In [28]:
```
v=int(input())
f=0
for i in range(1,v+1):
    if(v%i==0):
        f+=1
if(f==2):
    print("prime number")
else:
    print("not a prime number")
```

```
8
not a prime number
```

In [6]:
```
n=int(input())
v=0
for i in range(1,n):
    if (n%i==0):
        v=v+i
print("factors sum is:",v)
if(v==n):
    print("perfect number")
else:
    print("not a perfect number")
```

```
6
factors sum is: 6
perfect number
```

In [4]:
```
s=int(input())
e=int(input())
b=q=0
print("Even Numbers")
for k in range(s,e+1):
    if(k%2==0):
        print(k,end=' ')
        b=b+k
        q=q+1
print("\n Even numbers sum=",b)
print("Even Numbers count",q)
```

```
5
6
Even Numbers
6
 Even numbers sum= 6
Even Numbers count 1
```

# while loop syntax

initialization while(condition): statements updation(increment/decrement)

In [5]:
```
# 1 to 10 number
i=1
while(i<=10):
    print(i,end=' ')
    i=i+2
```

```
1 3 5 7 9
```

In [6]:
```
 # 10 to 1 numbers
i=10
while(i>=1):
    print(i,end=' ')
    i=i-2
```

```
10 8 6 4 2
```

In [8]:
```
u=int(input())
rev=0
while(u>0):
    r=u%10 # 328%10=32
    rev=rev*10+r # 0*10+5=5
    u=u//10 # 328//10=32
print("Reverse=",rev)
```

```
328
Reverse= 823
```

In [15]:
```
# i/p:121
# o/p: Palindrome
t=int(input())
temp=t
h=0
while(t>0):
    m=t%10
    h=h*10+m
    t=t//10
if(temp==h):
    print("palindrome")
else:
    print("not palindrome")
```

```
345
not palindrome
```

In [19]:
```
# witn arguments with return value
a,b=int(input()),int(input())
def add1(a,b):
    return a+b
add1(a,b)
```

```
85
85
```

Out[19]:
```
170
```

In [21]:
```
# with arguments without return value
a,b=int(input()),int(input())
def add2(a,b):
    print(a+b)
add2(a,b)
```

```
45
78
123
```

In [26]:
```
# without arguments with return value
v,m=int(input()),int(input())
def add3():
    return v+m
add3()
```

```
5
7
```

Out[26]:
```
12
```

In [25]:
```
# without arguments withoutv return value
j,k=int(input()),int(input())
def add4():
    print(j+k)
add4()
```

```
23
30
53
```

In [31]:
```
n=int(input())
def even_odd(n):
    if(n%2==0):
        print("even number")
    else:
        print("odd number")
even_odd(n)
```

```
8
even number
```

In [9]:
```
def prime(z):
    f_c=0
    for i in range(1,z+1):
        if(z%i==0):
            f_c+=1
    if(f_c==2):
        return True
    else:
        return False
z=int(input())
prime(z)
```

```
45
```

Out[9]:
```
False
```

In [10]:
```
s1,e1=int(input()),int(input())
def prime_range(s1,e1):
    for j in range(s1,e1+1):
        if(prime(j)==True):
            print(j,end=' ')
prime_range(s1,e1)
```

```
100
200
101 103 107 109 113 127 131 137 139 149 151 157 163 167 173 179 181 191 193 197
199
```

In [27]:
```
def perfect(h):
    m=0
    for c in range(1,h):
        if(h%c==0):
            m+=c
    if(m==h):
        return True
    else:
        return False
h=int(input())
perfect(h)
```

```
6
```

Out[27]:
```
True
```

In [26]:
```
def perfect_range(ss1,ee1):
    for j in range(ss1,ee1+1):
        if(perfect(j)==True):
            print(j,end=' ')
ss1,ee1=int(input()),int(input())
perfect_range(ss1,ee1)
```

```
1
1000
6 28 496
```

# String

- Collection of characters

In [29]:
```
aa="welcome"
print(aa)
```

```
welcome
```

In [34]:
```
a1=input()
print(a1)
```

```
welcome
welcome
```

In [37]:
```
print(len(a1))
print(type(a1))
print(min(a1))
print(max(a1))
print(sorted(a1))
```

```
7
<class 'str'>
c
w
['c', 'e', 'e', 'l', 'm', 'o', 'w']
```

In [38]:
```
a1
```

Out[38]: 'welcome'

In [41]:
```
a1[0:3]
```

Out[41]: 'wel'

In [43]:
```
a1[3:7]
```

Out[43]: 'come'

In [45]:
```
a1[3:]
```

Out[45]: 'come'

In [46]:
```
a1[0::2]
```

Out[46]: 'wloe'

