



CS50 P-Shirt

Problem set





دانشجویان در دانشگاه هاروارد بعد از تمام کردن دوره CS50، به طور سنتی تی‌شرتی برای خودشان که عبارت I took CS50 روی آن چاپ شده را دریافت می‌کنند. نیازی به خرید آنلاین نیست، اما آیا دوست دارید آن را به صورت مجازی امتحان کنید؟

در یک فایل به نام `shirt.py`، برنامه ای اجرا کنید که دقیقا دو آرگومان زیر را هنگام اجرا در ترمینال ورودی بگیرد:

- در `sys.argv[1]`، نام (یا مسیر) یک فایل `JPEG` یا `PNG` برای خواندن (یعنی باز کردن) به عنوان ورودی
- در `sys.argv[2]`، نام (یا مسیر) یک فایل `JPEG` یا `PNG` برای نوشتن (یعنی ذخیره) به عنوان خروجی.

این برنامه باید تصویر `shirt.png` را که پس زمینه شفاف (transparent) دارد، بعد از تغییر اندازه تصویر ورودی به اندازه تصویر `shirt.png`، روی تصویر ورودی قرار دهد و نتیجه را به عنوان خروجی ذخیره کند.

تصویر ورودی را با استفاده از `Image.open` با آدرس زیر باز کنید

<https://pillow.readthedocs.io/en/stable/reference/Image.html#PIL.Image.open>

به میزان دلخواه اندازه آنرا تغییر دهید و آن را با `ImageOps.fit` ([اطلاعات بیشتر](#)) برش (`crop`) دهید

با استفاده از مقادیر پیش فرض برای `method`، `bleed` و `centering`، تیشرت را بر روی تصویر اولیه با استفاده از `Image.paste` ([اطلاعات بیشتر](#)) جایگذاری کنید .

سپس نتیجه را با `Image.save` ([اطلاعات بیشتر](#)) ذخیره کنید.

برنامه باید در شرایط زیر با دستور `sys.exit` متوقف شود

- اگر کاربر دقیقا دو آرگومان `command-line` مشخص نکند.
- اگر پسوند فایل های ورودی و خروجی `.jpg`، `.jpeg` یا `.png` نباشد. بزرگی و کوچکی حروف اهمیتی ندارد.
- اگر پسوند نام فایل ورودی با پسوند نام فایل خروجی برابر نباشد.
- اگر فایل ورودی داده شده وجود نداشته باشد.

فرض کنید که ورودی یک عکس شخصی است که در یک حالت مناسب قرار دارد، مانند این عکس هایی که در بخش دمو میبینید، به طوری که در صورت تغییر اندازه و برش عکس، تی شرت به طور کامل برای او مناسب بنظر برسد.

اگر می خواهید برنامه را روی عکس خود اجرا کنید، ابتدا عکس را به `file explorer` برنامه `VSCode` بکشید (در همان پوشه ای که فایل `shirt.py` قرار دارد). نیازی به ارسال عکس، همراه با کد نیست. اما اگر می خواهید، به هر روش که دوست دارید، می توانید عکسی از خود با تی شرت مجازی با یکی از شبکه های اجتماعی `CS50` به اشتراک بگذارید. (اما اجباری نیست!)

▼ نکته

- توجه کنید که می‌توانید با استفاده از `os.path.splitext` ([اطلاعات بیشتر](#)) نوع فایل را تعیین کنید: توجه کنید که `open` ممکن است یک خطای `FileNotFoundError` ([اطلاعات بیشتر](#)) ایجاد کند.

- توجه داشته باشید که `Pillow` شامل تعدادی کلاس و متد است، می‌توانید آنها را در pypi.org/project/Pillow مشاهده کنید (شاید به شما کمک کند)

- شما می‌توانید `Pillow` را با استفاده از دستور زیر نصب کنید:

```
pip install Pillow
```

- شما می‌توانید یک تصویر (مانند `shirt.png`) را با کدی مانند کد زیر باز کنید:

```
shirt = Image.open("shirt.png")
```

شما می‌توانید با استفاده از کدی مانند زیر، عرض و طول یک تصویر (مثلاً `shirt.png`) را بدست آورید:

```
size = shirt.size
```

همچنین می‌توانید تصویر را بر روی تصویر دیگری با کدی مانند زیر قرار دهید:

```
photo.paste(shirt, shirt)
```

در این کد، اولین `"shirt"` تصویری است که بر روی تصویر دیگر (به عنوان مثال `"photo"`) قرار داده می‌شود و دومین `"shirt"` یک "ماسک" است که نشان می‌دهد کدام پیکسل‌ها در تصویر `"photo"` باید به‌روزرسانی شوند.

