Python

Sort

Sorting a list

Do a bubble sort

Use built in functions

Bubble sort -- Sorting Lists

There is two loops
    Outer loop
    Inner loop

Use either while or for loop control

Details

Get number of elements using len

Outer loop = number of elements

Inner loop = number of elements
                      minus one

```python
def bubblesort(arr):
    outer = len(arr)
    inner = outer -1
    while outer > 0:
        index =0
        while index <= inner-1:
            if arr[index] > arr[index +1]:
                temp = arr[index]
                arr[index] = arr[index +1]
                arr[index+1]=temp
            index = index+1
        outer = outer-1
    return
```

```
def showarray(arr):
    print('List -> ',arr)
    return
def spacing(a):
    print('\n' * a)
    return
def printsep():
    print
    print('--------------------')
    print
    return
```

```
def main():
    spacing(30)
    arrmain = [11,44,77,2,9,15]
    printsep()
    showarray(arrmain)
    bubblesort(arrmain)
    printsep()
    showarray(arrmain)
    printsep()
    spacing(10)
    return
main()
```

Whole Program

```python
def bubblesort(arr):
    outer = len(arr)
    inner = outer -1
    while outer > 0:
        index =0
        while index <= inner-1:
            if arr[index] > arr[index +1]:
                temp = arr[index]
                arr[index] = arr[index +1]
                arr[index+1]=temp
            index = index+1
        outer = outer-1
    return
def showarray(arr):
    print('List -> ',arr)
    return
def spacing(a):
    print('\n' * a)
    return
```

```python
def printsep():
    print
    print('-------------------')
    print
    return
def main():
    spacing(30)
    arrmain = [11,44,77,2,9,15]
    printsep()
    showarray(arrmain)
    bubblesort(arrmain)
    printsep()
    showarray(arrmain)
    printsep()
    spacing(10)
    return
main()
```

Built in Sort

There are two built in functions that do sorting:

sort()

sorted()

sort()

This will sort the list in place default is ascending order. There is a parameter to do descending order.

Example:

Ascending: list.sort()

Descending: list.sort(reverse=True)

Try This

```python
def spacing(a):
    print('\n' *a)
    return
def sortit(arr):
    arr.sort()
    # arr.sort(reverse = True)
    return
def showarray(arr):
    print('list-. ',arr)
    return
def printsep():
    print
    print('-------------------')
    print
    return
```

```python
def main():
    spacing(40)
    arrmain = [11,44,77,2,9,15]
    printsep()
    showarray(arrmain)
    sortit(arrmain)
    printsep()
    showarray(arrmain)
    printsep()
    spacing(10)
    return
main()
```

sort11

sorted()

This will create a new sorted list, default is ascending order. There is a parameter to do descending order. The list to be sorted is enclosed in the parentheses.

Example:

Ascending: newlist = sorted(list)

Descending: newlist=sorted(list,reverse=True)

```python
def spacing(a):
    print('\n' * a)
    return
def sortit(arr):
    # brr=sorted(arr)
    brr=sorted(arr,reverse=True)
    return brr
```

```
def showarray(srr):
    y=len(srr)
    print('number of elements -> ',y)
    print(srr)
    return
def printsep():
    print
    print('--------------------')
    print
    return
```

```
def main():
    spacing(40)
    arrmain = [11,44,77,2,9,15]
    printsep()
    print('orginal list')
    showarray(arrmain)
    bb=sortit(arrmain)
    printsep()
    print('sorted list')
    showarray(bb)
    spacing(5)
    return
main()
```

sort12

Whole Program

```python
def spacing(a):
    print('\n' * a)
    return
def sortit(arr):
    # brr=sorted(arr)
    brr=sorted(arr,reverse=True)
    return brr
def showarray(srr):
    y=len(srr)
    print('number of elements -> ',y)
    print(srr)
    return
def printsep():
    print
    print('-------------------')
    print
    return
```

```python
def main():
    spacing(40)
    arrmain = [11,44,77,2,9,15]
    printsep()
    print('orginal list')
    showarray(arrmain)
    bb=sortit(arrmain)
    printsep()
    print('sorted list')
    showarray(bb)
    spacing(5)
    return
main()
```

done