You can view this report online at : https://www.hackerrank.com/x/tests/997250/candidates/20596591/report

Full Name:	Emmanuel Dadzie
Email:	eadobolous@gmail.com
Test Name:	**ACDS Assessment 5 - Regex, Functions, Generators, Error Handling**
Taken On:	8 Jan 2021 01:08:47 EST
Time Taken:	95 min 17 sec/ 270 min
Work Experience:	> 5 years
Invited by:	TTS
Skills Score:	Python (Basic) 150/150
Tags Score:	Easy 150/150 Exceptions 50/50 Functions 50/50 Language Proficiency 50/50 Python 150/150 python 60/60

100%

210/210

scored in **ACDS Assessment 5 - Regex, Functions, Generators, Error Handling** in 95 min 17 sec on 8 Jan 2021 01:08:47 EST

Recruiter/Team Comments:

No Comments.

Plagiarism flagged

We have marked questions with suspected plagiarism below. Please review.

	Question Description	Time Taken	Score	Status
Q1	Python: List of Even Integers > Coding	12 min 39 sec	50/ 50	✓
Q2	Python: Alphabet Filter > Coding	59 min 42 sec	50/ 50	!
Q3	Python: Return or Raise ValueError > Coding	12 min 20 sec	50/ 50	✓
Q4	Python > Multiple Choice	44 sec	10/ 10	✓
Q5	Python > Multiple Choice	2 min 1 sec	10/ 10	✓
Q6	Python > Multiple Choice	22 sec	10/ 10	✗
Q7	Python > Multiple Choice	1 min 31 sec	10/ 10	✓
Q8	Python > Multiple Choice	59 sec	10/ 10	✓

QUESTION 1

Correct Answer

Score 50

Python: List of Even Integers › Coding

Python | Easy | Functions

QUESTION DESCRIPTION

In this challenge, you are required to implement a function that:

1. is named *even*.
2. takes 2 integer arguments, *start*, and *n*.
3. returns a list of *n* smallest even integers greater than or equal to *start*, in ascending order.

Implementation of the function will be tested by a provided code stub on several input files. Each input file contains parameters for the function call. The function will be called with those parameters, and the result of its execution will be printed to the standard output by the provided code.

Constraints

- $1 \le start, n \le 100$

▼ Input Format Format for Custom Testing

In the first and only line, there are two space-separated integers, *start*, and *n*.

▼ Sample Case 0

Sample Input

```
STDIN       Function
-----       --------
2 4     →   start = 2, n = 4
```

Sample Output

```
2 4 6 8
```

Explanation

The function must return a list of the 4 smallest even integers that are greater than or equal to 2, sorted into ascending order: 2, 4, 6, and 8.

▼ Sample Case 1

Sample Input

```
STDIN       Function
-----       --------
5 7     →   start = 5, n = 7
```

Sample Output

```
6 8 10 12 14 16 18
```

Explanation

The function must return a list of the 7 smallest even integers that are greater than or equal to 5, sorted in ascending order: 6, 8, 10, 12, 14, 16, 18.

INTERVIEWER GUIDELINES

Setter's solution (pawel):

```
def even(start, n):
    if start % 2 == 1:
```

```
    if start % 2 == 1:
        start += 1
    return [start+2*i for i in range(n)]
```

CANDIDATE ANSWER

Language used: **Python 3**

```
from itertools import count

def even(start, n):
    # write your code here
    lst = []
    for i in count(start):
        if i % 2 == 0:
           lst.append(i)
        else:
            continue
        if len(lst) == n:
            break
    return lst

