

Test Flask constructor: Not yet defined so expect a fail.

Traceback (most recent call last):

File "/home/ubuntu/workspace/test_riddle.py", line 4, in <module>
from run import app

ImportError: cannot import name app

app = Flask(__name__) *(Added to run.py)*

if __name__ == '__main__':

app.run(host=os.environ.get('IP', '0.0.0.0'), port=int(os.environ.get('PORT', 0)), debug=True)

Ran 0 tests in 0.000s

OK

//

test_index route: Test that '/'url returns the index page (html document). If the request is successful then a 200 response will be received. Route not established so expect a fail.

test_index (__main__.test_run) ... FAIL

=====

FAIL: test_index (__main__.test_run)

Traceback (most recent call last):

File "/home/ubuntu/workspace/test_riddle.py", line 15, in test_index
self.assertEqual(response.status_code, 200)

AssertionError: 404 != 200

Ran 1 test in 0.052s

FAILED (failures=1)

Added the code below which establishes a route and returns type text:

@app.route('/', methods=["GET", "POST"])

def index():

return "<h1>Hello World!</h1>"

test_index (__main__.test_run) ... ok

Ran 1 test in 0.019s

OK

//

Run a similar test to establish the leader_board route which is a page in the web application.

test_index (__main__.test_run) ... ok

test_leader_board (__main__.test_run) ... FAIL

=====

FAIL: test_leader_board (__main__.test_run)

Traceback (most recent call last):

File "/home/ubuntu/workspace/test_riddle.py", line 27, in test_leader_board
self.assertEqual(response.status_code, 200)
AssertionError: 404 != 200

Ran 2 tests in 0.021s

FAILED (failures=1)

Added the code below:

```
@app.route('/leader_board', methods=["GET", "POST"])  
def leader_board():  
    return "<h1>You are leading!</h1>"
```

```
test_index (__main__.test_run) ... ok  
test_leader_board (__main__.test_run) ... ok
```

Ran 2 tests in 0.021s

OK

////////////////////////////////////

Again run a similar test to establish the user route which is bound to the user function that creates the user page.

```
test_index (__main__.test_run) ... ok  
test_leader_board (__main__.test_run) ... ok  
test_user_page (__main__.test_run) ... FAIL
```

=====

FAIL: test_user_page (__main__.test_run)

Traceback (most recent call last):

File "/home/ubuntu/workspace/test_riddle.py", line 31, in test_user_page
self.assertEqual(response.status_code, 200)
AssertionError: 404 != 200

Ran 3 tests in 0.024s

FAILED (failures=1)

Added the code below:

```
@app.route('/<username>', methods=["GET", "POST"])  
def user(username):  
    return username
```

```
test_index (__main__.test_run) ... ok  
test_leader_board (__main__.test_run) ... ok
```

test_user_page (__main__.test_run) ... ok

Ran 3 tests in 0.022s

OK

//

Test that messages are bound to username, message route.

test_index (__main__.test_run) ... ok
test_leader_board (__main__.test_run) ... ok
test_message_can_be_sent (__main__.test_run) ... FAIL
test_user_page (__main__.test_run) ... ok

FAIL: test_message_can_be_sent (__main__.test_run)

Traceback (most recent call last):

File "/home/ubuntu/workspace/test_riddle.py", line 34, in test_message_can_be_sent
self.assertEqual(response.status_code, 200)
AssertionError: 404 != 200

Ran 4 tests in 0.024s

FAILED (failures=1)

Added the code below:

```
@app.route('/<username>/<message>')  
def send_message(username, message):  
    return "{0}: {1}\n".format(username, message)
```

test_index (__main__.test_run) ... ok
test_leader_board (__main__.test_run) ... ok
test_message_can_be_sent (__main__.test_run) ... ok
test_user_page (__main__.test_run) ... ok

Ran 4 tests in 0.022s

OK

//

Test that base html page exists:

test_for_base_html (__main__.test_run) ... FAIL
test_index (__main__.test_run) ... ok
test_leader_board (__main__.test_run) ... ok
test_message_can_be_sent (__main__.test_run) ... ok
test_user_page (__main__.test_run) ... ok

