Today Agenda

Questions

1. Read n space separated 3 digit values from user and find the sum of the middle digits in it 123 789 765 345 908 765 389:sum---
2. write down major differences b/w tuple and list
3. difference b/w remove and pop
4. difference b/w remove and discard
5. what is the use of with keyword
6. what is the use of dictionary
7. i/p string:vijayawada;o/p-->v*****a

In [1]:

```
print(*'123,789,765,345,908,765,889',sep = "+")
```

```
1+2+3+,+7+8+9+,+7+6+5+,+3+4+5+,+9+0+8+,+7+6+5+,+8+8+9
```

In [2]:

```
lists are mutable
tuple are immutable
lists are represented by []
tuple are represented by ()

```

```
  Input In [2]
    lists are mutable
          ^
SyntaxError: invalid syntax
```

In [4]:

```
my_dict1{1.'ram',2.'raju'}
print{my_dict1}
```

```
  Input In [4]
    my_dict1{1.'ram',2.'raju'}
            ^
SyntaxError: invalid syntax
```

In [9]:

```
my_dict=dict('bhavana':9)
print(my_dict)
```

```
  Input In [9]
    my_dict=dict('bhavana':9)
                          ^
SyntaxError: invalid syntax
```

In [10]:

```
dir(tuple)
```

...

In [11]:

```
s="vijayawada"
print(s[0],"*********",s[9])
```

```
v ********* a
```

In [12]:

```
dict={1:'injure',2:'for,3:'game'}

```

```
  Input In [12]
    dict={1:'injure',2:'for,3:'game'}
                               ^
SyntaxError: invalid syntax
```

In [13]:

```
a=[23,33,44]
a.pop(2)
```

Out[13]:

```
44
```

In [27]:

```
a=input()
print(a[0]+"****"+a[-1])
```

```
vanitha
v****a
```

In [30]:

```
a=input()
print(a[0]+"###"+a[-1])
```

```
dhoni
d###i
```

In [31]:

```
my_dict={1:'kittu'}
print(my_dict)
```

```
{1: 'kittu'}
```

In [33]:

```
print(my_dict[1])
```

```
kittu
```

In [42]:

```
number=[1,3,6,9,5]
print(number)
number.remove(3)
print(number)

```

```
[1, 3, 6, 9, 5]
[1, 6, 9, 5]
```

In [ ]:

```
#
```

In [35]:

```
numbers=[1,6,7,9]
numbers.pop(0)
print(numbers)

```

```
[6, 7, 9]
```

In [43]:

```
dir(dict)
```

Out[43]:

```
['__class__',
 '__class_getitem__',
 '__contains__',
 '__delattr__',
 '__delitem__',
 '__dir__',
 '__doc__',
 '__eq__',
 '__format__',
 '__ge__',
 '__getattribute__',
 '__getitem__',
 '__gt__',
 '__hash__',
 '__init__',
 '__init_subclass__',
 '__ior__',
 '__iter__',
 '__le__',
 '__len__',
 '__lt__',
 '__ne__',
 '__new__',
 '__or__',
 '__reduce__',
 '__reduce_ex__',
 '__repr__',
 '__reversed__',
 '__ror__',
 '__setattr__',
 '__setitem__',
 '__sizeof__',
 '__str__',
 '__subclasshook__',
 'clear',
 'copy',
 'fromkeys',
 'get',
 'items',
 'keys',
 'pop',
 'popitem',
 'setdefault',
 'update',
 'values']
```

In [44]:

```python
dir(list)
```

Out[44]:

```
['__add__',
 '__class__',
 '__class_getitem__',
 '__contains__',
 '__delattr__',
 '__delitem__',
 '__dir__',
 '__doc__',
 '__eq__',
 '__format__',
 '__ge__',
 '__getattribute__',
 '__getitem__',
 '__gt__',
 '__hash__',
 '__iadd__',
 '__imul__',
 '__init__',
 '__init_subclass__',
```

In [45]:

```python
dir(tuple)
```

...

