

- Functions Tasks

1. Average of cubes of all even numbers in range
2. Average of first N prime numbers
3. Check the given number is perfect or not
4. print odd number in reverse order in a range

- Stings Tasks

- print even ASCII characters from input string
 - input - python
- print words which are not ends with vowels
 - input -- Welcome to Apssdc python workshop thank you
 - output -- Apssdc python workshop thank

In [2]: *## Average of cubes of all even numbers in range*

```
def avg_cubes(s,e):
    s1 = c = 0
    for i in range(s,e+1):
        if i%2 == 0:
            s1 = s1 + i**3
            c = c+1
    return s1//c

s = int(input('enter start number: '))
e = int(input('enter end number: '))
avg_cubes(s,e)
```

enter start number: 1
enter end number: 10

Out[2]: 360

In [3]: `2**3+4**3+6**3+8**3+10**3`

Out[3]: 1800

In [4]: `1800//5`

Out[4]: 360

In [5]: *## Average of first N prime numbers*

```
def avg_prime(s,e):
    s1 = c = 0
    for i in range(s,e+1):
        fc = 0
        for j in range(1,i+1):
            if(i%j==0):
                fc = fc+1
```

```

        if fc == 2:
            s1 = s1 +i
            c = c+1
        return s1//c

s = int(input())
e = int(input())
avg_prime(s,e)

```

1
10

Out[5]: 4

In [6]: 2+3+5+7

Out[6]: 17

In [7]: 17//4

Out[7]: 4

In [10]: *## Check the given number is perfect or not*

```

def is_Perfect(n):
    f_sum = 0
    for i in range(1,n):
        if(n%i == 0):
            f_sum = f_sum + i
    if f_sum == n:
        print('Perfect')
    else:
        print('Not perfect')

n = int(input('enter number: '))
is_Perfect(n)

```

enter number: 6
Perfect

In [12]: *## print odd number in reverse order in a range*

```

def odd_reverse(e,s): ## 10, 1
    for i in range(e,s+1,-1): ## 10,1,-1
        if i%2 != 0:
            print(i,end=' ')
s = int(input())
e = int(input())
odd_reverse(e,s)

```

1
10
9 7 5 3

In [15]:

```
# print even ASCII characters from input string
# input - python

def even_ascii(s):
    for i in s:
        if(ord(i)%2 == 0):
            print(i,end=' ')
s = input()
even_ascii(s)
```

```
python
p t h n
```

In [13]: `ord('p')`

Out[13]: 112

In [14]: `s = 'python'`
`for i in s:`
 `print(i,end=' ')`

```
p y t h o n
```

In [16]: `# print words which are not ends with vowels`
`# input -- Welcome to Apssdc python workshop thank you`
`# output -- Apssdc python workshop thank`

```
def vowel(x): ## x = i[-1]
    if(x == 'a' or x == 'e' or x == 'i' or x == 'o' or x == 'u'
       or x == 'A' or x == 'E' or x == 'I' or x == 'O' or x == 'U'):
        return True
    return False

s2 = input()
l = s2.split()
for i in l:
    if not(vowel(i[-1])): ## i = welcome i= to i= Apssdc
        print(i,end=' ')
```

```
Welcome to Apssdc python workshop thank you
Apssdc python workshop thank
```

In [18]: `s2 = input()`
`l = s2.split()`
`l`

```
Welcome to Apssdc python workshop thank you
```

Out[18]: ['Welcome', 'to', 'Apssdc', 'python', 'workshop', 'thank', 'you']

In []:

In []:

List cont...

```
In [20]: ## Empty list

l = []
type(l)
```

Out[20]: list

```
In [22]: li = [1,2,3,4,'abc',8.6]
print(li[0])
```

1

```
In [23]: li
```

Out[23]: [1, 2, 3, 4, 'abc', 8.6]

```
In [24]: li[2] = 'xyz'
```

```
In [25]: li
```

