pytest (tentative)

simple check50-inspired code testing utility

Self-introduction

Name:	Nathanael Hananto Putro
Location of origin:	Jakarta, Indonesia
Current location:	Melbourne, Australia
Current occupation:	Undergraduate at the University of Melbourne

How to use

1. Install `pytest`
2. Make a YAML file that specifies tests to run
3. Run the tests in the YAML file using `pytest`
4. Get your test results

Step 1: Installing `pytest`

Requirements

- PyYAML
- termcolor

Can be quickly installed by running:

`pip install [filename].zip`
(if installing from a .zip file)

`pip install .`
(if installing from inside extracted pytest folder)

Step 2: Making a YAML tests file

Supported Commands

`description`	describes the test; printed in test results if available
`run`*	CLI command to run
`stdin`	what keyboard input to simulate
`stdout`*	what `run` should output to stdout
`exit`	what exit code `run` should exit with (defaults to `0`)
`timeout`	how many seconds `run` is allowed to execute (defaults to 2)
`shell`	enable shell functions for `run` execution (defaults to `False`)

* required

Sample

```
pytest:
␣␣tests:
␣␣␣␣␣hello:
␣␣␣␣␣␣␣␣description: Correctly prints “hello world”
␣␣␣␣␣␣␣␣run: ./hello
␣␣␣␣␣␣␣␣stdout: Hello, world!\n

␣␣␣␣␣multiline hello:
␣␣␣␣␣␣␣␣description: Correctly prints multiline hello world
␣␣␣␣␣␣␣␣run: ./hello_multiline
␣␣␣␣␣␣␣␣stdout: |
␣␣␣␣␣␣␣␣␣␣Hello,
␣␣␣␣␣␣␣␣␣␣world!
```

␣ indicates a space

Step 3: Running tests

Usage

`pytest`	`-h, --help`	shows this information in your terminal
	`-f [FILENAME], --filename [FILENAME]`	specify name of a test file (defaults to tests.yaml)
	`-d, --dev`	run `pytest` in development mode (implies --verbose)
	`-l, --log`	display tests' raw execution log (`stdin`, `stdout`, `exitcode`, etc.)
	`-t [TARGET], --target [TARGET]`	target a specific test to run
	`-v, --verbose`	display full tracebacks of any errors (also implies --log)

Step 4: Reading your test results

Legend

This test has passed!

- Program output expected output and exited with expected exit code.

This test has not been successfully executed.

- Program timed out.
- Program raised an error during execution.

This test has failed.

- Program didn’t exit with expected exit code.
- Program didn’t output the expected output.

Acknowledgements

- check50
- r/learnpython
- Stack Overflow