```

TESTCASE	DIFFICULTY	TYPE	STATUS	SCORE	TIME TAKEN	MEMORY USED
TestCase 0	Easy	Sample case	Success	1	0.0429 sec	10.7 KB
TestCase 1	Easy	Sample case	Success	1	0.0417 sec	10.9 KB
TestCase 2	Easy	Sample case	Success	1	0.0381 sec	11 KB
TestCase 3	Easy	Sample case	Success	3	0.0449 sec	10.9 KB
TestCase 4	Easy	Sample case	Success	4	0.0369 sec	10.7 KB
TestCase 5	Easy	Hidden case	Success	4	0.0427 sec	11 KB
TestCase 6	Easy	Hidden case	Success	4	0.0406 sec	10.5 KB
TestCase 7	Easy	Hidden case	Success	4	0.0657 sec	10.9 KB
TestCase 8	Easy	Hidden case	Success	4	0.0402 sec	11 KB
TestCase 9	Easy	Hidden case	Success	4	0.0479 sec	10.9 KB
TestCase 10	Easy	Hidden case	Success	4	0.0463 sec	10.7 KB
TestCase 11	Easy	Hidden case	Success	4	0.0458 sec	10.6 KB
TestCase 12	Easy	Hidden case	Success	4	0.045 sec	10.6 KB
TestCase 13	Easy	Hidden case	Success	4	0.0487 sec	11 KB
TestCase 14	Easy	Hidden case	Success	4	0.0395 sec	10.9 KB

No Comments

QUESTION 2

Needs Review

Score 50

Python: Alphabet Filter › Coding

Language Proficiency | Python | Easy

QUESTION DESCRIPTION

Given a string consisting of only lowercase characters, create two methods that remove all the consonants or vowels from the given word. They must retain the original order of the characters in the returned strings.

Example:

Example:

s = 'onomatopoeia'

- The *filter_vowels* method removes all vowels from *s* and returns the string '*nmtp*'.
- The *filter_consonants* method removes all consonants from *s* and returns the string '*ooaooeia*'.

Function Description

For a given definition of a class *LetterFilter*, complete its methods *filter_vowels* and *filter_consonants*. The class takes a string in the constructor and stores it to its *s* attribute. The method *filter_vowels* must return a new string with all vowels removed from it. Similarly, the method *filter_consonants* must return a new string with all consonants removed from it.

Constraints

- The string contains only lowercase letters in the range ascii[a-z]
- The string contains at least one vowel and at least one consonant

▼ Input Format For Custom Testing

The first line contains a string, *s*, that denotes the string to be transformed.

▼ Sample Case 0

Sample Input 0

```
STDIN        Function
-----        --------
hackerrank → string s = 'hackerrank'
```

Sample Output 0

```
hckrrnk
aea
```

Explanation 0

- The first result is after removing all vowels, {a, e, i, o, u}, from the string.
- The second result is after removing all consonants.

▼ Sample Case 1

Sample Input 1

```
STDIN          Function
-----          --------
programming →  string s = 'programming'
```

Sample Output 1

```
prgrmmng
oai
```

Explanation 1

- The first result is after removing all vowels, {a, e, i, o, u}, from the string.
- The second result is after removing all consonants.

CANDIDATE ANSWER

Language used: **Python 3**

```
# Enter your code here.
# Complete the classes below.
# Reading the inputs and writing the outputs are already done for you.
#
class LetterFilter:
```

```
    vowels = ["a", "e", "i", "o", "u"]

    def __init__(self, s):
        self.s = s

    def filter_vowels(self):
    # Enter your code here
        new_str = self.s
        for i in new_str:
            if i in LetterFilter.vowels:
                new_str = new_str.replace(i, "")
        # Return a string
        return new_str

    def filter_consonants(self):
    # Enter your code here
        new_str = self.s
        for i in new_str:
            if i not in LetterFilter.vowels:
                new_str= new_str.replace(i, "")
        # Return a string
        return new_str
```