Out[25]: [1, 2, 'xyz', 4, 'abc', 8.6]

```
In [26]: print(dir(__builtins__),end=' ')
```

```
['ArithmeticError', 'AssertionError', 'AttributeError', 'BaseException', 'BlockingIOError', 'BrokenPipeError', 'BufferError', 'BytesWarning', 'ChildProcessError', 'ConnectionAbortedError', 'ConnectionError', 'ConnectionRefusedError', 'ConnectionResetError', 'DeprecationWarning', 'EOFError', 'Ellipsis', 'EnvironmentError', 'Exception', 'False', 'FileExistsError', 'FileNotFoundError', 'FloatingPointError', 'FutureWarning', 'GeneratorExit', 'IOError', 'ImportError', 'ImportWarning', 'IndentationError', 'IndexError', 'InterruptedError', 'IsADirectoryError', 'KeyError', 'KeyboardInterrupt', 'LookupError', 'MemoryError', 'ModuleNotFoundError', 'NameError', 'None', 'NotADirectoryError', 'NotImplemented', 'NotImplementedError', 'OSError', 'OverflowError', 'PendingDeprecationWarning', 'PermissionError', 'ProcessLookupError', 'RecursionError', 'ReferenceError', 'ResourceWarning', 'RuntimeError', 'RuntimeWarning', 'StopAsyncIteration', 'StopIteration', 'SyntaxError', 'SyntaxWarning', 'SystemError', 'SystemExit', 'TabError', 'TimeoutError', 'True', 'TypeError', 'UnboundLocalError', 'UnicodeDecodeError', 'UnicodeEncodeError', 'UnicodeError', 'UnicodeTranslateError', 'UnicodeWarning', 'UserWarning', 'ValueError', 'Warning', 'WindowsError', 'ZeroDivisionError', '__IPYTHON__', '__build_class__', '__debug__', '__doc__', '__import__', '__loader__', '__name__', '__package__', '__spec__', 'abs', 'all', 'any', 'ascii', 'bin', 'bool', 'breakpoint', 'bytearray', 'bytes', 'callable', 'chr', 'classmethod', 'compile', 'complex', 'copyright', 'credits', 'delattr', 'dict', 'dir', 'display', 'divmod', 'enumerate', 'eval', 'exec', 'filter', 'float', 'format', 'frozenset', 'get_ipython', 'getattr', 'globals', 'hasattr', 'hash', 'help', 'hex', 'id', 'input', 'int', 'isinstance', 'issubclass', 'iter', 'len', 'license', 'list', 'locals', 'map', 'max', 'memoryview', 'min', 'next', 'object', 'oct', 'open', 'ord', 'pow', 'print', 'property', 'range', 'repr', 'reversed', 'round', 'set', 'setattr', 'slice', 'sorted', 'staticmethod', 'str', 'sum', 'super', 'tuple', 'type', 'vars', 'zip']
```

```
In [43]: ## bulit in functions

l = [60,2,8,0,1,6,3]
```

```
print('type',type(l))
print('length',len(l))
print('max value',max(l))
print('min value',min(l))
print('sum value',sum(l))
print('sorted list',sorted(l))
print('sorted in reverse order',sorted(l,reverse=True))
```

```
type <class 'list'>
length 7
max value 60
min value 0
sum value 80
sorted list [0, 1, 2, 3, 6, 8, 60]
sorted in reverse order [60, 8, 6, 3, 2, 1, 0]
```

In [34]:

```
l = [89,5,6,'a']
print(sorted(l))
```

```
-----
TypeError                                Traceback (most recent call last)
<ipython-input-34-e0d6eae92bb2> in <module>
      1 l = [89,5,6,'a']
----> 2 print(sorted(l))
```

TypeError: '<' not supported between instances of 'str' and 'int'

In [41]:

```
li = ['z','g','a','s','i','o']
print(type(li))
print(len(li))
print(max(li))
print(min(li))
# print(sum(li))
print(sorted(li))
print(sorted(li,reverse=True))
```

```
<class 'list'>
6
z
a
['a', 'g', 'i', 'o', 's', 'z']
['z', 's', 'o', 'i', 'g', 'a']
```

In [48]:

```
## Accessing list elements
## Indexing
print(l)
print(l[2]) ## forward index
print(l[-1]) ## backward index
print(l[0:3]) ## Slicing
print(l[::-1]) ## reverse of list
```

```
[60, 2, 8, 0, 1, 6, 3]
8
3
[60, 2, 8]
[3, 6, 1, 0, 8, 2, 60]
```

In [49]:

```
## concatenation
```

```
l1 = [1,2,3,4,5]
l2 = [5,6,7,8,9]
s = l1+l2
s
```