FAIL: test_for_base_html (__main__.test_run)

Ran 5 tests in 0.051s

FAILED (failures=1)

Added the text Home to the base page nav bar.

test_for_base_html (__main__.test_run) ... ok
test_index (__main__.test_run) ... ok
test_leader_board (__main__.test_run) ... ok
test_message_can_be_sent (__main__.test_run) ... ok
test_user_page (__main__.test_run) ... ok

Ran 5 tests in 0.045s

OK

//

Test that insex html page exists:

test_for_base_html (__main__.test_run) ... ok
test_for_index_html (__main__.test_run) ... FAIL
test_index (__main__.test_run) ... ok
test_leader_board (__main__.test_run) ... ok
test_message_can_be_sent (__main__.test_run) ... ok
test_user_page (__main__.test_run) ... ok

=====

FAIL: test_for_index_html (__main__.test_run)

Ran 6 tests in 0.047s

FAILED (failures=1)

Added Riddle Me This text to the page

test_for_base_html (__main__.test_run) ... ok
test_for_index_html (__main__.test_run) ... ok
test_index (__main__.test_run) ... ok
test_leader_board (__main__.test_run) ... ok
test_message_can_be_sent (__main__.test_run) ... ok
test_user_page (__main__.test_run) ... ok

Ran 6 tests in 0.123s

OK

//

Test that leader_board html page exists:

test_for_base_html (__main__.test_run) ... ok
test_for_index_html (__main__.test_run) ... ok
test_for_leader_board_html (__main__.test_run) ... FAIL


```
    return redirect(request.form["username"])
return render_template("index.html")
```

```
test_for_base_html (__main__.test_run) ... ok
test_for_index_html (__main__.test_run) ... ok
test_for_leader_board_html (__main__.test_run) ... ok
test_index (__main__.test_run) ... ok
test_leader_board (__main__.test_run) ... ok
test_message_can_be_sent (__main__.test_run) ... ok
test_riddle_page (__main__.test_run) ... ok
test_user_page (__main__.test_run) ... ok
```

Ran 8 tests in 0.080s

OK

//

(At this point the html pages were further developed. The base page was developed to include background logo and other common elements across all pages. A start bootstrap theme was uploaded to give some definition and the index page username form added.

The run.py file at this stage is reproduced below

```
app = Flask(__name__)

data = []

def write_to_file(filename, data):
    """Handle the process of writing data to a file using writelines method"""
    with open(filename, "a") as file:
        file.writelines(data)

@app.route('/', methods=["GET", "POST"])
def index():
    """Main page with instructions"""
    # Handle POST request
    if request.method == "POST":

        return redirect(request.form["username"])
    return render_template("index.html")

@app.route('/<username>', methods=["GET", "POST"])
def user(username):
    return render_template("answer_riddles.html")

@app.route('/<username>/<message>')
def send_message(username, message):
    return "{0}: {1}\n".format(username, message)

@app.route('/leader_board', methods=["GET", "POST"])
```

```
def leader_board():
    return render_template("leader_board.html")
```

```
if __name__ == '__main__':
    app.run(host=os.environ.get('IP', '0.0.0.0'), port=int(os.environ.get('PORT', 0)), debug=True)
    //////////////////////////////////////
```

User.txt file is appended to so will create a file if it does not exist. Is it writing data?

```
test_check_user_file_is_storing_username (__main__.test_run) ... FAIL
test_for_base_html (__main__.test_run) ... ok
test_for_index_html (__main__.test_run) ... ok
test_for_leader_board_html (__main__.test_run) ... ok
test_index (__main__.test_run) ... ok
test_leader_board (__main__.test_run) ... ok
test_message_can_be_sent (__main__.test_run) ... ok
test_riddle_page (__main__.test_run) ... ok
test_user_page (__main__.test_run) ... ok
```

```
FAIL: test_check_user_file_is_storing_username (__main__.test_run)
```

```
-----
Traceback (most recent call last):
```

```
  File "/home/ubuntu/workspace/test_riddle.py", line 56, in
test_check_user_file_is_storing_username
    self.assertGreater(len(lines), 0)
AssertionError: 0 not greater than 0
```

```
-----
Ran 9 tests in 0.054s
```

```
FAILED (failures=1)
```

```
*****
```