In [46]:

```python
#delitem
my_dict
```

Out[46]:

```
{1: 'kittu'}
```

In [47]:

```python
my_dict.pop(1)

```

Out[47]:

```
'kittu'
```

In [48]:

```python
my_dict
```

Out[48]:

```
{}
```

In [49]:

```python
dir(set)
```

Out[49]:

```
['__and__',
 '__class__',
 '__class_getitem__',
 '__contains__',
 '__delattr__',
 '__dir__',
 '__doc__',
 '__eq__',
 '__format__',
 '__ge__',
 '__getattribute__',
 '__gt__',
 '__hash__',
 '__iand__',
 '__init__',
 '__init_subclass__',
 '__ior__',
 '__isub__',
 '__iter__',
 '__ixor__',
 '__le__',
 '__len__',
 '__lt__',
 '__ne__',
 '__new__',
 '__or__',
 '__rand__',
 '__reduce__',
 '__reduce_ex__',
 '__repr__',
 '__ror__',
 '__rsub__',
 '__rxor__',
 '__setattr__',
 '__sizeof__',
 '__str__',
 '__sub__',
 '__subclasshook__',
 '__xor__',
 'add',
 'clear',
 'copy',
 'difference',
 'difference_update',
 'discard',
 'intersection',
 'intersection_update',
 'isdisjoint',
 'issubset',
 'issuperset',
 'pop',
 'remove',
 'symmetric_difference',
 'symmetric_difference_update',
 'union',
 'update']
```

In [53]:

```
a=[1,2,3,4,45]
a.remove(3)
print(a)
a.pop()
```

```
[1, 2, 4, 45]
```

Out[53]:

```
45
```

In [51]:

```
b=(4,5,6,7,8)
b
```

Out[51]:

```
(4, 5, 6, 7, 8)
```

In [59]:

```
a=input()
print("i am ",a)
```

```
siva
i am  siva
```

In [61]:

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

```
siv
siv

---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Input In [61], in <cell line: 3>()
      1 s=input()
      2 print(s)
----> 3 print(rev.s)

NameError: name 'rev' is not defined
```

In [65]:

```
my_dict1={1:'dee',2:'dharani'}
my_dict1[2]='sweety'
print(my_dict1)
```

```
{1: 'dee', 2: 'sweety'}
```

In [66]:

```
my_dict1.popitem()
```

Out[66]:

```
(2, 'sweety')
```

In [67]:

```
dir(dict)
```

Out[67]:

```
['__class__',
 '__class_getitem__',
 '__contains__',
 '__delattr__',
 '__delitem__',
 '__dir__',
 '__doc__',
 '__eq__',
 '__format__',
 '__ge__',
 '__getattribute__',
 '__getitem__',
 '__gt__',
 '__hash__',
 '__init__',
 '__init_subclass__',
 '__ior__',
 '__iter__',
 '__le__',
 '__len__',
 '__lt__',
 '__ne__',
 '__new__',
 '__or__',
 '__reduce__',
 '__reduce_ex__',
 '__repr__',
 '__reversed__',
 '__ror__',
 '__setattr__',
 '__setitem__',
 '__sizeof__',
 '__str__',
 '__subclasshook__',
 'clear',
 'copy',
 'fromkeys',
 'get',
 'items',
 'keys',
 'pop',
 'popitem',
 'setdefault',
 'update',
 'values']
```

In [69]:

```
dir()
```

...

In [1]:

```python
a=input()
print(a)

```

...

