



Felipe's Taqueria

Problem set



پاییز و زمستان ۱۴۰۴



یکی از محبوب ترین مکان‌ها برای خوردن در میدان هاروارد، Felipe's Taqueria است که طبق دستور زیر فهرستی از خوراکی‌ها را ارائه می‌دهد که key برابر با اسم غذا و value آن، قیمت خوراکی به دلار است که در زیر آورده شده است:

```
{  
    "Baja Taco": 4.00,  
    "Burrito": 7.50,  
    "Bowl": 8.50,  
    "Nachos": 11.00,  
    "Quesadilla": 8.50,  
    "Super Burrito": 8.50,  
    "Super Quesadilla": 9.50,  
    "Taco": 3.00,  
    "Tortilla Salad": 8.00  
}
```



در فایلی به نام `taqueria.py`، برنامه‌ای بنویسید که از کاربر بخواهد تا سفارش ثبت کند و در هر خط یک درخواست سفارش را از کاربر دریافت کند.

برنامه باید پشت سر هم از کاربر نام غذا‌ها را دریافت کند به صورت `Item: Item` و بعد از وارد کردن نام هر غذا هزینه کل غذا‌هایی که وارد کرد را با دو رقم اعشار و با پیشوند `$` به صورت `Total: $00.00` نمایش دهد. برنامه باید تا زمانی ادامه پیدا کند که کاربر با زدن کلید ترکیبی `Ctrl + d` برنامه را متوقف کند. برنامه شما باید به حروف کوچک و بزرگ حساس باشد، اگر غذا وارد شده کاربر در `dict` وجود ندارد، آن را نادیده بگیرید. فرض کنید که هر آیتم titlecased است.

نکته ▼

- دقت داشته باشید شما با زدن کلید ترکیبی `Ctrl + d` با خطای `EOFError` مواجه می‌شوید. با نوشتن کدی مانند زیر می‌توانید این خطا را مدیریت کنید:

```
try:  
    item = input("Item: ")  
except EOFError:  
    ...
```

- ممکن است بخواهید یک خط جدید چاپ کنید تا نشانگر کاربر (و پیام بعدی) روی همان خطی که درخواست برنامه شماست باقی نماند. کد خود را به صورت زیر تغییر دهید تا بعد از پایان برنامه، نشانگر به خط بعد منتقل شود:

```
try:  
    item = input("Item: ")  
except EOFError:  
    exit("")
```



نکته ▼

- به یاد داشته باشید که dictionary توابع مختلفی دارد. توضیحات بیشتر را در اینجا می‌توانید ببینید:

docs.python.org/3/library/stdtypes.html#mapping-types-dict

- یکی از آن توابع get است که شما با نوشتن `d[key]` می‌توانید به آیتم‌های درون یک دیکشنری دسترسی داشته باشید با فرض این که `d` یک دیکشنری و `key` کلید آن آیتم مدنظر است. برای مثال به کد زیر توجه کنید:

```
if key in d:  
    item = d[key]
```

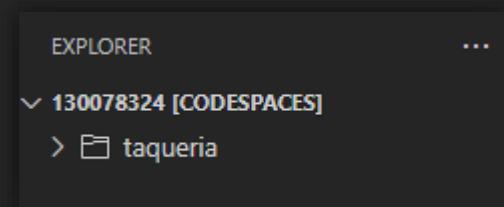


وارد code.cs50.io شوید، سپس بر روی پنجره Terminal کلیک کنید. توجه داشته باشید که دستور پنجره Terminal شما باید به صورت زیر باشد:

\$

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

```
$ mkdir taqueria
```

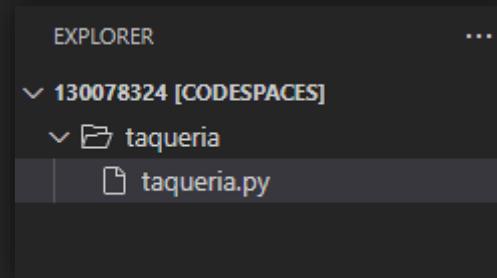


سپس `taqueria` را اجرا کنید تا به پوشه‌ی `taqueria` منتقل شوید. اکنون شما باید آدرس `$/taqueria` را در Terminal مشاهده کنید:

```
$ cd taqueria  
taqueria/ $
```

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

```
taqueria/ $ code taqueria.py
```



اکنون می‌توانید در این فایل برنامه‌ی خود را بنویسید.

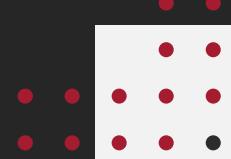
اکنون می‌خواهیم برنامه‌ی نوشته شده‌ی خود را تست کنیم:

- دستور `python taqueria.py` را اجرا کنید. Taco را تایپ کنید و کلید Enter را فشار دهید. دوباره Taco را تایپ کنید و کلید Enter را فشار دهید. خروجی شما برابر می‌شود با: Total: \$6.00
- دستور `python taqueria.py` را اجرا کنید. Baja Taco را تایپ کنید و کلید Enter را فشار دهید. دوباره Total: \$12.00 را تایپ کنید و کلید Enter را فشار دهید. خروجی شما برابر می‌شود با: Tortilla Salad
- دستور `python taqueria.py` را اجرا کنید. Burger را تایپ کنید و کلید Enter را فشار دهید. برنامه باید دوباره از کاربر درخواست ورودی کند.
- حتما سعی کنید سایر غذاها را هم تست کنید و حالت حروف را در ورودی خود تغییر دهید. برنامه شما باید بدون توجه به حالت حروف ورودی، به طور مطلوب عمل کند.

- ❖ 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/taqueria
```

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/taqueria
```

CS50x Iran

Harvard's Computer Science 50x Iran