Add write_to_file("data/users.txt", request.form["username"] + "\n") to def index

```
functiontest_check_user_file_is_storing_username (__main__.test_run) ... ok
test_for_base_html (__main__.test_run) ... ok
test_for_index_html (__main__.test_run) ... ok
test_for_leader_board_html (__main__.test_run) ... ok
test_index (__main__.test_run) ... ok
test_leader_board (__main__.test_run) ... ok
test_message_can_be_sent (__main__.test_run) ... ok
test_riddle_page (__main__.test_run) ... ok
test_user_page (__main__.test_run) ... ok
```

```
-----
Ran 9 tests in 0.058s
```

```
OK
```

```
////////////////////////////////////
```

Revised code makes a test fail

```
@app.route('/<username>/<message>')
```

```
def send_message(username, message):
    """Create a new message and redirect back to the game page"""
    add_messages(username, message)
    return redirect(username)
test_check_user_file_is_storing_username (__main__.test_run) ... ok
test_for_base_html (__main__.test_run) ... ok
test_for_index_html (__main__.test_run) ... ok
test_for_leader_board_html (__main__.test_run) ... ok
test_index (__main__.test_run) ... ok
test_leader_board (__main__.test_run) ... ok
test_message_can_be_sent (__main__.test_run) ... FAIL
test_riddle_page (__main__.test_run) ... ok
test_user_page (__main__.test_run) ... ok
```

FAIL: test_message_can_be_sent (__main__.test_run)

Traceback (most recent call last):

```
  File "/home/ubuntu/workspace/test_riddle.py", line 34, in test_message_can_be_sent
    self.assertEqual(response.status_code, 200)
AssertionError: 302 != 200
```

Ran 9 tests in 0.056s

FAILED (failures=1)

*New code is redirecting messages so changing test code to self.assertEqual
(response.status_code, 302) passes*

```
test_check_user_file_is_storing_username (__main__.test_run) ... ok
test_for_base_html (__main__.test_run) ... ok
test_for_index_html (__main__.test_run) ... ok
test_for_leader_board_html (__main__.test_run) ... ok
test_index (__main__.test_run) ... ok
test_leader_board (__main__.test_run) ... ok
test_message_can_be_sent (__main__.test_run) ... ok
test_riddle_page (__main__.test_run) ... ok
test_user_page (__main__.test_run) ... ok
```

Ran 9 tests in 0.056s

OK

////////////////////////////////////

Test that the JSON data file loads. File should load.

ERROR: test_data_file_loads (__main__.test_run)

Traceback (most recent call last):


```
File "/home/ubuntu/workspace/test_riddle.py", line 60, in test_data_file_loads
    with open("data/riddles.json", "r") as json_data:
IOError: [Errno 2] No such file or directory: 'data/riddles.json'
```

Ran 10 tests in 0.059s

FAILED (errors=1)

Added file to the data folder and now test passes

```
test_check_user_file_is_storing_username (__main__.test_run) ... ok
test_data_file_loads (__main__.test_run) ... ok
test_for_base_html (__main__.test_run) ... ok
test_for_index_html (__main__.test_run) ... ok
test_for_leader_board_html (__main__.test_run) ... ok
test_index (__main__.test_run) ... ok
test_leader_board (__main__.test_run) ... ok
test_message_can_be_sent (__main__.test_run) ... ok
test_riddle_page (__main__.test_run) ... ok
test_user_page (__main__.test_run) ... ok
```

Ran 10 tests in 0.051s

OK

//

Riddle form should load when user name entered

```
test_check_user_file_is_storing_username (__main__.test_run) ... ok
test_data_file_loads (__main__.test_run) ... ok
test_for_base_html (__main__.test_run) ... ok
test_for_index_html (__main__.test_run) ... ok
test_for_leader_board_html (__main__.test_run) ... ok
test_index (__main__.test_run) ... ok
test_leader_board (__main__.test_run) ... ok
test_message_can_be_sent (__main__.test_run) ... ok
test_riddle_form (__main__.test_run) ... FAIL
test_riddle_page (__main__.test_run) ... ok
test_user_page (__main__.test_run) ... ok
```