Conditional Statements

- if
- if else
- elif
- nested if

if statement syntax

---

if(condition): statements

In [3]:

```python
n = int(input())
if(n%2==0):
    print('Even')
```

```
9
```

Syntax for if else statement

---

if(condition): statement else: statement

In [6]:

```python
# i/p: 6
# i/p: 8
# 8 is big number
a=int(input())
b=int(input())
if(a>b):
    print(a,'is big')
else:
    print(b,'is big')
```

```
6
8
8 is big
```

In [8]:

```
# i/p: 15
# o/p: Not Eligible for vote
age=15
if age>=18:
    print("eligible for vote")
else:
    print("not eligible for vote")
```

```
not eligible for vote
```

In [10]:

```
# i/p:  (100-200)  ---> 178
# o/p: Exist
a=int(input("enter a number:"))
if(a>=100 and a<=200):
    print("Exist")
else:
    print("Does not exist")
```

```
enter a number:17
Does not exist
```

In [11]:

```
d1,d2,d3=int(input()),int(input()),int(input())
if(d1>d2 and d1>d3):
    print(d1,"is big")
elif(d2>d3):
    print(d2,"is big")
else:
    print(d3,"is big")
```

...

In [ ]:

```
90-100---> Excellent
80-89---> A grade
70-79---> B grade
60-69---> C grade
50-59---> D grade
below 49--- fail
```

In [15]:

```
marks=int(input())
if(90<=marks<=100):
    print("Excellent")
elif(80<=marks<=89):
    print("A Grade")
elif(70<=marks<=79):
    print("B Grade")
elif(60<=marks<=69):
    print("C Grade")
elif(50<=marks<=59):
    print("D Grade")
elif(1<=marks<=49):
    print("Fail")
else:
    print("Invalid Marks")
```

```
189
Invalid Marks
```

In [17]:

```
# i/p: u/U
# o/p; Vowel
ch=input()
vowels="AEIOUaeiou"
if ch in vowels:
    print('Vowel')
else:
    print("Consonant")
```

...

In [19]:

```
# I/P:  april
# o/p:  april has 30 days

# i/p: october
# o/p: october has 31 days
month=input()
a=["april","june","september","november"]
b=["january","march","may","july","august","october","december"]
if month in a:
    print(month,"has 30 days")
elif month in b:
    print(month,"has 31 days")
elif(month=="february"):
    print(month,"has 28 or 29 days")
else:
    print("Invalid Input")


# nested if structure
--------------------------
if(condition):
    if(condition):
        statements
    else:
        statements
else:
    statements
```

```
june
june has 30 days
```

In [ ]:

```
# I/P: 6
       EVEN
       10> ---> CUBE
       10< ----> SQUAEE--> 36

# I/P: 7
# o/p: ODD
```

In [23]:

```
n1 = int(input())
if(n1%2==0):
    print("Even")
    if(n1>10):
        print(n1**3)
    else:
        print(n1**2)
else:
    print("Odd")
```

...

Loops

1.for loop 2.while loop

for loop syntax

for value in range(start,end,stepcount): statements

In [25]:

```python
num=int(input())
for i in range(1,num+1,3):
    print(i,end=' ')
```

```
100
1 4 7 10 13 16 19 22 25 28 31 34 37 40 43 46 49 52 55 58 61 64 67 70 73 76 79 8
2 85 88 91 94 97 100
```

In [1]:

```python
num=int(input())
for i in range(1,num+1):
    if(i%2!=0):
        print(i,end=' ')
```

...

5 x 1 = 5 5 x 2 = 10 5 x 3 = 15 5 x 4 = 20 . . 5 x 10 = 50

In [2]:

```python
m=int(input())
for i in range(1,11):
    print(m,"x",i,"=",m*i)
```

...

In [3]:

```python
# factors
j = int(input())
for i in range(1,j+1):
    if(j%i==0):
        print(i,end=' ')
```

...

In [5]:

```python
# 5 = 1*2*3*4*5=120
k = int(input())
f=1
for j in range(1,k+1):
    f=f*j
print(f)
```

...

In [8]:

```
# 5*4*3*2*1
k1 = int(input())
f1=1
for k in range(k1,0,-1):
    f1=f1*k
print(f1)
```

```
5
120
```

In [ ]:

```
i/p:  7
o/p:  prime number

i/p: 6
o/p: perfect number

(1-200)
even numbers : 2 4 6 8 10 ...........200
even numbers total:
even numbers count:

```

In [ ]:

```

```