Out[49]: [1, 2, 3, 4, 5, 5, 6, 7, 8, 9]

```
In [51]: li = [1,2,3,4,[5,6,7],8,9,10]
print(li[4])
print(li[4][1])  ## li[outer list index][inner list/sub/nested list index]
```

```
[5, 6, 7]
6
```

```
In [52]: print(dir(list),end=' ')
```

```
['_add_', '__class__', '__contains__', '__delattr__', '__delitem__', '__dir__', '__doc__', '__eq__', '__format__', '__ge__', '__getattr__', '__getitem__', '__gt__', '__hash__', '__iadd__', '__imul__', '__init__', '__init_subclass__', '__iter__', '__le__', '__len__', '__lt__', '__mul__', '__ne__', '__new__', '__reduce__', '__reduce_ex__', '__repr__', '__reversed__', '__rmul__', '__setattr__', '__setitem__', '__sizeof__', '__str__', '__subclasshook__', 'append', 'clear', 'copy', 'count', 'extend', 'index', 'insert', 'pop', 'remove', 'reverse', 'sort']
```

```
In [53]: l1 = [12,5,7,9,3]
l1.append(50)
print(l1)
```

```
[12, 5, 7, 9, 3, 50]
```

```
In [54]: l1.append([1,2,3])
l1
```

Out[54]: [12, 5, 7, 9, 3, 50, [1, 2, 3]]

```
In [56]: l = [1,2,3,4]
h = [5,6,7,8]
l.extend(h)
h.extend(l)
print(l)
print(h)
```

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

```
In [57]: l.count(1)
```

Out[57]: 1

```
In [58]: l.append(1)
```

```
In [59]: 1
```

Out[59]: [1, 2, 3, 4, 5, 6, 7, 8, 1]

In [60]: `l.count(1)`

Out[60]: 2

In [61]: `l.index(1)`

Out[61]: 0

In [62]: `l.index(3)`

Out[62]: 2

In [63]: `l.pop(5)`

Out[63]: 6

In [64]: `l`

Out[64]: [1, 2, 3, 4, 5, 7, 8, 1]

In [65]: `l.pop(1)`

Out[65]: 2

In [66]: `l`

Out[66]: [1, 3, 4, 5, 7, 8, 1]

In [67]: `l.pop()`

Out[67]: 1

In [68]: `l`

Out[68]: [1, 3, 4, 5, 7, 8]

In [69]: `l.pop()`

Out[69]: 8

In [70]: `l`

Out[70]: [1, 3, 4, 5, 7]

```
In [71]: 1.remove(4)
1
```

Out[71]: [1, 3, 5, 7]

```
In [72]: 1.insert(4,'abc')  ## (index position,value)
1
```

Out[72]: [1, 3, 5, 7, 'abc']

```
In [76]: 12 = [4,5,6,7,2]
12.sort()
12
```

Out[76]: [2, 4, 5, 6, 7]

```
In [77]: 12.reverse()
```

```
In [78]: 12
```

Out[78]: [7, 6, 5, 4, 2]

```
In [79]: 12
```

Out[79]: [7, 6, 5, 4, 2]

```
In [80]: 12.clear()  ## it removes the data in the given list
```

```
In [81]: 12
```

Out[81]: []

```
In [82]: del 12
```

```
In [83]: 12
```

```
-----
NameError                                Traceback (most recent call last)
<ipython-input-83-ea320d2ace30> in <module>
----> 1 12
```

NameError: name '12' is not defined

```
In [84]:  ## dynamic input
```



```
s = input().split()
print(s)
```

```
1 2 3 4 5 5
['1', '2', '3', '4', '5', '5']
```

In [86]:

```
for i in s:
    print(type(i),end=' ')
```

```
<class 'str'> <class 'str'> <class 'str'> <class 'str'> <class 'str'> <class 'str'>
```

In [88]:

```
## using map function
# map(converttype,input)

res = list(map(int,s))
res
```

Out[88]: [1, 2, 3, 4, 5, 5]

In [90]:

```
for i in res:
    print(type(i))
```

```
<class 'int'>
<class 'int'>
<class 'int'>
<class 'int'>
<class 'int'>
<class 'int'>
```