=====

```
FAIL: test_riddle_form (__main__.test_run)
```

Traceback (most recent call last):

```
File "/home/ubuntu/workspace/test_riddle.py", line 67, in test_riddle_form
    self.assertIn("Write guess here", response.data)
AssertionError: 'Write guess here' not found
```

Ran 11 tests in 0.059s

FAILED (failures=1)

Added form contents to html and post method in run file:

```
@app.route('/<username>', methods=["GET", "POST"])
def user(username):
    """Display game messages for the user"""
    #set up list to hold the data once file opened
    data = []
    with open("data/riddles.json", "r") as json_data:
        data = json.load(json_data)
    #set variables used in the game
    riddle_counter = 0
    total_riddles = len(data)
    numRight = 0
    numWrong = 0
    score = 0
    if request.method == "POST":
        write_to_file("data/users.txt", username + "\n")
        return render_template("answer_riddles.html", username=username, riddles_data=data,
            riddle_counter=riddle_counter)

<div class = "row riddle-question form-group">
    <div class = "col-sm-12">
        <form method="POST" class = "form-group text-center">
            <label for="riddle">Guess the riddle below. Click button to guess</label>
            <h3><textarea class="form-control text-center" rows="3"
id="riddle">{{ riddles_data[riddle_counter]["riddle"] }}</textarea></h3>
            <div class = "row">
                <div class = "col-sm-4">
                    <label for="riddle_counter">Riddle Counter:</label>
                    <input type="text" id="riddle_counter" placeholder = "Riddle Counter"
maxlength="4" size="4" name="riddle_counter" readonly value = "{{ riddle_counter }}">
                </div>
                <div class = "col-sm-5"><input type="text" class = "form-control" placeholder =
"Write guess here, then click button or hit enter" id="message" name="message"></div>
                <div class = "col-sm-3"><button class = "btn btn-info">Guess the Riddle!
</button></div>
            </div>
        </div>
    </div>
```

```
test_check_user_file_is_storing_username (__main__.test_run) ... ok
test_data_file_loads (__main__.test_run) ... ok
test_for_base_html (__main__.test_run) ... ok
test_for_index_html (__main__.test_run) ... ok
test_for_leader_board_html (__main__.test_run) ... ok
test_index (__main__.test_run) ... ok
test_leader_board (__main__.test_run) ... ok
test_message_can_be_sent (__main__.test_run) ... ok
test_riddle_form (__main__.test_run) ... ok
test_riddle_page (__main__.test_run) ... ok
test_user_page (__main__.test_run) ... ok
```

Ran 11 tests in 0.124s

OK

//

Test that the riddle counter works:(as with riddle form above ; if loaded will pass)

test_check_user_file_is_storing_username (__main__.test_run) ... ok
test_data_file_loads (__main__.test_run) ... ok
test_for_base_html (__main__.test_run) ... ok
test_for_index_html (__main__.test_run) ... ok
test_for_leader_board_html (__main__.test_run) ... ok
test_index (__main__.test_run) ... ok
test_leader_board (__main__.test_run) ... ok
test_message_can_be_sent (__main__.test_run) ... ok
test_riddle_counter (__main__.test_run) ... ok
test_riddle_form (__main__.test_run) ... ok
test_riddle_page (__main__.test_run) ... ok
test_user_page (__main__.test_run) ... ok

Ran 12 tests in 0.063s

OK

//

As each user answers a riddle the scores are written to the text file:

test_check_data_is_in_scores_file (__main__.test_run) ... ERROR
test_check_user_file_is_storing_username (__main__.test_run) ... ok
test_data_file_loads (__main__.test_run) ... ok
test_for_base_html (__main__.test_run) ... ok
test_for_index_html (__main__.test_run) ... ok
test_for_leader_board_html (__main__.test_run) ... ok
test_index (__main__.test_run) ... ok
test_leader_board (__main__.test_run) ... ok
test_message_can_be_sent (__main__.test_run) ... ok
test_riddle_counter (__main__.test_run) ... ok
test_riddle_form (__main__.test_run) ... ok
test_riddle_page (__main__.test_run) ... ok
test_user_page (__main__.test_run) ... ok

