In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

21. Prime Number --> Whose factors are 1 and itself , it means fact count=2

2,3,5,7,11,13,17...etc

In [3]:
```
num=int(input("Enter number:"))

fact_count=0

for i in range(1,num+1):
    if num%i==0:
        fact_count+=1

if fact_count==2:
    print("Prime Number")
else:
    print("Not Prime")
```

executed in 2.41s, finished 15:34:46 2024-08-14

```
Enter number:13
Prime Number
```

22.Write a program to find the all the prime numbers in the given range.

In [5]:
```
start=int(input("Enter start:"))
end=int(input("Enter End:"))

count=0

for i in range(start,end+1):
    fact_count=0
    for j in range(1,i+1):
        if i%j==0:
            fact_count+=1
    if fact_count==2:
        count+=1
        print(i)

print("Count of prime numbers in the given range is:",count)
```

executed in 4.03s, finished 15:38:49 2024-08-14

```
Enter start:1
Enter End:100
2
3
5
7
11
13
17
19
23
29
31
37
41
43
47
53
59
61
67
71
73
79
83
89
97
Count of prime numbers in the given range is: 25
```

Controlling Keys for loops: (flow control statements)

1. break
2. continue
3. pass

1.break

- Used to forcefully exit from the loop once the condition is met.

In [6]:
```
for i in range(1,10):
    if i==5:
        break
    else:
        print(i)
```

executed in 8ms, finished 15:48:06 2024-08-14

```
1
2
3
4
```

2. continue

- Used for skipping the values.

In [7]:
```
for i in range(1,10):
    if i==5:
        continue
    else:
        print(i)
```

executed in 7ms, finished 15:49:13 2024-08-14

```
1
2
3
4
6
7
8
9
```

3.pass

- It is a null statement , it does nothing and acts as a placeholder for any block and used in the future purpose.

In [11]:
```
for i in range(1,10):
    pass
```

executed in 6ms, finished 15:51:31 2024-08-14

In [19]:
```
while True:
    num1=int(input("enter Val1:"))
    num2=int(input("Enter Val2:"))
    print("="*50)

    print("Options")
    print("1.Addition\n 2.Substraction\n 3.Multiplication\n 4.Division \n5.Exi


    print("="*50)

    option=int(input("enter Choice:"))

    if option==1:
        res=num1+num2
    elif option==2:
        res=num1-num2
    elif option==3:
        res=num1*num2
    elif option==4:
        if num2!=0:
            res=num1/num2
        else:
            res="Infinite"
            print("Denominator should not be zero")
    elif option==5:
        break

    print("result is {}".format(res))
```

executed in 31.7s, finished 16:01:34 2024-08-14

```
enter Val1:10
Enter Val2:20
==================================================
Options
1.Addition
 2.Substraction
 3.Multiplication
 4.Division
5.Exit
==================================================
enter Choice:1
result is 30
enter Val1:25
Enter Val2:36
==================================================
Options
1.Addition
 2.Substraction
 3.Multiplication
 4.Division
5.Exit
==================================================
enter Choice:3
result is 900
enter Val1:1
Enter Val2:1
==================================================
Options
1.Addition
 2.Substraction
 3.Multiplication
 4.Division
5.Exit
==================================================
enter Choice:1
result is 2
enter Val1:5
Enter Val2:42
==================================================
Options
1.Addition
 2.Substraction
 3.Multiplication
 4.Division
5.Exit
==================================================
enter Choice:5
```

Data Structures:

- A Varibale/object containing more than one value is called data structure.

> We have following Data structures:

1. Strings
2. List
3. Tuple
4. Set
5. Dictionary

1. Strings

- Collection of characters stored inside quotes '', "" or '''.

In [22]:
```
a="hyderabad"
b='hyderabad'

c=''' Hyderabad is
in India.'''
```

executed in 5ms, finished 16:08:40 2024-08-14

In [23]:
```
print(type(a),type(b),type(c))
```

executed in 7ms, finished 16:08:50 2024-08-14

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

Operations on Strings

1. Indexing : Extracting a particular character from string using its index values.
2. Slicing: Extracting a range of characters using its index values.

In [26]:
```
s="Hyderabad"

print(s[0])
print(s[8])
```

executed in 8ms, finished 16:11:30 2024-08-14

```
H
d
```

In [38]:
```
s="Hyderabad"

print(s[0:4])
print(s[0:4:2])
print(s[::])
print(s[0::2])
print(s[-1:-8:-1])
print(s[::-1])
print(s[-4:-8:-1])
```

executed in 13ms, finished 16:18:44 2024-08-14

```
Hyde
Hd
Hyderabad
Hdrbd
dabared
dabaredyH
ared
```

In [36]:
```
s=input("Enter string:")

if s==s[::-1]:
    print("Palindrome")
else:
    print("Not Palindrome")
```

executed in 1.34s, finished 16:16:03 2024-08-14

```
Enter string:bro
Not Palindrome
```

Inbuilt Functions

1. Capitalize()

In [44]:
```
s="the quick brown fox jumps over the lazy dog"

x=s.capitalize()
print(x)
print(s)
```

executed in 7ms, finished 16:22:36 2024-08-14

```
The quick brown fox jumps over the lazy dog
the quick brown fox jumps over the lazy dog
```

2. upper()