In [47]:
```
a1[0::3]
```

Out[47]:
```
'wce'
```

In [49]:
```
a1[::-1]
```

Out[49]:
```
'emoclew'
```

In [51]:
```
print(dir(str),sep=' ')
```

```
['__add__', '__class__', '__contains__', '__delattr__', '__dir__', '__doc__',
'__eq__', '__format__', '__ge__', '__getattribute__', '__getitem__', '__getnewa
rgs__', '__gt__', '__hash__', '__init__', '__init_subclass__', '__iter__', '__l
e__', '__len__', '__lt__', '__mod__', '__mul__', '__ne__', '__new__', '__reduce
__', '__reduce_ex__', '__repr__', '__rmod__', '__rmul__', '__setattr__', '__siz
eof__', '__str__', '__subclasshook__', 'capitalize', 'casefold', 'center', 'cou
nt', 'encode', 'endswith', 'expandtabs', 'find', 'format', 'format_map', 'inde
x', 'isalnum', 'isalpha', 'isascii', 'isdecimal', 'isdigit', 'isidentifier', 'i
slower', 'isnumeric', 'isprintable', 'isspace', 'istitle', 'isupper', 'join',
'ljust', 'lower', 'lstrip', 'maketrans', 'partition', 'removeprefix', 'removesu
ffix', 'replace', 'rfind', 'rindex', 'rjust', 'rpartition', 'rsplit', 'rstrip',
'split', 'splitlines', 'startswith', 'strip', 'swapcase', 'title', 'translate',
'upper', 'zfill']
```

In [52]:
```
# captialize()
d="welcome"
d.capitalize()
```

Out[52]:
```
'Welcome'
```

In [58]:
```
# Count()
d.count("o")
```

Out[58]:
```
1
```

In [55]:
```
# upper(),lower()
k1="PYTHON"
print(k1.lower())
k2="python"
print(k2.upper())
```

```
python
PYTHON
```

In [59]:
```
# index()
d1="python programming"
print(d1.index('o'))
print(d1.rindex('o'))
```

```
4
9
```

In [63]:
```
# find()
print(d1.find("o"))
print(d1.find("z"))
print(d1.rfind("o"))
print(d1.rfind("s"))
```

```
4
-1
9
-1
```

In [1]:
```
h="ENGLISH"
h1="english"
print(h.isupper())
print(h1.islower())
```

```
True
True
```

In [2]:
```
h2="ENgLiSh"
h2.swapcase()
```

Out[2]:
```
'enGlIsH'
```

In [10]:
```
# isalpha(),isalnum(),isdigit(),isspace()
b="jupiter"
print(b.isalpha())
b1="jupiter1234"
print(b1.isalnum())
print(b1.isdigit())
n=""
print(n.isspace())
```

```
True
True
False
False
```

In [8]:
```
h="he@@"
h.isalnum()
```

Out[8]:
```
False
```

In [11]:
```
# startswith(),endswith()
f,l="hi hello gd evng","apssdc"
print(f.startswith("hi"))
print(f.startswith("g"))
print(l.endswith("c"))
```

```
True
False
True
```

In [12]:
```
# replace()
v1="day"
v1.replace("d","S")
```

Out[12]:
```
'Say'
```

In [13]:
```
# title()
m="good afternoon"
print(m.title())
print(m.istitle())
```

```
Good Afternoon
False
```

In [14]:
```
# split()
v="this is srk college located at vijayawada"
print(v.split())
```

```
['this', 'is', 'srk', 'college', 'located', 'at', 'vijayawada']
```

In [16]:
```
g="py@t h@on"
print(g.split())
print(g.split("@"))
```

```
['py@t', 'h@on']
['py', 't h', 'on']
```

In [18]:
```
# join()
print("#".join(g))
print("&".join(g))
```

```
p#y#@#t# #h#@#o#n
p&y&@&t& &h&@&o&n
```

In [19]:
```
# strip(),lstrip(),rstrip()
h1,h2,h3="    lenovo     ","     hiii","hello   "
h1.strip()
```

Out[19]:
```
'lenovo'
```

In [20]:
```
h2.lstrip()
```

Out[20]:
```
'hiii'
```

In [21]:
```
h3.rstrip()
```

Out[21]:
```
'hello'
```

In [25]:
```
# center()
fg="vanitha"
fg.center(20)
```

Out[25]:
```
'      vanitha       '
```

In [30]:
```
# zfill()
fg.zfill(20)
```

Out[30]:
```
'0000000000000vanitha'
```

In [36]:
```
# string iteration iterable
# var:value
#iterable contains more than value
students="siva sai" #8 chars
for ch in students:
    print(ch,end=" ")
```

```
s i v a   s a i
```

In [38]:
```
# index: to represent the values/chars present in str
for ix in range(len(students)):
    print(students[ix],end=" ")
```

```
s i v a   s a i
```

In [43]:
```
word="ApsSdC@123"
ii=""
for i in word:
    if(i.isupper()):
        ii=ii+i
print(ii)
```

```
ASC
```

In [41]:
```
w=input()
up=lw=dig=sp=""
for k in w:
    if(k.isdigit()):
        dig=dig+k
    elif(k.isupper()):
        up=up+k
    elif(k.islower()):
        lw=lw+k
    else:
        sp+=k
print("Uppercase Letters:",up)
print("Lowercase Letters:",lw)
print("Digits are:",dig)
print("Special Characters:",sp)
```

```
SIVAsai@@@23112001
Uppercase Letters: SIVA
Lowercase Letters: sai
Digits are: 23112001
Special Characters: @@@
```

In [ ]:
```

```