=====

ERROR: test_check_data_is_in_scores_file (__main__.test_run)

Traceback (most recent call last):

File "/home/ubuntu/workspace/test_riddle.py", line 73, in test_check_data_is_in_scores_file
score = run.add_scores("1")
AttributeError: 'module' object has no attribute 'add_scores'

Ran 13 tests in 0.072s

FAILED (errors=1)

Added the functions that append the data: (def write to file and def add_scores());

```
test_check_data_is_in_scores_file (__main__.test_run) ... FAIL
test_check_user_file_is_storing_username (__main__.test_run) ... ok
test_data_file_loads (__main__.test_run) ... ok
test_for_base_html (__main__.test_run) ... ok
test_for_index_html (__main__.test_run) ... ok
test_for_leader_board_html (__main__.test_run) ... ok
test_index (__main__.test_run) ... ok
test_leader_board (__main__.test_run) ... ok
test_message_can_be_sent (__main__.test_run) ... ok
test_riddle_counter (__main__.test_run) ... ok
test_riddle_form (__main__.test_run) ... ok
test_riddle_page (__main__.test_run) ... ok
test_user_page (__main__.test_run) ... ok
```

FAIL: test_check_data_is_in_scores_file (__main__.test_run)

Traceback (most recent call last):

```
File "/home/ubuntu/workspace/test_riddle.py", line 75, in test_check_data_is_in_scores_file
    self.assertIn("1", (str(score)))
```

AssertionError: '1' not found in "['Bobby']"

Ran 13 tests in 0.067s

FAILED (failures=1)

Data not written yet. Once data added passes:(if this file is empty; this test will fail);

```
test_check_data_is_in_scores_file (__main__.test_run) ... ok
test_check_user_file_is_storing_username (__main__.test_run) ... ok
test_data_file_loads (__main__.test_run) ... ok
test_for_base_html (__main__.test_run) ... ok
test_for_index_html (__main__.test_run) ... ok
test_for_leader_board_html (__main__.test_run) ... ok
test_index (__main__.test_run) ... ok
test_leader_board (__main__.test_run) ... ok
test_message_can_be_sent (__main__.test_run) ... ok
test_riddle_counter (__main__.test_run) ... ok
test_riddle_form (__main__.test_run) ... ok
test_riddle_page (__main__.test_run) ... ok
test_user_page (__main__.test_run) ... ok
```

Ran 13 tests in 0.067s

OK

////////////////////////////////////

As each user answers a riddle the scores are also written to the tot_scores.text file:

```
test_check_data_is_in_scores_file (__main__.test_run) ... ok
```

```
test_check_data_is_in_tot_scores_file (__main__.test_run) ... FAIL
test_check_user_file_is_storing_username (__main__.test_run) ... ok
test_data_file_loads (__main__.test_run) ... ok
test_for_base_html (__main__.test_run) ... ok
test_for_index_html (__main__.test_run) ... ok
test_for_leader_board_html (__main__.test_run) ... ok
test_index (__main__.test_run) ... ok
test_leader_board (__main__.test_run) ... ok
test_message_can_be_sent (__main__.test_run) ... ok
test_riddle_counter (__main__.test_run) ... ok
test_riddle_form (__main__.test_run) ... ok
test_riddle_page (__main__.test_run) ... ok
test_user_page (__main__.test_run) ... ok
```

=====

FAIL: test_check_data_is_in_tot_scores_file (__main__.test_run)

Traceback (most recent call last):

File "/home/ubuntu/workspace/test_riddle.py", line 79, in test_check_data_is_in_tot_scores_file
self.assertIn("I", (str(score)))
AssertionError: 'I' not found in '[]'

Ran 14 tests in 0.072s

FAILED (failures=1)

