Papers Dock

PYTHON

9618

BUBBLE SORT

Bubble Sort

It's an algorithm to arrange an array in either ascending or descending order

Swapping Variable

Incorrect

```
array = [ 9, 0 ]
array[0] = array[1]
array[1] = array[0]
```

Correct

```
array = [ 9, 0 ]
temp = array[0]
array[0] = array[1]
array[1] = temp
```

There are two loops in a Bubble Sort Algorithm.
One inner loop means that one value is at the correct position and is responsible for swapping of each element

array = [10, 5, 6, 3, 7, 8]

After one correct positioning of an element there will be another outer loop which basically determines the number of elements in an array

Difference Between Efficient and Inefficient code

Inefficient code

Uses both For Loops and performs extra unwanted loops

Efficient code

Outer loop is conditional loop with flag looping and inner loop is for loop

arrayData = [10, 5, 6, 7, 1, 12, 13, 15, 21, 8]

Write a Bubble Sort Code to sort the arrayData in ascending order

Inefficient

```
arrayData = [10, 5, 6, 7, 1, 12, 13, 15, 21, 8]


def bubblesort():
    for x in range(0, 10):
        for y in range(0, 9):
            if arrayData[y] > arrayData[y+1]:
                temp = arrayData[y]
                arrayData[y] = arrayData[y+1]
                arrayData[y+1] = temp
```

Efficient

```
arrayData = [10, 5, 6, 7, 1, 12, 13, 15, 21, 8]

def bubblesort():
    Noswaps = False
    Boundary = 9
    while Noswaps == False:
        Noswaps =True
        for y in range(0, Boundary):
            if arrayData[y] > arrayData[y+1]:
                temp = arrayData[y]
                arrayData[y] = arrayData[y+1]
                arrayData[y+1] = temp
                Noswaps = False
        Boundary = Boundary - 1
bubblesort()
print(arrayData)
```