In [1]:
```
for i in range(0,4):
    print("hello, world!")
```

```
hello, world!
hello, world!
hello, world!
hello, world!
```

In [2]:
```
def duplicate(lst):
    lst1 = []
    for i in lst:
        if i not in lst1:
            lst1.append(i)
        else:
            print(i,end=' ')
lst = [1,2,3,4,5,2,3,4,7,9,5]
duplicate(lst)
```

```
2 3 4 5
```

In [3]:
```
def group(x,l):
        gl=[]
        g=[]
        i=0
        while i<len(x):
                if(len(gl)<l):
                        gl.append(x[i])
                        i=i+1
                else:
                        g.append(gl)
                        gl=[]
        g.append(gl)
        return g
group([1,3,2,2,9,7,2,4],3)
```

Out[3]:
```
[[1, 3, 2], [2, 9, 7], [2, 4]]
```

In [4]:
```
xs = ['dddd','a','bb','ccc']
sorted(xs, key=len)
```

Out[4]:
```
['a', 'bb', 'ccc', 'dddd']
```

In [5]:
```
import os
def extsort(x):
    i=0
    while(i<len(x)):
         x[i]=x[i].split('.')
         i=i+1
    x.sort(key=lambda x:x[1])
    i=0
    while(i<len(x)):
         x[i]="".join(x[i])
         i=i+1
    return x
print(extsort(['a.c', 'a.py', 'b.py', 'bar.txt', 'foo.txt', 'x.c']))
```

```
['a.c', 'x.c', 'a.py', 'b.py', 'bar.txt', 'foo.txt']
```

In [6]:
```
f= open("guru99.txt","w+")
#f=open("guru99.txt","a+")
for i in range(10):
    f.write("This is line %d\r\n" % (i+1))
f.close()
#Open the file back and read the contents
#f=open("guru99.txt", "r")
#   if f.mode == 'r':
#     contents =f.read()
#     print contents
#or, readlines reads the individual line into a list
#fl =f.readlines()
#for x in fl:
#print x
```

In [7]:
```
file = open("sample.txt", "r")

number_of_lines = 0
number_of_words = 0
number_of_characters = 0
for line in file:
  line = line.strip("\n")

  words = line.split()
  number_of_lines += 1
  number_of_words += len(words)
  number_of_characters += len(line)

file.close()

print("lines:", number_of_lines, "words:", number_of_words, "characters:", number
```

```
lines: 3 words: 5 characters: 29
```

In [8]:
```
textfile = open("sample.txt")
lines = textfile.readlines()
for line in reversed(lines):
    print(line)
textfile.close()
```

```
Goodbye
Hello Again

Hello World
```

In [9]:
```
def revline(x):
    i=0
    z=len(open(x).readlines())
    rev=[None]*z
    f=open(x)
    while(i<z):
         rev[i]=f.readline()
         rev[i]=rev[i].strip()
         print(rev[i][::-1])
         i=i+1
revline("sample.txt")
```

```
dlroW olleH
niagA olleH
eybdooG
```

In [10]:
```
def wrap(filename,k):
 f=open(filename).readlines()
 for i in f:
  new=i
  while len(new)>k:
    print(new[:k])
    new=new[k:]
  print(new)
wrap("sample.txt",5)
```

```
Hello
 Worl
d

Hello
 Agai
n

Goodb
ye
```

In [11]:
```
def num (n) :
    return n * 2

lst = [x for x in range(10)]
x = map(num, lst)
print(list(x))
```

```
[0, 2, 4, 6, 8, 10, 12, 14, 16, 18]
```

In [12]:
```
even_squares = [x * x for x in range(10) if x % 2 == 0]
print(even_squares)
```

```
[0, 4, 16, 36, 64]
```

In [13]:
```
def triplets(n):
    return [(a,c-a,c) for c in range(2,n) for a in range(1,c//2+1)]
triplets(4)
```

