Ruby Dynamic Tests Tasks

First test for checking an ace will true and false:

```ruby
require('minitest/autorun')
require_relative('task2.rb')


class Task2Test < MiniTest::Test

  def setup
    @card1 = Card.new("ace", 1)
    @card2 = Card.new("jack", 11)
    @card_game = CardGame.new()
  end

def test_check_for_ace_is_ace
  assert_equal(true, @card_game.checkforAce(@card1))
end

def test_check_for_ace_is_not_ace
  assert_equal(false, @card_game.checkforAce(@card2))
end




  end
```

Screenshot showing the tests can't run because the Cardgame class doesn't have the correct end statements at the end of the class, too many for the 'highest_card' method and also because def was misspelled ruby was not recognising when the functions started and ended.

```
➜  Texting_Exercises git:(master) ✗ ruby task2_spec.rb
task2_spec.rb:2:in `require_relative': /Users/user/codeclan_work/pda/Texting_Exercises/task2.rb:28: syntax error, unexpected end-of-input, expecting keyword_end (SyntaxError)
        from task2_spec.rb:2:in `<main>'
```

Correcting the spelling of def and adding the right end statements has changed the error. Now the program can't run because the parameters of the 'highest_value' method are not separated by a comma.

```
➜  Texting_Exercises git:(master) ✗ ruby task2_spec.rb
task2_spec.rb:2:in `require_relative': /Users/user/codeclan_work/pda/Texting_Exercises/task2.rb:13: syntax error, unexpected tIDENTIFIER, expecting ')' (SyntaxError)
  def highest_card(card1 card2)
                          ^
/Users/user/codeclan_work/pda/Texting_Exercises/task2.rb:30: syntax error, unexpected keyword_end, expecting end-of-input
        from task2_spec.rb:2:in `<main>'
```

The next error is appearing because a double equals sign (==) is required to test for equality.

```
 1 runs, 0 assertions, 0 failures, 1 errors, 0 skips
➜  Texting_Exercises git:(master) × ruby task2_spec.rb
Run options: --seed 16705

# Running:

EE

Finished in 0.001047s, 1910.2198 runs/s, 0.0000 assertions/s.

  1) Error:
Task2Test#test_check_for_ace_is_not_ace:
NoMethodError: undefined method `value=' for #<Card:0x007fdba42a2ec0 @suit="jack", @value=11>
Did you mean?  value
    /Users/user/codeclan_work/pda/Texting_Exercises/task2.rb:6:in `checkforAce'
    task2_spec.rb:18:in `test_check_for_ace_is_not_ace'

  2) Error:
Task2Test#test_check_for_ace_is_ace:
NoMethodError: undefined method `value=' for #<Card:0x007fdba42a2858 @suit="ace", @value=1>
Did you mean?  value
    /Users/user/codeclan_work/pda/Texting_Exercises/task2.rb:6:in `checkforAce'
    task2_spec.rb:14:in `test_check_for_ace_is_ace'

2 runs, 0 assertions, 0 failures, 2 errors, 0 skips
➜  Texting_Exercises git:(master) × 
```

The tests run once this is fixed:

```
➜  Texting_Exercises git:(master) × ruby task2_spec.rb
Run options: --seed 45379

# Running:

..

Finished in 0.000917s, 2181.0251 runs/s, 2181.0251 assertions/s.

2 runs, 2 assertions, 0 failures, 0 errors, 0 skips
➜  Texting_Exercises git:(master) × 
```

Second tests for finding the higher card:

```ruby
def test_highest_card_returns_jack_correct
  assert_equal(@card2, @card_game.highest_card(@card1, @card2))
end

def test_highest_card_returns_jack_incorrect
  assert_equal(@card2, @card_game.highest_card(@card2, @card1))
end
```

