24BIT167 - VANDITA NAWANI AIM : To implement varous list operations in python, including random number generation, searching, filtering, stack and queue operations

Hardware and Software Requirements : Hardware 16GB RAM,Intel Processor, Software: Python (Version 3.x.), Google Colab

System Configration: Windows 11, Google Colab

Theory : List in python is mutuable sequence used to dissimilar data types. Various operations can be performed sorting, searching and sorting.

choose 5 odd numbers at random. Similarly create a list of 4 even integers. Replace the third element of odd integer with the list of 4 numbers, sort and print the list

```
import random
odd = [random.randrange(1,50,2) for a in range(5)]
print("Odd Numbers:", odd)
even = [random.randrange(2,50,2) for a in range(4)]
print("EVEN NUMBERS:", even)
odd[2] = even
print("AFTER REPLACEMENT:", odd)

flatten_lst = []
for i in odd:
  if isinstance(i,list):
    flatten_lst.extend(i)
  else:
    flatten_lst.append(i)

  print("FLATTENED LIST :", flatten_lst)

flatten_lst.sort()
print("SORTED LIST ", flatten_lst)
```

```
Odd Numbers: [13, 5, 21, 31, 49]
EVEN NUMBERS: [14, 26, 30, 24]
AFTER REPLACEMENT: [13, 5, [14, 26, 30, 24], 31, 49]
FLATTENED LIST : [13]
FLATTENED LIST : [13, 5]
FLATTENED LIST : [13, 5, 14, 26, 30, 24]
FLATTENED LIST : [13, 5, 14, 26, 30, 24, 31]
FLATTENED LIST : [13, 5, 14, 26, 30, 24, 31, 49]
SORTED LIST  [5, 13, 14, 24, 26, 30, 31, 49]
```

Generate 20 random numbers and store them in list. accept a number from user and print position of all occurences of it

```
import random
lst = [random.randint(1,30) for a in range (20)]
print("THE LIST IS :", lst)
num = int(input("ENTER A NUMBER FROM LIST :"))
position = [i for i in range(len(lst)) if lst[i]==num]
if position:
  print("the number",num, "is at the positions", position)
else :
  print("the number {num} is not found ")
```

```
THE LIST IS : [18, 20, 19, 6, 25, 28, 18, 7, 9, 8, 15, 13, 14, 13, 19, 8, 21, 12, 30, 14
ENTER A NUMBER FROM LIST :22
the number {num} is not found
```

Generate 50 numbers on random and put them in a list between 1-30. Remove duplicate values.

```
import random
lst = [random.randint(1,30) for a in range(50)]
print("THE LIST IS:", lst)
lst1 = list(set(lst))
print("AFTER REMOVAL OF DUPLICATES:", lst1)
```

```
', 5, 16, 23, 6, 23, 27, 21, 5, 8, 11, 24, 7, 18, 29, 16, 27, 29, 9, 8, 16, 3, 7, 24, 14,
24, 25, 26, 27, 29, 30]
```

Generate 30 random numbers and put them in a list. Create two more lists - one containing only +ve and _ve one

```
import random
lst = [random.randint(-20,20) for a in range(30)]
print("THE LIST IS :", lst)
pos = [x for x in lst if x>0]
neg = [x for x in lst if x<0]

print("POSITIVE LIST IS :", pos)
print("NEGATIVE LIST IS: ", neg)
```

```
THE LIST IS : [1, -11, 2, 15, -11, 9, -7, 3, -5, -18, 13, -14, -5, -7, -20, -14, -6, -13
POSITIVE LIST IS : [1, 2, 15, 9, 3, 13, 11, 4, 9, 14]
NEGATIVE LIST IS:  [-11, -11, -7, -5, -18, -14, -5, -7, -20, -14, -6, -13, -17, -6, -2,
```

a list containing 5 string, covert all to upper case

```
lst = ['book', 'pen', 'pencil', 'eraser', 'scale']
print("THE LIST IS:", lst)
upp = [x.upper() for x in lst]
print("UPPER CASE LIST:", upp)
```

```
THE LIST IS: ['book', 'pen', 'pencil', 'eraser', 'scale']
UPPER CASE LIST: ['BOOK', 'PEN', 'PENCIL', 'ERASER', 'SCALE']
```

