djoseph-png / py-cryptocurrency

Code Pull requests Actions Projects Wiki Security Insights Settings

0 stars 1.4k forks 0 watching Branches Activity
Tags

Public repository · Forked from mate-academy/py-cryptocurrency

master 1 Branch 0 Tags Go to file

This branch is up to date with mate-academy/py-cryptocurrency:master .

Abnormaltype Merge pull request mate-academy#439 from mate-academy/remove-redundar

.github/workflows	Update test.yml	
app	removed redundant code	
tests	Add tests for tests and auto-approve	3 years ago
.flake8	Update .flake8	3 years ago
.gitignore	Initial commit	3 years ago
README.md	Fix tests with xdist	3 years ago
requirements.txt	Update requirements.txt	3 years ago

Local Codespaces

Clone

HTTPS SSH GitHub CLI

https://github.com/djoseph-png/py-cryptocurrenc

Clone using the web URL.

Open with GitHub Desktop

Download ZIP

README

Cryptocurrency

- Read the guideline before start

You have some amount of cryptocurrency "Matecoin" in your online wallet. You also bought a program (function `get_exchange_rate_prediction`) that can predict the exchange rate of your cryptocurrency for the next day.

Inside `app/test_main.py`, write tests for `cryptocurrency_action` function. This function takes `current_rate` - current exchange rate of cryptocurrency. This function should return:

- `"Buy more cryptocurrency"`, if predicted exchange rate is more than 5% higher from the current.
- `"Sell all your cryptocurrency"`, if predicted exchange rate is more than 5% lower from the current.
- `"Do nothing"`, if difference is not that much.

Mock `get_exchange_rate_prediction` function.

Run `pytest app/` to check if function pass your tests.

Run `pytest --numprocesses=auto tests/` to check if your tests cover all boundary conditions and pass task tests.

Releases

No releases published
Create a new release

Packages

No packages published
Publish your first package

Languages

- **Python** 100.0%

Suggested workflows

Based on your tech stack

Publish Python Package Configure

Publish a Python Package to PyPI on release.

Pylint Configure

Lint a Python application with pylint.

SLSA Generic generator Configure

Generate SLSA3 provenance for your existing release workflows

More workflows Dismiss suggestions