Out[13]:
```
[(1, 1, 2), (1, 2, 3)]
```

In [14]:
```
import csv

with open('employee_birthday.txt') as csv_file:
    csv_reader = csv.reader(csv_file, delimiter=',')
    line_count = 0
    for row in csv_reader:
        if line_count == 0:
            print(f'Column names are {", ".join(row)}')
            line_count += 1
        else:
            print(f'\t{row[0]} works in the {row[1]} department, and was born in
            line_count += 1
    print(f'Processed {line_count} lines.')
```

```
Column names are name, department, birthday month
        John Smith works in the Accounting department, and was born in Novembe
r.
        Erica Meyers works in the IT department, and was born in March.
Processed 3 lines.
```

In [15]:
```
import csv

with open('employee_birthday.txt') as csv_file:
    csv_reader = csv.reader(csv_file, delimiter=',')
    line_count = 0
    for row in csv_reader:
        if line_count == 0:
            print(f'Column names are {", ".join(row)}')
            line_count += 1
        else:
            print(f'\t{row[0]} works in the {row[1]} department, and was born in
            line_count += 1
    print(f'Processed {line_count} lines.')
```

```
Column names are name, department, birthday month
        John Smith works in the Accounting department, and was born in Novembe
r.
        Erica Meyers works in the IT department, and was born in March.
Processed 3 lines.
```

In [16]:
```
def mutate(d):
    ret=[d]
    i=0
    l=len(d)
    alp=map(chr,range(97,123))

    while i<l:
        cop=d
        ret.append(cop[:i]+cop[i+1:])
        if i<l-2:
            ret.append(cop[:i]+cop[i+1]+cop[i]+cop[i+2:])
        elif i<l-1:
            ret.append(cop[:i]+cop[i+1]+cop[i])
        for x in alp:
            ret.append(cop[:i]+x+cop[i+1:])
        for x in alp:
            ret.append(d+x)
            ret.append(x+d)
            ret.append(cop[:i]+x+cop[i:])
        i=i+1
    return ret
print('hefllo') in mutate('hello')
print('hllo') in mutate('hello')
```

```
hefllo
hllo
```

Out[16]:
```
False
```

In [17]:
```
def nearly_equal(str1,str2):
    count=0
    i=j=0
    while(i<len(str1) and j<len(str2)):
        if(str1[i]!=str2[j]):
            count=count+1
            if(len(str1)>len(str2)):
                i=i+1
            elif(len(str1)==len(str2)):
                pass
            else:
                i=i-1
        if(count>1):
            return False
        i=i+1
        j=j+1
    if(count<2):
            return True

str1=input("Enter first string::\n")
str2=input("Enter second string::\n")
boolean=nearly_equal(str1,str2)
if(boolean):
    print("Strings are nearly equal.")
else:
    print("Strings are not equal.")
```

```
Enter first string::
rise
Enter second string::
risee
Strings are nearly equal.
```

In [18]:
```
f="sample.txt"
file = open ( f, "r" )
a=[]
b={}
for i in file:
    for j in range(0,len(i)):
        a.append(i[j])
for i in a:
    if i in b:
        b[i]+=1
    else:
        b[i]=1
print(b)
c=f.split(".")
if c[1]=="txt":
    print("\n\nIt is a text file.")
elif c[1]=="cpp":
    print("\n\nIt is a c++ file.")
else:
    print("\n\nIt is a c file.")
```

```
{'H': 2, 'e': 3, 'l': 5, 'o': 5, ' ': 2, 'W': 1, 'r': 1, 'd': 2, '\n': 2, 'A':
1, 'g': 1, 'a': 1, 'i': 1, 'n': 1, 'G': 1, 'b': 1, 'y': 1}


It is a text file.
```

In [19]:
```
def anagrams(x):
        from itertools import permutations
        s={}
        while len(x)>0:
                x1=x.pop()
                s[x1]=s.get(x1,[])
                s[x1].append(x1)
                i=0
                while i<len(x):
                        z1=x[i]
                        perm=[''.join(p) for p in permutations(x1)]
                        if z1 in perm:
                                x.remove(z1)
                                s[x1].append(z1)
                        else:i=i+1
        return s.values()
print(anagrams(['tae','souep','eat','ihba','node','peuos','ate','abhi','bhia','do
```

```
dict_values([['soupe', 'souep', 'peuos'], ['tea', 'tae', 'eat', 'ate'], ['don
e', 'node'], ['bhia', 'ihba', 'abhi']])
```