Convert list of temperature in fahrenheit degrees to equivalent celcius degree

```
lst = [10,20,30,40,50,60]
print("THE TEMPERATURE IN FAHRENHEIT IS :", lst)
lst1 = [(x-32)*5/9 for x in lst]
print("THE CELCIUS LIST IS :", lst1)
```

```
THE TEMPERATURE IN FAHRENHEIT IS : [10, 20, 30, 40, 50, 60]
THE CELCIUS LIST IS : [-12.222222222222221, -6.666666666666667, -1.1111111111111112, 4.4
```

Write a menu driven code to implement stack data structure

```
stack = []
while True:
  print("operations")
  print("1. APPEND")
  print("2. POP")
  print("3. DISPLAY")
  print("4. EXIT ")
  ch = int(input("ENTER THE CHOISE FROM ABOVE OPERATIONS:"))
  if ch==1:
    ele = input("enter an element to be pushed:")
    stack.append(ele)
    print("ELEMENT PUSHED")
    print("THE LIST IS", stack)

  elif ch==2:
    if stack:
      print("POPPED ELEMENT:", stack.pop())
      print("THE LIST IS", stack)
    else:
      print("Stack is empty")

  elif ch ==3:
    print("THE STACK IS :", stack)

  elif ch ==4:
    break
  else:
    print("INVALID CHOICE")
```

```
operations
1. APPEND
2. POP
3. DISPLAY
4. EXIT
ENTER THE CHOISE FROM ABOVE OPERATIONS:2
Stack is empty
operations
1. APPEND
2. POP
3. DISPLAY
4. EXIT
ENTER THE CHOISE FROM ABOVE OPERATIONS:1
enter an element to be pushed:3
ELEMENT PUSHED
THE LIST IS ['3']
operations
1. APPEND
2. POP
3. DISPLAY
4. EXIT
ENTER THE CHOISE FROM ABOVE OPERATIONS:5
INVALID CHOICE
operations
1. APPEND
2. POP
3. DISPLAY
4. EXIT
ENTER THE CHOISE FROM ABOVE OPERATIONS:4
```

write a menu driven program to implemnt Queue data structure

```
que = []
while True:
  print("operations")
  print("1. Add elements :")
  print("2. Removal of element")
  print("3. Dispaly")
  print("4. EXIT")

  ch = int(input("ENTER A CHOICE FROM THE ABOVE OPERATIONS:"))
  if ch==1:
    ele  = input("ENTER A ELEMENT:")
    que.append(ele)
    print("THE QUEUE IS :", que)

  elif ch==2:
    if que:
```

```
    if que:
      ele  = input("ENTER A ELEMENT TO BE REMOVED:")
      print(que.pop(ele))
    else:
      print("THE LIST IS EMPTY")

  elif ch==3:
    print("THE QUEUE IS :", que)

  elif ch==4:
    break

  else:
    print("INVALID CHOICE:")
```

```
operations
1. Add elements :
2. Removal of element
3. Dispaly
4. EXIT
ENTER A CHOICE FROM THE ABOVE OPERATIONS:1
ENTER A ELEMENT:3
THE QUEUE IS : ['3']
operations
1. Add elements :
2. Removal of element
3. Dispaly
4. EXIT
ENTER A CHOICE FROM THE ABOVE OPERATIONS:3
THE QUEUE IS : ['3']
operations
1. Add elements :
2. Removal of element
3. Dispaly
4. EXIT
ENTER A CHOICE FROM THE ABOVE OPERATIONS:4
```

take two list of numbers. create third list of numbers for only those numbers from the list which are not the comprehension

```
lst1 = [1,2,3,4,5,6]
lst2 = [2,4,6,8,9,10]

lst3 = [x for x in lst1 if x not in lst2]
print("THE NEW LIST IS :", lst3)
```

```
THE NEW LIST IS : [1, 3, 5]
```