The second is failing because the method refers to card when the parameters are card1 and card2. The first one is passing because the value of @card1 is not higher than the value of @card2 so it skips the statement with the error in it.

```
[→ Texting_Exercises git:(master) × ruby task2_spec.rb
Run options: --seed 9323

# Running:

...E

Finished in 0.001062s, 3766.4786 runs/s, 2824.8589 assertions/s.

  1) Error:
Task2Test#test_highest_card_returns_jack_incorrect:
NameError: undefined local variable or method `card' for #<CardGame:0x007f911ca99c78>
Did you mean?  card2
               card1
    /Users/user/codeclan_work/pda/Texting_Exercises/task2.rb:15:in `highest_card'
    task2_spec.rb:26:in `test_highest_card_returns_jack_incorrect'

4 runs, 3 assertions, 0 failures, 1 errors, 0 skips
```

Once this is fixed there is a new failure message because .name is not a defined method.

```
→ Texting_Exercises git:(master) × ruby task2_spec.rb
Run options: --seed 32199

# Running:

E...

Finished in 0.001049s, 3813.1555 runs/s, 2859.8667 assertions/s.

  1) Error:
Task2Test#test_highest_card_returns_jack_incorrect:
NoMethodError: undefined method `name' for #<Card:0x007f9ed2032d48 @suit="jack", @value=11>
    /Users/user/codeclan_work/pda/Texting_Exercises/task2.rb:15:in `highest_card'
    task2_spec.rb:26:in `test_highest_card_returns_jack_incorrect'

4 runs, 3 assertions, 0 failures, 1 errors, 0 skips
→ Texting_Exercises git:(master) × ▉
```

To fix this I've removed the .name so that it runs the card1 object. The tests now run:

```
▶ Texting_Exercises git:(master) × ruby task2_spec.rb
un options: --seed 44827

 Running:

...

inished in 0.001004s, 3984.0634 runs/s, 3984.0634 assertions/s.

 runs, 4 assertions, 0 failures, 0 errors, 0 skips
▶ Texting_Exercises git:(master) × ▉
```

Third test for totalling all the card values:

```ruby
def test_cards_total
  cards = [@card1, @card2]
  assert_equal("You have a total of 12", CardGame.cards_total(cards))
end
```

This test errors because the variable total has not been defined correctly. Needs corrected to total = 0.

```
➜  Texting_Exercises git:(master) ✗ ruby task2_spec.rb
Run options: --seed 4935

# Running:

E....

Finished in 0.001356s, 3687.3154 runs/s, 2949.8523 assertions/s.

  1) Error:
Task2Test#test_cards_total:
NameError: undefined local variable or method `total' for CardGame:Class
    /Users/user/codeclan_work/pda/Texting_Exercises/task2.rb:23:in `cards_total'
    task2_spec.rb:31:in `test_cards_total'

5 runs, 4 assertions, 0 failures, 1 errors, 0 skips
```

Once fixed we get a new error due to trying to return an integer as a string.

```
➜  Texting_Exercises git:(master) ✗ ruby task2_spec.rb
Run options: --seed 4935

# Running:

E....

Finished in 0.001356s, 3687.3154 runs/s, 2949.8523 assertions/s.

  1) Error:
Task2Test#test_cards_total:
NameError: undefined local variable or method `total' for CardGame:Class
    /Users/user/codeclan_work/pda/Texting_Exercises/task2.rb:23:in `cards_total'
    task2_spec.rb:31:in `test_cards_total'

5 runs, 4 assertions, 0 failures, 1 errors, 0 skips
```

The test is now failing for two reasons:
1) The string should have a space in before the closing "
2) The method is returning to early so is finishing after getting the value of the first card. This needs to be moved to after the for loop

```
 ➜  Texting_Exercises git:(master) ✗ ruby task2_spec.rb
Run options: --seed 21494

# Running:

F....

Finished in 0.001244s, 4019.2925 runs/s, 4019.2925 assertions/s.

  1) Failure:
Task2Test#test_cards_total [task2_spec.rb:31]:
Expected: "You have a total of 12"
  Actual: "You have a total of1"

5 runs, 5 assertions, 1 failures, 0 errors, 0 skips
```

Once fixed all the tests pass:

```
 ➜  Texting_Exercises git:(master) ✗ ruby task2_spec.rb
Run options: --seed 17525

# Running:

.....

Finished in 0.001122s, 4456.3278 runs/s, 4456.3278 assertions/s.

5 runs, 5 assertions, 0 failures, 0 errors, 0 skips
 ➜  Texting_Exercises git:(master) ✗ ▉
```