

مکتب شریف

اولین بوتکمپ آموزشی - استخدامی ایران



سری ششم

مکتب ۶۰





Python

سوال ۱:

- می‌خواهید برنامه‌ای بنویسید که فایل‌ها را خط به خط نمایش دهد. لیست فایل‌های شما در ابتدا باید قابل مشاهده باشد (چاپ شده باشد)، پس از نمایش نمایش خط، برنامه منتظر میماند تا `user` به آن ورودی دهد، `user` میتواند:
 - `Enter` را برای نمایش خط بعدی بفشارد.
 - اگر `user + Enter + n` را بفشارد، برنامه تمام فایل کنکونی را فراموش میکند و به فایل بعدی میرود.
 - اگر `user`، هر دکمه دیگری به جز `n` را علاوه `Enter` بفشارد، برنامه خط بعدی را نمایش میدهد.
- بخش اول به شکل زیر نوشته شده است. کار شما نوشتن دو قسمت دیگر برنامه است. `readline` در طول ساختار برنامه، لیستی از فایل‌ها را برای خواندن برنامه برمیگرداند. پس از اتمام آخرین فایل برنامه متوقف میشود. `User` همچنین میتواند برای برنامه استثنا قائل شود و با فراخوانی `(SkipThisFile)` به برنامه پیام دهد که از کل فایل کنونی عبور کند و تنها یک مقدار ساختگی برای عبور از فایل برگرداند.

نکته: بارم این سوال ۲۰ نمره میباشد.

```
class SkipThisFile(Exception):
    "The the generator to jump to the next file in list."
    pass

def read_lines(*files):
    """this is the generator to be written

    >>> list(read_lines('exercises.rst'))[:2]
    ['=====', 'Advanced Python -- excercises']
    """

    for file in files:
        yield 'dummy line'

def display_files(*files):
    source = read_lines(*files)
    for line in source:
        print(line, end='')
        inp = input()
        if inp == 'n':
            print('NEXT')
            source.throw(SkipThisFile) # return value is ignored
```

سوال ۲:

- به مقدار زیر 8000 از یک لیست دلخواه 2000 تا اضافه کنید:

نمونه لیست:

List = [1000, 500, 600, 700, 5000, 90000, 175000]

نکته: بارم این سوال ۵ نمره میباشد.



سوال ۳:

- اعداد منفی یک لیست دلخواه را اگر منفی هستند، به مثبت تبدیل کنید. سپس تنها المان هایی از لیست را که از منفی به مثبت تبدیل شدند، به درون لیست جدیدی انتقال دهید.
نمونه لیست:

List = [-1000, 500, -600, 700, 5000, -90000, -175000]

نکته: بارم این سوال ۱۰ نمره میباشد.

سوال ۴:

- قبل از تمام المانهای یک لیست دلخواه، “Hello” قرار دهید.
نمونه لیست:

List = [“Jane”, “Lee”, “Will”, “Brie”]

نکته: بارم این سوال ۵ نمره میباشد.

سوال ۵:

- با کمک تابع lambda المان های دولیست زیر را با هم جمع نمایید. از تابع یک lambda با دو المان ورودی استفاده نمایید:
دو لیست:

List_1 = [100, 200, 300, 400, 500]

List_2 = [1, 10, 100, 1000, 10000]

نکته: بارم این سوال ۵ نمره میباشد.

سوال ۶:

- پارامترهای یک لیست دلخواه مشابه لیست زیر را `reverse` نمایید. با استفاده از یک تابع `lambda`, `Tuple` های لیست را بر مبنای آخرین کاراکتر دومین آیت از `sort`, `Tuple` نماید.

نمونه لیست:

```
List = [(19542209, "New York"), (4887871, "Alabama"), ...]
```

نکته: بارم این سوال ۱۰ نمره میباشد.

سوال ۷:

- `Decorator` بنویسید که با استفاده از `try-exception` به یک فانکشن تنها اجازه دهد که پارامترهای زوج را بگیرد و در غیر این صورت `"Please only use even numbers!"` را برگرداند.

مثال :

```
@only_even_parameters
def myFunction(a, b):
    return a+b

myFunction(5,5) # "Please add even number!"
myFunction(4,4) # 8
```

نکته: بارم این سوال ۱۵ نمره میباشد.



سوال ۸:

• Decorator بنویسید که نتیجه یک تابع را به صورت دو برابر برگرداند:

مثال:

```
def myFunction(a, b):  
    return a+b  
  
myFunction(5,5) # 10  
  
@double_result  
def myFunction(a, b):  
    return a+b  
  
myFunction(5,5) # 20
```

نکته: بارم این سوال ۱۵ نمره میباشد.

سوال 9:

• تابعی بنویسید که مطابق نمونه زیر یک عدد را به عنوان ورودی بگیرد و عدد بخش پذیر بعدی بر آن عدد را برگرداند:



مثال:

```
gen_multiple_of_two = get_next_multiple(2)
next(gen_multiple_of_two) # 2
next(gen_multiple_of_two) # 4
next(gen_multiple_of_two) # 6
next(gen_multiple_of_two) # 8

gen_multiple_of_thirteen = get_next_multiple(13)
next(gen_multiple_of_two) # 13
next(gen_multiple_of_two) # 26
next(gen_multiple_of_two) # 39
next(gen_multiple_of_two) # 52
```

نکته: بارم این سوال ۱۵ نمره میباشد.

نکات

- مهلت ارسال تمرین تا پایان روز چهارشنبه ۱۴۰۰/۰۶/۱۷ است.
- نام فایل و ایمیل ارسالی خود را به این صورت قرار دهید: Name_hw5_maktab60 به عنوان مثال Mohammad_Ali_Kargar_hw5_maktab60.
- لطفاً از ارسال فایل تمرین در قالب pdf، jpg، png و ... خودداری فرمایید.
- در صورت لزوم یک فایل word یا pdf به عنوان توضیح در کنار کدهای خود قرار دهید.
- در صورتی که تمرین شامل چند فایل و فولدر می باشد حتماً آن ها را قالب یک فایل فشرده شده تجمیع کنید.
- در صورتی که سوالی دارید در گروه تلگرامی بپرسید.