Python_basic_pragramming_20

In [ ]:

```
1. Create a function that takes a list of strings and integers, and filters out the l
so that it returns a list of integers only.
Examples:
filter_list([1, 2, 3, "a", "b", 4]) [1, 2, 3, 4]
filter_list(["A", 0, "Edabit", 1729, "Python", "1729"]) [0, 1729]
filter_list(["Nothing", "here"]) []
```

In [1]:

```
def filter_list(in_list):
    out_list = []
    for ele in in_list:
        if type(ele) == int:
            out_list.append(ele)
    print(f'Output{out_list}')

filter_list([1, 2, 3, "a", "b", 4])
filter_list(["A", 0, "Edabit", 1729, "Python", "1729"])
filter_list(["Nothing", "here"])
```

```
Output[1, 2, 3, 4]
Output[0, 1729]
Output[]
```

In [ ]:

```
2. Given a list of numbers, create a function which returns the list but with each
element's index in the list added to itself. This means you add 0 to the number at
index 0, add 1 to the number at index 1, etc...
Examples:
add_indexes([0, 0, 0, 0, 0]) [0, 1, 2, 3, 4]
add_indexes([1, 2, 3, 4, 5]) [1, 3, 5, 7,9]
add_indexes([5, 4, 3, 2, 1]) [5, 5, 5, 5, 5]
```

In [2]:

```
def add_indexes(in_list):
    out_list = []
    for ele in range(len(in_list)):
        out_list.append(ele+in_list[ele])
    print(f'{in_list} {out_list}')

add_indexes([0, 0, 0, 0, 0])
add_indexes([1, 2, 3, 4,5])
add_indexes([5, 4, 3, 2, 1])
```

```
[0, 0, 0, 0, 0] [0, 1, 2, 3, 4]
[1, 2, 3, 4, 5] [1, 3, 5, 7, 9]
[5, 4, 3, 2, 1] [5, 5, 5, 5, 5]
```

In [ ]:

```
3. Create a function that takes the height and radius of a cone as arguments and retu
the volume of the cone rounded to the nearest hundredth.
See the resources tab for the formula.
Examples:
cone_volume(3, 2) 12.57
cone_volume(15, 6) 565.49
cone_volume(18,0) 0
```

In [3]:

```
import math

def cube_volume(height,radius):
    output = ((math.pi) *pow(radius,2))*(height/3)
    print(f'Output{output:.2f}')

cube_volume(3,2)
cube_volume(15,6)
cube_volume(18,0)
```

```
Output12.57
Output565.49
Output0.00
```

In [ ]:

```
4. This Triangular Number Sequence is generated from a pattern of dots that form a tri
The first 5 numbers of the sequence, or dots,are: 1, 3, 6, 10, 15
This means that the first triangle has just one dot, the second one has three dots,
the third one has 6 dots and so on. Write a function that gives the number of dots
with its corresponding triangle number of the sequence.
Examples:
triangle(1) 1
triangle(6) 21
triangle(215) 23220
```

In [1]:

```
def triangle(in_num):
    print(f'Output {int((in_num)*((in_num+1)/2))}')

triangle(1)
triangle(6)
triangle(215)
```

```
Output 1
Output 21
Output 23220
```

In [ ]:

```
5. Create a function that takes a list of numbers between 1 and 10 (excluding one numb
and returns the missing number.
Examples:
missing_num([1, 2, 3, 4, 6, 7, 8, 9, 10]) 5
missing_num([7, 2, 3, 6, 5, 9, 1, 4,8]) 10
missing_num([10, 5, 1, 2, 4, 6, 8, 3, 9]) 7
```

In [2]:

```
def missing_num(in_list):
    for i in range(1,11):
        if i not in in_list:
            print(f'{in_list} {i}')

missing_num([1, 2, 3, 4, 6, 7, 8, 9, 10])
missing_num([7, 2, 3, 6, 5, 9, 1, 4, 8])
missing_num([10, 5, 1, 2, 4, 6, 8, 3, 9])
```

```
[1, 2, 3, 4, 6, 7, 8, 9, 10] 5
[7, 2, 3, 6, 5, 9, 1, 4, 8] 10
[10, 5, 1, 2, 4, 6, 8, 3, 9] 7
```