels) in <top (required)>'</p> <p>Finished in 0.00338 seconds (files took 0.26146 seconds to load) 1 example, 1 failure</p> <p>Failed examples:</p> <p>rspec ./spec/make_10_spec.rb:24 # Operations should return the result of a mathematical operation</p>	<p>Operations should return the result of a mathematical operation</p> <p>Finished in 0.0051 seconds (files took 0.24503 seconds to load) 1 example, 0 failures</p>	passed

5	4	Sum two numbers	it "should return the result of a mathematical operation" do expect(operation.add (1,2)).to eq(4) end	Fail. It is trying to add 1 + 2 to equal 4.	<p>Operations should return the result of a mathematical operation (FAILED - 1)</p> <p>Failures:</p> <p> 1) Operations should return the result of a mathematical operation Failure/Error: expect(operation.add(1,2)).to eq(4)</p> <p> expected: 4 got: 3</p> <p> (compared using ==) # ./spec/make_10_spec.rb:26:in `block (2 levels) in <top (required)>'</p> <p>Finished in 0.40371 seconds (files took 0.24519 seconds to load) 1 example, 1 failure</p> <p>Failed examples:</p> <p>rspec ./spec/make_10_spec.rb:25 # Operations should return the result of a mathematical operation</p>	<p>Operations should return the result of a mathematical operation</p> <p>Finished in 0.00468 seconds (files took 0.24829 seconds to load) 1 example, 0 failures</p>	
6	5	Subtract two numbers	it "should return the result of a mathematical operation" do expect(operation.subt (1,2)).to eq(-1) end	Fail. No method exists.	<p>Operations should return the result of a mathematical operation (FAILED - 1)</p> <p>Failures:</p> <p> 1) Operations should return the result of a mathematical operation Failure/Error: expect(operation.subt(1,2)).to eq(-1)</p> <p> NoMethodError: undefined method `subt' for #<Operations: 0x00007ffff6ec6af0> Did you mean? stub # ./spec/make_10_spec.rb:30:in `block (2 levels) in <top (required)>'</p> <p>Finished in 0.00337 seconds (files took 0.25732 seconds to load) 1 example, 1 failure</p> <p>Failed examples:</p> <p>rspec ./spec/make_10_spec.rb:29 # Operations should return the result of a mathematical operation</p>	<p>Operations should return the result of a mathematical operation</p> <p>Finished in 0.00416 seconds (files took 0.2323 seconds to load) 1 example, 0 failures</p>	passed

7	5	Subtract two numbers	it "should return the result of a mathematical operation" do expect(operation.subt (1,2)).to eq(0) end	Fail. It is trying to subtract 2 from 1 to equal 0.	<p>Operations should return the result of a mathematical operation (FAILED - 1)</p> <p>Failures:</p> <p>1) Operations should return the result of a mathematical operation Failure/Error: expect(operation.subt(1,2)).to eq(0)</p> <p> expected: 0 got: -1</p> <p> (compared using ==) # ./spec/make_10_spec.rb:30:in `block (2 levels) in <top (required)>'</p> <p>Finished in 0.03446 seconds (files took 0.25347 seconds to load) 1 example, 1 failure</p> <p>Failed examples:</p> <p>rspec ./spec/make_10_spec.rb:29 # Operations should return the result of a mathematical operation</p>	<p>Operations should return the result of a mathematical operation</p> <p>Finished in 0.00475 seconds (files took 0.23849 seconds to load) 1 example, 0 failures</p>	
8	6	Multiply two numbers	it "should return the result of a mathematical operation" do expect(operation.mult (1,2)).to eq(2) end	Fail. No method exists.	<p>Operations should return the result of a mathematical operation (FAILED - 1)</p> <p>Failures:</p> <p>1) Operations should return the result of a mathematical operation Failure/Error: expect(operation.mult(1,2)).to eq(2)</p> <p> NoMethodError: undefined method `mult' for #<Operations: 0x00007fffeae208> # ./spec/make_10_spec.rb:34:in `block (2 levels) in <top (required)>'</p> <p>Finished in 0.0036 seconds (files took 0.24759 seconds to load) 1 example, 1 failure</p> <p>Failed examples:</p> <p>rspec ./spec/make_10_spec.rb:33 # Operations should return the result of a mathematical operation</p>	<p>Operations should return the result of a mathematical operation</p> <p>Finished in 0.00464 seconds (files took 0.25612 seconds to load) 1 example, 0 failures</p>	passed

9	6	Multiply two numbers	it "should return the result of a mathematical operation" do expect(operation.mult (1,2)).to eq(3) end	Fail. It is trying to multiple 1 by 2 to equal 3.	<p>Operations should return the result of a mathematical operation (FAILED - 1)</p> <p>Failures:</p> <p>1) Operations should return the result of a mathematical operation Failure/Error: expect(operation.mult(1,2)).to eq(3)</p> <p> expected: 3 got: 2</p> <p> (compared using ==) # ./spec/make_10_spec.rb:34:in `block (2 levels) in <top (required)>'</p> <p>Finished in 0.03882 seconds (files took 0.25139 seconds to load) 1 example, 1 failure</p> <p>Failed examples:</p> <p>rspec ./spec/make_10_spec.rb:33 # Operations should return the result of a mathematical operation</p>	<p>Operations should return the result of a mathematical operation</p> <p>Finished in 0.00493 seconds (files took 0.24015 seconds to load) 1 example, 0 failures</p>	
10	7	Divide two numbers	it "should return the result of a mathematical operation" do expect(operation.div (1.0,2)).to eq(0.5) end	Fail. No method exists.	<p>Operations should return the result of a mathematical operation (FAILED - 1)</p> <p>Failures:</p> <p>1) Operations should return the result of a mathematical operation Failure/Error: expect(operation.div(1.0,2)).to eq (0.5)</p> <p> NoMethodError: undefined method `div' for #<Operations: 0x00007fffc4be4d78> # ./spec/make_10_spec.rb:38:in `block (2 levels) in <top (required)>'</p> <p>Finished in 0.00365 seconds (files took 0.249 seconds to load) 1 example, 1 failure</p> <p>Failed examples:</p> <p>rspec ./spec/make_10_spec.rb:37 # Operations should return the result of a mathematical operation</p>	<p>Operations should return the result of a mathematical operation</p> <p>Finished in 0.00416 seconds (files took 0.23877 seconds to load) 1 example, 0 failures</p>	passed

11	7	Divide two numbers	it "should return the result of a mathematical operation" do expect(operation.div (1.0,2)).to eq(0.15) end	Fail. It is attempting to divide 1.0 by 2.0 to result in 0.15	Operations should return the result of a mathematical operation (FAILED - 1) Failures: 1) Operations should return the result of a mathematical operation Failure/Error: expect(operation.div(1.0,2)).to eq(0.15) expected: 0.15 got: 0.5 (compared using ==) # ./spec/make_10_spec.rb:38:in `block (2 levels) in <top (required)>' Finished in 0.03686 seconds (files took 0.2377 seconds to load) 1 example, 1 failure Failed examples: rspec ./spec/make_10_spec.rb:37 # Operations should return the result of a mathematical operation	Operations should return the result of a mathematical operation Finished in 0.00465 seconds (files took 0.23558 seconds to load) 1 example, 0 failures	
12	8	Exponential calculation on two numbers	it "should return the result of a mathematical operation" do expect(operation.exp (1,2)).to eq(1) end	Fail. No method exists.	Operations should return the result of a mathematical operation (FAILED - 1) Failures: 1) Operations should return the result of a mathematical operation Failure/Error: expect(operation.exp(1,2)).to eq(1) NoMethodError: undefined method `exp' for #<Operations: 0x00007ffcd6158f8> # ./spec/make_10_spec.rb:42:in `block (2 levels) in <top (required)>' Finished in 0.00338 seconds (files took 0.23693 seconds to load) 1 example, 1 failure Failed examples: rspec ./spec/make_10_spec.rb:41 # Operations should return the result of a mathematical operation	Operations should return the result of a mathematical operation Finished in 0.00661 seconds (files took 0.26128 seconds to load) 1 example, 0 failures	passed

13	8	Exponential calculation on two numbers	it "should return the result of a mathematical operation" do expect(operation.exp(1,2)).to eq(2) end	Fail. It is expecting 1 to the power of 2 to result in 2	<p>Operations should return the result of a mathematical operation (FAILED - 1)</p> <p>Failures:</p> <p>1) Operations should return the result of a mathematical operation Failure/Error: expect(operation.exp(1,2)).to eq(2)</p> <p>expected: 2 got: 1</p> <p>(compared using ==) # ./spec/make_10_spec.rb:42:in `block (2 levels) in <top (required)>'</p> <p>Finished in 0.03486 seconds (files took 0.2596 seconds to load) 1 example, 1 failure</p> <p>Failed examples:</p> <p>rspec ./spec/make_10_spec.rb:41 # Operations should return the result of a mathematical operation</p>	<p>Operations should return the result of a mathematical operation</p> <p>Finished in 0.00459 seconds (files took 0.23624 seconds to load) 1 example, 0 failures</p>	passed
----	---	--	--	--	---	--	--------