1. Use list comprehension to create a list of all numbers between 13 and 69 (both inclusive) that are divisible by 3.

In [1]:
```
[number for number in range(13,69+1,1) if number % 3 == 0]
```

Out[1]:
```
[15, 18, 21, 24, 27, 30, 33, 36, 39, 42, 45, 48, 51, 54, 57, 60, 63, 66, 69]
```

2. Use list comprehension to create a list of tuples of the first & last letters of every word in the string “Farmer jack realized that big yellow quilts were expensive”.

In [2]:
```
my_string = 'Farmer jack realized that big yellow quilts were expensive'
```

In [3]:
```
[(word[0], word[-1]) for word in my_string.split(' ')]
```

Out[3]:
```
[('F', 'r'),
 ('j', 'k'),
 ('r', 'd'),
 ('t', 't'),
 ('b', 'g'),
 ('y', 'w'),
 ('q', 's'),
 ('w', 'e'),
 ('e', 'e')]
```

3. Read a string and a width, wrap the string into a paragraph of width.

In [4]:
```
my_string = 'ABCDEFGHIJKLIMNOQRSTUVWXYZ'
width = 4
```

In [5]:
```
result = ''
for i in range(0,len(my_string),width):
    result+=my_string[i:i+width]+"\n"
print(result)
```

```
ABCD
EFGH
IJKL
IMNO
QRST
UVWX
YZ
```

In [6]:
```
# Alternate solution
import textwrap
print(textwrap.fill(my_string,width))
```

```
ABCD
EFGH
IJKL
IMNO
QRST
UVWX
YZ
```

4. Write a Python program to square and cube every number in a given list of integers using Lambda.

In [7]:
```
my_list = [1,2,3,4,5]
list(map(lambda x : (x**2, x**3), my_list))
```

Out[7]:
```
[(1, 1), (4, 8), (9, 27), (16, 64), (25, 125)]
```

5. Write a Python program to find intersection of two given arrays using Lambda.

In [8]:
```
list1 = [1,2,3,4,5]
list2 = [4,5,6,7,8]
```

In [9]:
```
list(filter(lambda num: num in list1, list2))
```

Out[9]:
```
[4, 5]
```

6. Write a Python program to add three given lists using Python map and Lambda.

In [10]:
```
list1 = [0,1,8]
list2 = [3,6,0]
list3 = [6,2,1]
```

In [11]:
```
list(map(lambda x,y,z : (x+y+z),list1,list2,list3))
```

Out[11]:
```
[9, 9, 9]
```

7. Using Lambda & filter, Given a list of strings, find all palindromes.

In [12]:
```
list(filter(lambda x: x == x[::-1], ['cat', 'dad', 'madam', 'bad']))
```

Out[12]:
```
['dad', 'madam']
```

8. Place a break statement in the for loop so that it prints from 0 to 7 only (including 7).

In [13]:
```
for i in range(100):
    print(i)
    if i == 7:
        break
```

```
0
1
2
3
4
5
6
7
```

9. Add an if statement and a continue statement to the loop so that it skips when iterator equals sun.

In [14]:
```
weather=["snow", "rain", "sun", "clouds"]
for i in weather:
    if i == 'sun':
        continue
    else:
        print(i)
```

```
snow
rain
clouds
```

10. What is the difference between a method and a function? Explain with example.

A function is a block of code to carry out a specific task, will contain its own scope and is called by name. All functions may contain zero(no) arguments or more than one arguments. On exit, a function can return one or more values.

A method in python is somewhat similar to a function, except it is associated with object/classes. Methods in python are very similar to functions except for two major differences.

The method is implicitly used for an object for which it is called.

The method is accessible to data that is contained within the class.

Method

In [15]:
```
from datetime import date
class Person:
    "This is a person class"
    name = input("Enter your Name:")
    happy_new_year = date.today().year
    def greet(self):
        print(f"Hello {self.name}!")
        print(f"Welcome to {self.happy_new_year}!")

# create a new object of Person class called "student"
student = Person()
```

```
Enter your Name:Mir
```

In [16]:
```
student.greet()
```

```
Hello Mir!
Welcome to 2022!
```

Here greet() is a method as it is attached with 'Person' class.

Function

In [17]:
```
def greet(string):
        print(f"This is your string - {string}.")
        print(f"The length of your string is {len(string)} bytes!")
```

In [18]:
```
greet("Mir")
```

```
This is your string - Mir.
The length of your string is 3 bytes!
```

Here greet() is a function as it is not attached to any class.