In [1]:

```
## input -- 1 2 3 4 5
## output -- 1 4 9 16 25

li = input().split() ## 1 2 3 4 5
li1 = [] ## empty List
for i in li:
    li1.append(int(i)**2)
print(li1)
```

```
1 2 3 4 5
[1, 4, 9, 16, 25]
```

List Tasks

- Remove the Duplicate elements
 - input -- [11,5,9,1,2,11,5]
 - output -- [11,5,9,1,2]
- Unique elements
 - input -- [11,5,9,1,2,11,5]
 - output -- [9,1,2]

In []:

Tuple

- represent with ()
- Ordered and Immutable in nature
- used to hetrogenous data

```
In [1]: ## empty tuple

t = ()
type(t)
```

Out[1]: tuple

```
In [2]: t = (1,2,3,'a',7.8)
print(t)
```

(1, 2, 3, 'a', 7.8)

```
In [3]: t[0]
```

Out[3]: 1

```
In [4]: t
```

Out[4]: (1, 2, 3, 'a', 7.8)

```
In [5]: t[1] = 't'
```

```
-----
TypeError                                Traceback (most recent call last)
<ipython-input-5-f6b971756c86> in <module>
----> 1 t[1] = 't'
```

TypeError: 'tuple' object does not support item assignment

```
In [7]: t[-1]
```

Out[7]: 7.8

```
In [8]: t[0:4]
```

Out[8]: (1, 2, 3, 'a')

```
In [9]: t[::-1]
```

Out[9]: (7.8, 'a', 3, 2, 1)

```
In [10]: print(dir(t),end=' ')
```

```
['_add_', '__class__', '__contains__', '__delattr__', '__dir__', '__doc__', '__eq__',  
'_format_', '__ge__', '__getattribute__', '__getitem__', '__getnewargs__', '__gt__',  
'_hash_', '__init__', '__init_subclass__', '__iter__', '__le__', '__len__', '__lt__',  
'_mul_', '__ne__', '__new__', '__reduce__', '__reduce_ex__', '__repr__', '__rmul__',  
'_setattr_', '__sizeof__', '__str__', '__subclasshook__', 'count', 'index']
```

```
In [11]: t
```

```
Out[11]: (1, 2, 3, 'a', 7.8)
```

```
In [12]: t.count(2)
```

```
Out[12]: 1
```

```
In [13]: t.index(2)
```

```
Out[13]: 1
```

```
In [14]: t.index(3)
```

```
Out[14]: 2
```

```
In [15]: t
```

```
Out[15]: (1, 2, 3, 'a', 7.8)
```

```
In [16]: type(t)
```

```
Out[16]: tuple
```

```
In [21]: r = list(t)  
print(r)  
type(r)
```

```
[1, 2, 3, 'a', 7.8]
```

```
Out[21]: list
```

```
In [18]: t
```

```
Out[18]: (1, 2, 3, 'a', 7.8)
```

```
In [19]: type(t)
```

```
Out[19]: tuple
```

```
In [22]: t
```

```
Out[22]: (1, 2, 3, 'a', 7.8)
```

```
In [26]: r = list(t)
         res = r.pop()
         res
         tuple(r)
```

```
Out[26]: (1, 2, 3, 'a')
```

```
In [27]: l = [1,1,3,3,5,6,3]
         l.count(3)
```

```
Out[27]: 3
```

```
In [30]: t = int(input()).split()
         t
```

```
4 5 6 7
```

```
-----
ValueError                                Traceback (most recent call last)
<ipython-input-30-3f1cf9f56a38> in <module>
----> 1 t = int(input()).split()
      2 t
```

```
ValueError: invalid literal for int() with base 10: '4 5 6 7'
```

```
In [31]: 1
```

```
Out[31]: [1, 1, 3, 3, 5, 6, 3]
```

```
In [32]: l.pop()
```

```
Out[32]: 3
```

```
In [33]: l.remove(5)
```

```
In [34]: 1
```

```
Out[34]: [1, 1, 3, 3, 6]
```

```
In [35]: 1
```

```
Out[35]: [1, 1, 3, 3, 6]
```

```
In [36]: type(1)
```

Out[36]: list

```
In [37]: y = (4,5,6,7)
         type(y)
```

Out[37]: tuple

```
In [ ]:
```