

Steuerung von Tests

Quelle: <https://docs.pytest.org/en/7.1.x/how-to/skipping.html>

Konstante im Programm

Diese Beispiele basieren auf einer Konstante im Programm.

Bedingtes Ausführen von Tests

```
import pytest

@pytest.mark.skipif(primes.TESTSTEP == 4, reason='Test not relevant for this
step')
def test_sieve_1a(monkeypatch, capsys):
    ...
```



Bei Verwendung im **autograding** von GitHub Classroom wird dieser Test als erfolgreich gewertet.

Erwarteter Fail des Tests

```
import python
@pytest.mark.xfail(primes.TESTSTEP!=4, reason='test only works for step 4')
def test_sieve_4(monkeypatch, capsys):
    ...
```



Bei Verwendung im **autograding** von GitHub Classroom wird dieser Test als gescheitert gewertet.

From:

<https://wiki.bzz.ch/> - **BZZ - Modulwiki**

Permanent link:

<https://wiki.bzz.ch/howto/pytest/steuerung>

Last update: **2024/03/28 14:07**