TESTCASE	DIFFICULTY	TYPE	STATUS	SCORE	TIME TAKEN	MEMORY USED
TestCase 0	Easy	Sample case	Success	1	0.0274 sec	7.65 KB
TestCase 1	Easy	Sample case	Success	1	0.0262 sec	7.6 KB
TestCase 2	Easy	Sample case	Success	1	0.0232 sec	7.31 KB
TestCase 3	Easy	Hidden case	Success	3	0.0258 sec	7.31 KB
TestCase 4	Easy	Hidden case	Success	3	0.0236 sec	7.34 KB
TestCase 5	Easy	Hidden case	Success	3	0.0269 sec	7.51 KB
TestCase 6	Medium	Sample case	Success	5	0.023 sec	7.65 KB
TestCase 7	Medium	Hidden case	Success	5	0.0242 sec	7.33 KB
TestCase 8	Easy	Hidden case	Success	7	0.0482 sec	7.59 KB
TestCase 9	Easy	Hidden case	Success	7	0.0493 sec	7.38 KB
TestCase 10	Easy	Sample case	Success	7	1.9705 sec	7.75 KB
TestCase 11	Easy	Hidden case	Success	7	1.9999 sec	7.86 KB

No Comments

QUESTION 3

Correct Answer

Score 50

Python: Return or Raise ValueError › Coding

Python | Easy | Exceptions

QUESTION DESCRIPTION

Implement a function, *multiply,* that takes 3 integer arguments: *a*, *b,* and *bound.*

- If the result of multiplying *a* and *b* is less than or equal to *bound*, the function returns the result.
- If the result of multiplying *a* and *b* is greater than *bound*, the function raises a ValueError exception with the following message: if *a=2*, *b=5*, and *bound=8*, then the message must be: "multiplication of 2 and 5 with bound 8 not possible"

Implementation of the function will be tested by a provided code stub on several input files. Each input file

contains several queries, and each query contains parameters for the function call. The function will be called with those parameters and the result of its execution will be printed to the standard output by the provided code.

Constraints

- 1 ≤ the number of queries in one test file ≤ 1000
- 1 ≤ *a, b* ≤ *bound* ≤ 10^5

▼ Input Format Format for Custom Testing

Input from stdin will be processed as follows and passed to the function.

In the first line, there is a single integer *q*.
Then, *q* lines follow. In the i^{th} of them, there are 3 space-separated integers that denote the values for *a*, *b*, and *bound* respectively.

▼ Sample Case 0

Sample Input

```
STDIN       Function
-----       --------
3         → number of queries q = 3
5 2 23    → query[0]: a = 5, b = 2, bound = 23
5 2 10    → query[1]: a = 5, b = 2, bound = 10
5 2 9     → query[2]: a = 5, b = 2, bound = 9
```

Sample Output

```
10
10
multiplication of 5 and 2 with bound 9 not possible
```

Explanation
There are 3 queries. In all of them, a=5 and b=2 but the bound parameter is different.

- In the first query, bound is 23 and since 5*2=10 is not greater than 23, 10 is returned by the function.
- Similarly, in the second query, 5*2=10 is not greater than 10, so 10 is returned.
- In the last query, bound is 9, and since 5*2 is greater than 9, the function raises an exception with the message "multiplication of 2 and 5 with bound 9 not possible".
-

▼ Sample Case 1

Sample Input

```
STDIN         Function
-----         --------
2           → number of queries q = 2
11 11 120   → query[0]: a = 11, b = 11, bound = 120
7 8 100     → query[1]: a = 7, b = 8, bound = 100
```

Sample Output

```
multiplication of 11 and 11 with bound 120 not possible
56
```

Explanation
There are 2 queries.

- In the first query, 11*11=121 is greater than 120, so the function raises the exception with the message "multiplication of 11 and 11 with bound 120 not possible"

- In the second query, 7*8=56 is not greater than 100, so 56 is returned.