▪ توجه داشته باشید که می‌توانید با اجرای دستور زیر، یک تصویر (مانند `shirt.png`) را در `VSCode` باز کنید:

```
code shirt.png
```

یا می‌توانید با دوبار کلیک بر روی آیکون تصویر (مثلاً `shirt.png`) در اکسپلورر فایل `VSCode`، تصویر را باز کنید.


```
$ python shirt.py
Too few command-line arguments
$ python shirt.py foo
Too few command-line arguments
$ python shirt.py foo bar
Invalid input
$ python shirt.py foo bar baz
Too many command-line arguments
$ python shirt.py before.jpg after.png
Input and output have different extensions
$ python shirt.py before.jpg after.jpg
$ python shirt.py before.png after.png
$
```

قبل:



بعد:



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 **shirt** در **codespace** ایجاد شود:

```
$ mkdir shirt
```

سپس **cd shirt** را اجرا کنید تا به پوشه‌ی **shirt** منتقل شوید. اکنون شما باید آدرس **\$ /shirt** را در **Terminal** مشاهده کنید:

```
$ cd shirt  
shirt/ $
```

حالا می‌توانید دستور `code shirt.py` را اجرا کنید تا فایل با نام `shirt.py` ایجاد شود:

```
shirt/ $ code shirt.py
```

برای این که فایل `shirt.py` را بسازید تا برنامه تان را در آن بنویسید. همچنین دستور زیر را اجرا کنید: ❖ برای اینکه فایل `shirt.png` دانلود شود .

```
wget https://cs50.harvard.edu/python/2022/psets/6/shirt/shirt.png
```

برای این که `shirt.png` را دانلود کنید. سپس کد زیر را اجرا کنید:

```
wget https://cs50.harvard.edu/python/2022/psets/6/shirt/muppets.zip
```

برای این که `muppets.zip` را دانلود کنید. و در آخر دستور زیر را اجرا کنید:

```
unzip muppets.zip
```

❖ برای این که آن را از حالت فشرده خارج کنید.

در این قسمت توضیح داده شده که چطور خودتون به عنوان یک کاربر برنامه نوشته شده را تست کنید

- برنامه خود را با `python shirt.py` اجرا کنید. برنامه شما باید با دستور `sys.exit` خارج شده و یک پیام خطا را نمایش دهد:

```
Too few command-line arguments
```

- مطمئن شوید که [muppets.zip](#) را دانلود و عکس‌های آن را با استفاده از دستور `unzip muppets.zip` از حالت فشرده خارج کردید. برنامه خود را با `python shirt.py before1.jpg before2.jpg before3.jpg` اجرا کنید. خروجی برنامه شما باید به این صورت باشد:

```
Too many command-line arguments
```

- برنامه خود را با `python shirt.py before1.jpg invalid_format.bmp` اجرا کنید. برنامه شما باید با استفاده از `sys.exit` متوقف شود و یک خطا با پیام زیر نشان دهد:

```
Invalid output
```

- برنامه خود را با `python shirt.py before1.jpg after1.png` اجرا کنید. برنامه شما باید با دستور `sys.exit` خارج شده و یک پیام خطا را نمایش دهد:

```
Input and output have different extensions
```

- برنامه خود را با `python shirt.py non_existent_image.jpg after1.jpg` اجرا کنید. برنامه شما باید با دستور `sys.exit` خارج شده و یک پیام خطا را نمایش دهد:

```
Input does not exist
```

- برنامه خود را با `python shirt.py before1.jpg after1.jpg` اجرا کنید. با توجه به این که انتظار می‌رود شما [muppets.zip](#) را دانلود و از حالت فشرده خارج کرده اید. برنامه شما باید خروجی زیر را نمایش دهد:



❖ 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/shirt
```

لبخند های سبز :) به این معنی هستند که برنامه‌ی شما در تست **قبول شده** و اخم های قرمز :) یعنی برنامه‌ی شما **دارای ایراد** هست، همچنین پیام های زرد ! بدین معنی است که برای اینکه **check50** بتواند این بخش برنامه شما را بررسی کند باید ابتدا ارورهایی که با اخم های قرمز :) جلوتر آمده اند را برطرف کنید.

. برای دیدن جزییات بیشتر درباره این تست که توسط **check50** صورت گرفت، وارد لینکی که به شما خروجی داده است شوید.

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

```
$ submit50 cs50/problems/2022/python/shirt
```

توجه کنید هرچند بار که لازم باشد میتوانید کد خود را اصلاح و ارسال کنید ، نمره آخرین پاسخ شما در نظر گرفته میشود .

CS50x Iran

Harvard's Computer Science 50x Iran