After function wtritten, data needed to be added to pass this test:

```
test_check_data_is_in_scores_file (__main__.test_run) ... ok
test_check_data_is_in_tot_scores_file (__main__.test_run) ... ok
test_check_user_file_is_storing_username (__main__.test_run) ... ok
test_data_file_loads (__main__.test_run) ... ok
test_for_base_html (__main__.test_run) ... ok
test_for_index_html (__main__.test_run) ... ok
test_for_leader_board_html (__main__.test_run) ... FAIL
test_index (__main__.test_run) ... ok
test_leader_board (__main__.test_run) ... ok
test_message_can_be_sent (__main__.test_run) ... ok
test_riddle_counter (__main__.test_run) ... ok
test_riddle_form (__main__.test_run) ... ok
test_riddle_page (__main__.test_run) ... ok
test_user_page (__main__.test_run) ... ok
```

This now passes but the test_for_leader_board_html now fails

////////////////////////////////////

When leaderboard is rendered it is likely to contain a vowel in a username and thus verifys that the leaderboard html page is rendered and that the leaderboard contains data.(This fails if the vowel a is not present).

```
test_check_data_is_in_leaderBoard (__main__.test_run) ... ok
test_check_data_is_in_scores_file (__main__.test_run) ... ok
test_check_data_is_in_tot_scores_file (__main__.test_run) ... ok
test_check_user_file_is_storing_username (__main__.test_run) ... ok
test_data_file_loads (__main__.test_run) ... ok
test_for_base_html (__main__.test_run) ... ok
test_for_index_html (__main__.test_run) ... ok
test_index (__main__.test_run) ... ok
test_leader_board (__main__.test_run) ... ok
test_message_can_be_sent (__main__.test_run) ... ok
test_riddle_counter (__main__.test_run) ... ok
test_riddle_form (__main__.test_run) ... ok
test_riddle_page (__main__.test_run) ... ok
test_user_page (__main__.test_run) ... ok
```

Ran 14 tests in 0.056s

OK

////////////////////////////////////

Ensure that riddles have the correct answers

```
test_check_data_is_in_leaderBoard (__main__.test_run) ... ok
test_check_data_is_in_scores_file (__main__.test_run) ... ok
test_check_data_is_in_tot_scores_file (__main__.test_run) ... ok
test_check_user_file_is_storing_username (__main__.test_run) ... ok
test_data_file_loads (__main__.test_run) ... ok
test_for_base_html (__main__.test_run) ... ok
test_for_index_html (__main__.test_run) ... ok
test_index (__main__.test_run) ... ok
test_leader_board (__main__.test_run) ... ok
test_message_can_be_sent (__main__.test_run) ... ok
test_riddle_counter (__main__.test_run) ... ok
test_riddle_form (__main__.test_run) ... ok
test_riddle_has_correct_answer (__main__.test_run) ... FAIL
test_riddle_page (__main__.test_run) ... ok
test_user_page (__main__.test_run) ... ok
```

=====

FAIL: test_riddle_has_correct_answer (__main__.test_run)

Traceback (most recent call last):

```
File "/home/ubuntu/workspace/test_riddle.py", line 93, in test_riddle_has_correct_answer
    self.assertEqual(('Fill in the missing number: Give the answer as a number (for example 24): 46
65 84 ?? 28 ', '56'), riddle)
AssertionError: Tuples differ: ('Fill in the missing number: ... != (u'Fill in the missing number:...
```

First differing element 1:

```
56
2
```

Once the correct answer “56” is written in the test passes

```
test_check_data_is_in_leaderBoard (__main__.test_run) ... ok
test_check_data_is_in_scores_file (__main__.test_run) ... ok
test_check_data_is_in_tot_scores_file (__main__.test_run) ... ok
test_check_user_file_is_storing_username (__main__.test_run) ... ok
test_data_file_loads (__main__.test_run) ... ok
test_for_base_html (__main__.test_run) ... ok
test_for_index_html (__main__.test_run) ... ok
test_index (__main__.test_run) ... ok
test_leader_board (__main__.test_run) ... ok
test_message_can_be_sent (__main__.test_run) ... ok
test_riddle_counter (__main__.test_run) ... ok
test_riddle_form (__main__.test_run) ... ok
test_riddle_has_correct_answer (__main__.test_run) ... ok
test_riddle_page (__main__.test_run) ... ok
test_user_page (__main__.test_run) ... ok
```

Ran 15 tests in 0.121s

OK

////////////////////////////////////