INTERVIEWER GUIDELINES

Setter's solution (pawel):

```
def multiply(a, b, bound):
    if a*b <= bound:
        return a*b
    raise ValueError("multiplication of %d and %d with bound %d not
possible" % (a, b, bound))
```

CANDIDATE ANSWER

Language used: **Python 3**

```python
def multiply(a, b, bound):
    # write your code here
    result = a * b
    if result <= bound:
        return result
    else:
        raise ValueError(f"multiplication of {a} and {b} with bound {bound}
not possible")


```

TESTCASE	DIFFICULTY	TYPE	STATUS	SCORE	TIME TAKEN	MEMORY USED
TestCase 0	Easy	Sample case	Success	1	0.0473 sec	10.9 KB
TestCase 1	Easy	Sample case	Success	1	0.0399 sec	10.9 KB
TestCase 2	Easy	Sample case	Success	1	0.0449 sec	10.9 KB
TestCase 3	Easy	Sample case	Success	3	0.0438 sec	11.1 KB
TestCase 4	Easy	Sample case	Success	4	0.0723 sec	10.6 KB
TestCase 5	Easy	Hidden case	Success	4	0.0385 sec	10.5 KB
TestCase 6	Easy	Hidden case	Success	4	0.0386 sec	10.6 KB
TestCase 7	Easy	Hidden case	Success	4	0.0453 sec	11 KB
TestCase 8	Easy	Hidden case	Success	4	0.0491 sec	10.7 KB
TestCase 9	Easy	Hidden case	Success	4	0.047 sec	11.1 KB
TestCase 10	Easy	Hidden case	Success	4	0.0526 sec	11 KB
TestCase 11	Easy	Hidden case	Success	4	0.0462 sec	10.6 KB
TestCase 12	Easy	Hidden case	Success	4	0.0498 sec	10.9 KB
TestCase 13	Easy	Hidden case	Success	4	0.0666 sec	11 KB
TestCase 14	Easy	Hidden case	Success	4	0.0604 sec	10.7 KB

No Comments

QUESTION 4

Correct Answer

Score 10

Python › Multiple Choice python

QUESTION DESCRIPTION

What is the structure used for handling runtime excpetions?

CANDIDATE ANSWER

Options: (Expected answer indicated with a tick)

- [x] Try/Except
- [ ] Try/Catch
- [ ] Attempt/Except
- [ ] Attempt/Catch

No Comments

QUESTION 5

Correct Answer

Score 10

Python › Multiple Choice python

QUESTION DESCRIPTION

What can you use to throw your own exceptions?

CANDIDATE ANSWER

Options: (Expected answer indicated with a tick)

- [ ] throw
- [x] raise
- [ ] err
- [ ] printerr()

No Comments

QUESTION 6

Wrong Answer

Score 10

Python › Multiple Choice python

QUESTION DESCRIPTION

True or False: Code written in a finally clause will print regardless of the success of a try block.

CANDIDATE ANSWER

Options: (Expected answer indicated with a tick)

- TRUE
- ✓ FALSE

Student gave correct answer - answer key has been corrected.
- TTS Curriculum (26 Jan 2021 12:32:50 EST)

QUESTION 7

Correct Answer

Score 10

Python › Multiple Choice python

QUESTION DESCRIPTION

What type of function is this? def say_hello():
print("Hello")

CANDIDATE ANSWER

Options: (Expected answer indicated with a tick)

- ✓ void
- value-returning

No Comments

QUESTION 8

Correct Answer

Score 10

Python › Multiple Choice python

QUESTION DESCRIPTION

What can you use to throw your own exceptions?

CANDIDATE ANSWER

Options: (Expected answer indicated with a tick)

- throw
- ✓ raise
- err
- printerr()

No Comments

QUESTION 9

Correct Answer

Score 10

Python › Multiple Choice python

QUESTION DESCRIPTION

What will the following RegEx return? import re

```
s = "Right now I am taking an assessment. Later, I am going to the gym."
re.findall(r'\w+ing\b', s)
```

CANDIDATE ANSWER

Options: (Expected answer indicated with a tick)

- ['taking']
- ['going']
- ✓ ['taking', 'going']
- ['Right', 'now', 'I', 'am', 'an', 'assessment', 'Later', 'I', 'am', 'to', 'the', 'gym']

No Comments

PDF generated at: 24 Feb 2021 17:33:53 UTC