



Testing my twttr

Problem set



در فایلی به اسم `twttr.py` سوال Setting up my twttr.py از پرابلمست هفته‌ی دوم را مجدداً به نحوی پیاده سازی کنید که تابع `shorten` یک `str` ورودی بگیرد و تمام حروف صدادار آن (A, E, I, O, U) چه حروف کوچک و چه حروف بزرگ را حذف کند.

```
def main():
    ...
def shorten(word):
    ...
if __name__ == "__main__":
    main()
```

سپس در فایلی به نام `test_twttr.py`, یک و یا بیشتر از یک تابع پیاده سازی کنید که با استفاده از آنها درستی `shorten` را تست کنید. اسم هر کدام از آنها باید با `_test` شروع شود تا بتوانید کد زیر را اجرا کنید.

```
pytest test_twttr.py
```

نکته ▼

- استفاده از فایل `twttr` را فراموش نکنید.

```
import twttr
```

یا

```
from twttr import shorten
```

با این کار می‌توانید `shorten` را در تست‌های خود فراخوانی کنید.

- همچنین دقت کنید که از `print` فقط در تابع `main` می‌توانید در این سناریو استفاده کنید و در دیگر موارد باید از `return` استفاده کنید.

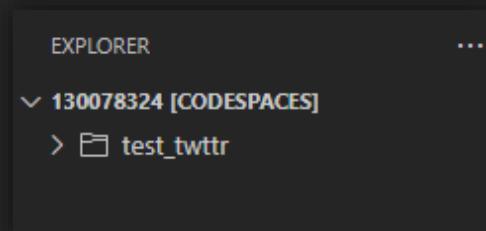


وارد code.cs50.io شوید، سپس بر روی پنجرهی Terminal کلیک کنید. توجه داشته باشید که دستور پنجرهی Terminal شما باید به صورت زیر باشد:

\$

سپس کد زیر را اجرا کنید تا یک پوشه به نام `test_twttr` در codespace ایجاد شود:

```
$ mkdir test_twttr
```

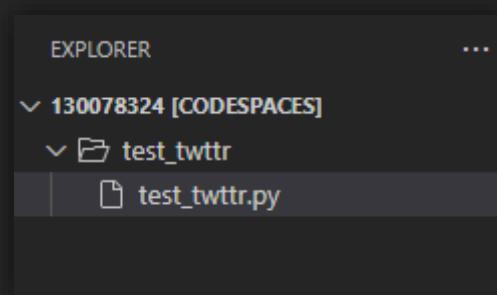


سپس `test_twtr` را اجرا کنید تا به پوشه‌ی `test_twtr` منتقل شوید. اکنون شما باید آدرس Terminal را در `test_twtr/` مشاهده کنید:

```
$ cd test_twtr  
test_twtr/ $
```

حالا می‌توانید دستور `code test_twtr.py` ایجاد شود:

```
test_twtr/ $ code test_twtr.py
```



اکنون می‌توانید در این فایل برنامه‌ی خود را بنویسید.

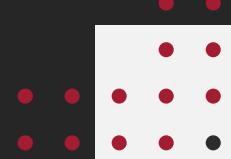
برای استفاده از تست‌های خود، `pytest test_twtr.py` را اجرا کنید و دقت کنید که حتماً یک کپی از فایل در همان پوشه داشته باشید و برای بررسی بیشتر تست‌ها از ورژن‌های نادرست `twtr.py` هم استفاده کنید:

- مطمئن شوید که نسخه درستی از `twtr.py` را دارید. تست‌های خود را با `pytest test_twtr.py` اجرا کنید. باید نشان دهد که تمام تست‌های شما درست هستند.
- ورژن درست `twtr.py` را طوری تغییر دهید که برنامه شما باگ داشته باشد و نتایج اشتباهی رخ بدهد. برای مثال برنامه شما به اشتباه فقط حروف صدادار کوچک را حذف کند. اکنون، تست‌های خود را با `pytest test_twtr.py` اجرا کنید. باید نشان دهد که حداقل یکی از تست‌های شما مورد قبول نبوده است.

- ❖ شما می‌توانید از آدرس زیر برای بررسی کردن کد خود استفاده کنید. CS50 از این برنامه برای آزمایش کد شما استفاده می‌کند. از این دستور استفاده کنید تا کدهایتان را امتحان کنید.

```
$ check50 cs50/problems/2022/python/tests/twttr
```

لبخند های سبز به این معنی هستند که برنامه‌ی شما در تست قبول شده و اخم های قرمز یعنی برنامه‌ی شما دارای ایراد هست. با مراجعه به URL check50 می‌توانید خروجی مورد انتظار و خروجی برنامه‌ی خود را بررسی کنید.



با اجرا کردن صورت زیر در Terminal پاسخ خود را ارسال کنید.

```
$ submit50 cs50/problems/2022/python/tests/twttr
```

CS50x Iran

Harvard's Computer Science 50x Iran