In [45]:
```
s="the quick brown fox jumps over the lazy dog"

print(s.upper())
```

executed in 7ms, finished 16:23:29 2024-08-14

```
THE QUICK BROWN FOX JUMPS OVER THE LAZY DOG
```

3.lower()

In [46]:
```
s="THE QUICK BROWN FOX JUMPS OVER THE LAZY DOG"
print(s.lower())
```

executed in 8ms, finished 16:23:51 2024-08-14

```
the quick brown fox jumps over the lazy dog
```

4.count()

In [49]:
```
s="the quick brown fox jumps over the lazy dog"

print(s.count('o'))
```

executed in 8ms, finished 16:25:59 2024-08-14

```
4
```

5. find()

In [50]:
```
s="the quick brown fox jumps over the lazy dog"

print(s.find('o'))
```

executed in 9ms, finished 16:29:12 2024-08-14

```
12
```

6.rfind()

In [52]:
```
s="the quick brown fox jumps over the lazy dog"

print(len(s))
print(s.rfind('o'))
```

executed in 9ms, finished 16:30:09 2024-08-14

```
43
41
```

7. index()

In [57]:
```
s="the quick brown fox jumps over the lazy dog"

print(s.index('o',13))
```
executed in 6ms, finished 16:32:05 2024-08-14

```
17
```

Imp Methods

1. split()

In [63]:
```
s="the quick brown fox jumps over the lazy dog"

print(s.split())
# s1=list(s)
# print(s1)
```
executed in 9ms, finished 16:36:38 2024-08-14

```
['t', 'h', 'e', ' ', 'q', 'u', 'i', 'c', 'k', ' ', 'b', 'r', 'o', 'w', 'n', '
', 'f', 'o', 'x', ' ', 'j', 'u', 'm', 'p', 's', ' ', 'o', 'v', 'e', 'r', ' ',
't', 'h', 'e', ' ', 'l', 'a', 'z', 'y', ' ', 'd', 'o', 'g']
```

email="sham@reddifmail.com (mailto:sham@reddifmail.com)" (from user input)

output= reddifmail

In [67]:
```
email="sham@reddifmail.com"

print(email.split("@")[1].split(".")[0])
```
executed in 8ms, finished 16:47:37 2024-08-14

```
reddifmail
```

In [68]:
```
emails=["dhana@gmail.com","sham@socialtek.in","ram@reddifmail.com"]

for i in emails:
    print(i.split("@")[1].split(".")[0])
```
executed in 11ms, finished 16:54:07 2024-08-14

```
gmail
socialtek
reddifmail
```

2. replace()

In [69]:
```
s="the quick brown fox jumps over the lazy dog"

s1=s.replace("dog","cat")
print(s1)
```
executed in 7ms, finished 16:55:01 2024-08-14

```
the quick brown fox jumps over the lazy cat
```

In [70]:
```
email="dhana@gmail.com"

print(email.replace("@","_"))
```
executed in 8ms, finished 16:55:43 2024-08-14

```
dhana_gmail.com
```

In [71]:
```
print(email)
```
executed in 6ms, finished 16:56:03 2024-08-14

```
dhana@gmail.com
```

3.strip()

In [74]:
```
s="the quick brown fox jumps over the lazy dog"
print(s)
s1="     the quick brown fox jumps over the lazy dog    "

print(s1.strip())
```
executed in 7ms, finished 16:57:10 2024-08-14

```
the quick brown fox jumps over the lazy dog
the quick brown fox jumps over the lazy dog
```

4. join()

In [77]:
```
s="the quick brown fox jumps over the lazy dog"

s1=s.split()
print(s1)

print(" ".join(s1))
```
executed in 9ms, finished 16:58:27 2024-08-14

```
['the', 'quick', 'brown', 'fox', 'jumps', 'over', 'the', 'lazy', 'dog']
the quick brown fox jumps over the lazy dog
```

logical functions

In [81]:
```
a="12325"

print(a.isnumeric())
```

executed in 5ms, finished 16:59:32 2024-08-14

```
False
```

In [86]:
```
b="sgvdjasg"

print(b.isalpha())
```

executed in 8ms, finished 17:00:20 2024-08-14

```
True
```

In [91]:
```
c="hkjgh4637"

print(c.isalnum())
```

executed in 6ms, finished 17:01:07 2024-08-14

```
False
```

24. Write a Python code to print the no. of digits in the given string

```
input - "Hello1 how2 8are6 you9."
output - 5
```

25. Write a program to print the words whose length is greater than k (length given by user) from a sentence.

```
Input : str = "string is fun in python"
k = 3
Output : string python
```

26. Write the code for the folowing:

```
input - "Hello How are You"
output - Hello-How-are-You
```

27. write the code for the following:

```
input - a="132,476,7.78327.8932,3"
output - "132.476.7,78327,8932.3"
```

28.Write the python code which converts the second half of the string to upper case letters

```
input - "nayankumar"
output -nayanKUMAR
```

29.Write a program which finds Maximum occurring Substring from test_list

```
test_str = "gfghsisbjknlmkesbestgfgsdcngfgcsdjnisdjnlbestdjsklgfgcdsbestbnjdsgfgdbhisbhsbestdkgfgb"
test_list = ['es',lm,'lgf,'gfg']
```

In [ ]: