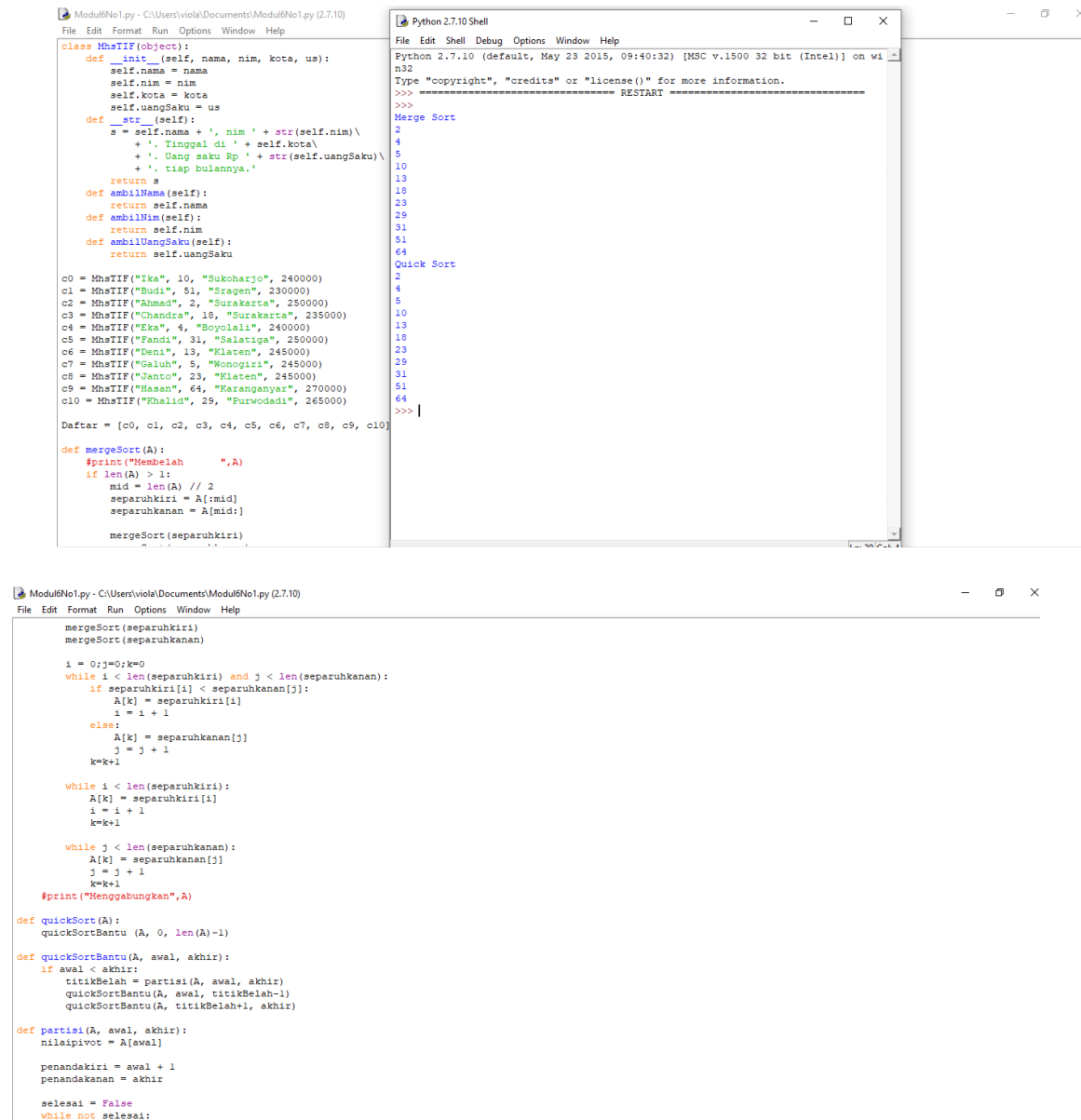


Nama : Viola Lovitasari
NIM : L200180173
Kelas : G
MODUL 6 PRAK ASD

1.



```
Modul6No1.py - C:\Users\viola\Documents\Modul6No1.py (2.7.10)
File Edit Format Run Options Window Help

class MhsTIF(object):
    def __init__(self, nama, nim, kota, us):
        self.nama = nama
        self.nim = nim
        self.kota = kota
        self.uangSaku = us
    def __str__(self):
        s = self.nama + ', ' + str(self.nim) + '\n'
        + 'Tinggal di ' + self.kota + '\n'
        + 'Uang saku Rp ' + str(self.uangSaku) + '\n'
        + 'tiap bulannya.'
        return s
    def ambilNama(self):
        return self.nama
    def ambilNim(self):
        return self.nim
    def ambilUangSaku(self):
        return self.uangSaku

c0 = MhsTIF("Ika", 10, "Sukoharjo", 240000)
c1 = MhsTIF("Budi", 51, "Sragen", 230000)
c2 = MhsTIF("Ahmad", 2, "Surakarta", 250000)
c3 = MhsTIF("Chandra", 18, "Surakarta", 235000)
c4 = MhsTIF("Eka", 4, "Soyolali", 240000)
c5 = MhsTIF("Fandi", 31, "Salatiga", 250000)
c6 = MhsTIF("Deni", 13, "Klaten", 245000)
c7 = MhsTIF("Galuh", 5, "Wonogiri", 245000)
c8 = MhsTIF("Janto", 23, "Klaten", 245000)
c9 = MhsTIF("Hasan", 64, "Karanganyar", 270000)
c10 = MhsTIF("Khalid", 29, "Purwodadi", 265000)

Daftar = [c0, c1, c2, c3, c4, c5, c6, c7, c8, c9, c10]

def mergeSort(A):
    #print("Memelah", A)
    if len(A) > 1:
        mid = len(A) // 2
        separuhkiri = A[:mid]
        separuhkanan = A[mid:]

        mergeSort(separuhkiri)
        mergeSort(separuhkanan)

        i = 0; j = 0; k = 0
        while i < len(separuhkiri) and j < len(separuhkanan):
            if separuhkiri[i] < separuhkanan[j]:
                A[k] = separuhkiri[i]
                i = i + 1
            else:
                A[k] = separuhkanan[j]
                j = j + 1
            k = k + 1

        while i < len(separuhkiri):
            A[k] = separuhkiri[i]
            i = i + 1
            k = k + 1

        while j < len(separuhkanan):
            A[k] = separuhkanan[j]
            j = j + 1
            k = k + 1

        #print("Menggabungkan", A)

def quickSort(A):
    quickSortBantu(A, 0, len(A)-1)

def quickSortBantu(A, awal, akhir):
    if awal < akhir:
        titikBelah = partisi(A, awal, akhir)
        quickSortBantu(A, awal, titikBelah-1)
        quickSortBantu(A, titikBelah+1, akhir)

def partisi(A, awal, akhir):
    nilaiPivot = A[awal]
    penandakiri = awal + 1
    penandakanan = akhir

    selesai = False
    while not selesai:
        while i < len(separuhkiri):
            A[k] = separuhkiri[i]
            i = i + 1
            k = k + 1

        while j < len(separuhkanan):
            A[k] = separuhkanan[j]
            j = j + 1
            k = k + 1

        #print("Menggabungkan", A)

def quickSort(A):
    quickSortBantu(A, 0, len(A)-1)

def quickSortBantu(A, awal, akhir):
    if awal < akhir:
        titikBelah = partisi(A, awal, akhir)
        quickSortBantu(A, awal, titikBelah-1)
        quickSortBantu(A, titikBelah+1, akhir)

def partisi(A, awal, akhir):
    nilaiPivot = A[awal]
    penandakiri = awal + 1
    penandakanan = akhir

    selesai = False
    while not selesai:
```

```

while not selesai:
    while penandakiri <= penandakanan and A[penandakiri] <= nilaipivot:
        penandakiri = penandakiri + 1
    while penandakanan >= penandakiri and A[penandakanan] >= nilaipivot:
        penandakanan = penandakanan - 1
    if penandakanan < penandakiri:
        selesai = True
    else:
        temp = A[penandakiri]
        A[penandakiri] = A[penandakanan]
        A[penandakanan] = temp
    temp = A[awal]
    A[awal] = A[penandakanan]
    A[penandakanan] = temp
    return penandakanan

def convert(arr, obj):
    hasil=[]
    for x in range (len(arr)):
        for i in range (len(obj)):
            if arr[x] == obj[i].nim:
                hasil.append(obj[i])
    return hasil

def urutkanQuick(arr):
    A = []
    for x in Daftar:
        A.append(x.nim)
    print("Quick Sort")
    quickSort(A)
    for x in convert(A, Daftar):
        print(x.nim)

def urutkanMerge(arr):
    A = []

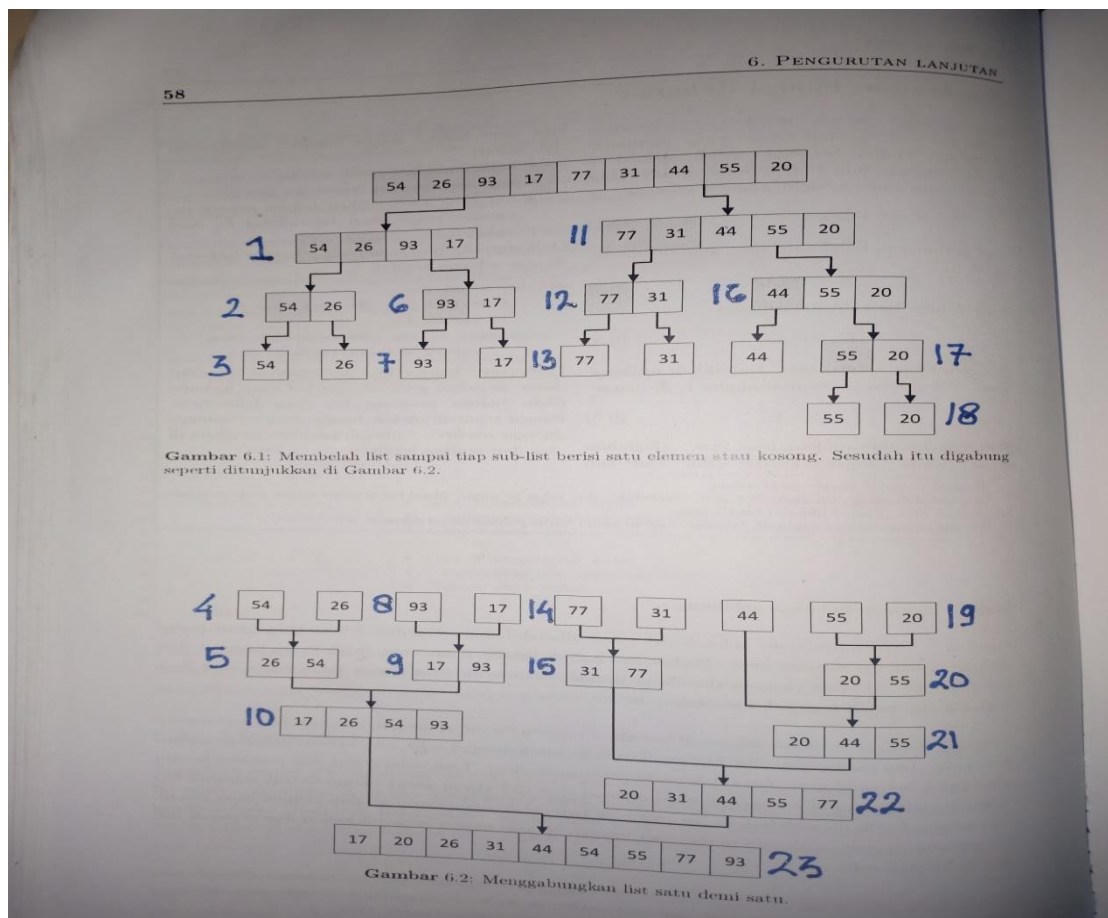
```

```

    urutkanMerge(Daftar)
    urutkanQuick(Daftar)

```

2.



3.

```

Modul6No3.py - C:\Users\viola\Documents\Modul6No3.py (2.7.10)
File Edit Format Run Options Window Help

while penandakanan >= penandakiri and A[penandakanan] >= milaipivot:
    penandakanan = penandakanan - 1

if penandakanan < penandakiri:
    selesai = True
else:
    temp = A[penandakiri]
    A[penandakiri] = A[penandakanan]
    A[penandakanan] = temp

temp = A[awal]
A[awal] = A[penandakanan]
A[penandakanan] = temp

return penandakanan

def quickSortBantu(A, awal, akhir):
    if awal < akhir:
        titikBelah = partisi(A, awal, akhir)
        quickSortBantu(A, awal, titikBelah-1)
        quickSortBantu(A, titikBelah+1, akhir)

def quickSort(A):
    quickSortBantu(A, 0, len(A)-1)

daftar = [10, 51, 2, 18, 4, 31, 13, 5, 23, 64, 29]

k = [[i] for i in range(1, 6001)]
Kocok(k)
u_bub = k[:]
u_sel = k[:]
u_ins = k[:]
u_mrg = k[:]
u_qck = k[:]

aw=detak();bubbleSort(u_bub);ak=detak();print("bubble: %g detik" %(ak-aw));
aw=detak();selectionSort(u_sel);ak=detak();print("selection: %g detik" %(ak-aw));
aw=detak();insertionSort(u_ins);ak=detak();print("insertion: %g detik" %(ak-aw));
aw=detak();mergeSort(u_mrg);ak=detak();print("merge: %g detik" %(ak-aw));
aw=detak();quickSort(u_qck);ak=detak();print("quick: %g detik" %(ak-aw));
  
```

```

Python 2.7.10 Shell
File Edit Shell Debug Options Window Help
Python 2.7.10 (default, May 23 2015, 09:40:32) [MSC v.1500 32 bit (Intel)] on wi
n32
Type "copyright", "credits" or "license()" for more information.
>>> ===== RESTART =====
>>>
bubble: 11.106 detik
selection: 4.731 detik
insertion: 4.613 detik
merge: 0.0839999 detik
quick: 0.0510001 detik
>>> |
  
```

4. a) Merge sort

L = [80, 7, 24, 16, 43, 91, 35, 2, 19, 72]

80	7	24	16	43	91	35	2	19	72
----	---	----	----	----	----	----	---	----	----

Proses 1

7	80	26	24	43	91	2	35	19	72
---	----	----	----	----	----	---	----	----	----

Proses 2

7	16	24	80	2	35	43	91	19	72
---	----	----	----	---	----	----	----	----	----

Proses 3

2	7	16	24	35	43	80	91	19	72
---	---	----	----	----	----	----	----	----	----

Proses 4

2	7	16	19	24	35	43	72	80	91
---	---	----	----	----	----	----	----	----	----

b) Quick sort

L = [80, 7, 24, 16, 43, 91, 35, 2, 19, 72]

80	7	24	16	43	91	35	2	19	72
----	---	----	----	----	----	----	---	----	----

Pivot

80	7	24	16	43	91	35	2	19	72
----	---	----	----	----	----	----	---	----	----

Low

High

Pivot

72	7	24	16	43	91	35	2	19	80
----	---	----	----	----	----	----	---	----	----

Low

High

Pivot

72	7	24	16	43	91	35	2	19	80
----	---	----	----	----	----	----	---	----	----

Low

High

Pivot

72	7	24	16	43	80	35	2	19	91
----	---	----	----	----	----	----	---	----	----

Low

High

Pivot

72	7	24	16	43	19	35	2	80	91
----	---	----	----	----	----	----	---	----	----

Low

High

Pivot

72	7	24	16	43	19	35	2	80	91
----	---	----	----	----	----	----	---	----	----

Low

High

Pivot

2	7	24	16	43	19	35	72	80	91
---	---	----	----	----	----	----	----	----	----

Low

High

Pivot

2	7	24	16	43	19	35	72	80	91
---	---	----	----	----	----	----	----	----	----

Low

High

Pivot

2	7	24	16	43	19	35	72	80	91
---	---	----	----	----	----	----	----	----	----

Low

High

Pivot

2	7	24	16	43	19	35	72	80	91
---	---	----	----	----	----	----	----	----	----

Low

High

Pivot

2	7	24	16	43	19	35	72	80	91
---	---	----	----	----	----	----	----	----	----

Low

High

Pivot

2	7	19	16	43	24	35	72	80	91
---	---	----	----	----	----	----	----	----	----

Low

High

Pivot

2	7	19	16	43	24	35	72	80	91
---	---	----	----	----	----	----	----	----	----

Low

High

Pivot

2	7	19	16	24	43	35	72	80	91
---	---	----	----	----	----	----	----	----	----

Low

High

Pivot

2	7	19	16	24	43	35	72	80	91
---	---	----	----	----	----	----	----	----	----

Low

High

Pivot

2	7	16	19	24	43	35	72	80	91
---	---	----	----	----	----	----	----	----	----

Low

High

Pivot

2	7	16	19	24	43	35	72	80	91
---	---	----	----	----	----	----	----	----	----

Low

High

Pivot

2	7	16	19	24	35	43	72	80	91
---	---	----	----	----	----	----	----	----	----

Low

High

2	7	16	19	24	35	43	72	80	91
---	---	----	----	----	----	----	----	----	----

5.

Modul6No5.py - C:\Users\viola\Documents\Modul6No5.py (2.7.10)

File Edit Format Run Options Window Help

```
class MhsTIF(object):
    def __init__(self, nama, nim, kota, us):
        self.nama = nama
        self.nim = nim
        self.kota = kota
        self.uang = us
    def __str__(self):
        s = self.nama + " " + str(self.nim) + " " + self.kota + " " + str(self.uang)
        return s
    def ambilNama(self):
        return self.nama
    def ambilNim(self):
        return self.nim
    def ambilUangSaku(self):
        return self.uang

c0 = MhsTIF("Ika", 10, "Sukoharjo", 240000)
c1 = MhsTIF("Budi", 51, "Sragen", 230000)
c2 = MhsTIF("Ahmad", 2, "Surakarta", 250000)
c3 = MhsTIF("Chandra", 18, "Surakarta", 235000)
c4 = MhsTIF("Eka", 4, "Boyolali", 240000)
c5 = MhsTIF("Fandi", 31, "Salatiga", 250000)
c6 = MhsTIF("Deni", 13, "Klaten", 245000)
c7 = MhsTIF("Galuh", 5, "Wonogiri", 245000)
c8 = MhsTIF("Janto", 23, "Klaten", 245000)
c9 = MhsTIF("Hasan", 64, "Karanganyar", 270000)
c10 = MhsTIF("Khalid", 29, "Purwodadi", 265000)

Daftar = [c0, c1, c2, c3, c4, c5, c6, c7, c8, c9, c10]

def cetak(A):
    for i in A:
        print (i)

def mergeSort2(A, awal, akhir):
    mid = (awal+akhir)//2
    if awal < akhir:
        mergeSort2(A, awal, mid)
        mergeSort2(A, mid+1, akhir)
```

Python 2.7.10 Shell

File Edit Shell Debug Options Window Help

```
Python 2.7.10 (default, May 23 2015, 09:40:32) [MSC v.1500 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> ===== RESTART =====
>>>
Sebelum diurutkan
Ika 10 Sukoharjo 240000
Budi 51 Sragen 230000
Ahmad 2 Surakarta 250000
Chandra 18 Surakarta 235000
Eka 4 Boyolali 240000
Fandi 31 Salatiga 250000
Deni 13 Klaten 245000
Galuh 5 Wonogiri 245000
Janto 23 Klaten 245000
Hasan 64 Karanganyar 270000
Khalid 29 Purwodadi 265000

Setelah diurutkan
Budi 51 Sragen 230000
Chandra 18 Surakarta 235000
Eka 4 Boyolali 240000
Ika 10 Sukoharjo 240000
Janto 23 Klaten 245000
Galuh 5 Wonogiri 245000
Deni 13 Klaten 245000
Fandi 31 Salatiga 250000
Ahmad 2 Surakarta 250000
Khalid 29 Purwodadi 265000
Hasan 64 Karanganyar 270000
>>> |
```

Modul6No5.py - C:\Users\viola\Documents\Modul6No5.py (2.7.10)

File Edit Format Run Options Window Help

```
print (i)

def mergeSort2(A, awal, akhir):
    mid = (awal+akhir)//2
    if awal < akhir:
        mergeSort2(A, awal, mid)
        mergeSort2(A, mid+1, akhir)

    a, f, l = 0, awal, mid+1
    tmp = [None] * (akhir - awal + 1)
    while f <= mid and l <= akhir:
        if A[f].ambilUangSaku() < A[l].ambilUangSaku():
            tmp[a] = A[f]
            f += 1
        else:
            tmp[a] = A[l]
            l += 1
        a += 1

    if f <= mid:
        tmp[a:] = A[f:mid+1]

    if l <= akhir:
        tmp[a:] = A[l:akhir+1]

    a = 0
    while awal <= akhir:
        A[awal] = tmp[a]
        awal += 1
        a += 1

def mergeSort(A):
    mergeSort2(A, 0, len(A)-1)

print("Sebelum diurutkan")
cetak(Daftar)
mergeSort(Daftar)
print("\nSetelah diurutkan")
cetak(Daftar)
```

6.

No 6.py - C:\Users\viola\Documents\Modul 6\No 6.py (2.7.10)
File Edit Format Run Options Window Help

```

class MhsTIF():
    def __init__(self, nama, nim, kota, us):
        self.nama = nama
        self.nim = nim
        self.kota = kota
        self.us = us

    def __str__(self):
        s = self.nama + ', NIM ' + str(self.nim) \
            + '. Tinggal di ' + self.kota \
            + '. Uang saku Rp. ' + str(self.us) \
            + ' tiap bulannya.'
        return s

    def ambilNama(self):
        return self.nama
    def ambilNim(self):
        return self.nim
    def ambilUangSaku(self):
        return self.us

c0 = MhsTIF("Ika", 10, "Sukoharjo", 240000)
c1 = MhsTIF("Budi", 51, "Sragen", 230000)
c2 = MhsTIF("Ahmad", 2, "Surakarta", 250000)
c3 = MhsTIF("Chandra", 18, "Surakarta", 235000)
c4 = MhsTIF("Eka", 4, "Boyolali", 240000)
c5 = MhsTIF("Fandi", 31, "Salatiga", 250000)
c6 = MhsTIF("Fendi", 13, "Klaten", 245000)
c7 = MhsTIF("Galuh", 5, "Wonogiri", 245000)
c8 = MhsTIF("Janto", 23, "Klaten", 245000)
c9 = MhsTIF("Hasan", 64, "Karanganyar", 270000)
c10 = MhsTIF("Khalid", 29, "Purwodadi", 265000)

Daftar = [c0, c1, c2, c3, c4, c5, c6, c7, c8, c9, c10]
A = []
for i in Daftar:
    A.append(i.nama)

def cetak():
    for i in A:
        print(i)

```

Python 2.7.10 Shell
File Edit Shell Debug Options Window Help

```

Python 2.7.10 (default, May 23 2015, 09:40:32) [MSC v.1500 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> -----RESTART-----
>>>
Sebelum diurutkan
Ika
Budi
Ahmad
Chandra
Eka
Fandi
Deni
Galuh
Janto
Hasan
Khalid
>>>
Setelah diurutkan
Ika
Budi
Ahmad
Chandra
Eka
Fandi
Deni
Galuh
Janto
Hasan
Khalid
>>>

```

No 6.py - C:\Users\viola\Documents\Modul 6\No 6.py (2.7.10)
File Edit Format Run Options Window Help

```

c0 = MhsTIF("Fandi", 31, "Salatiga", 250000)
c5 = MhsTIF("Fendi", 13, "Klaten", 245000)
c6 = MhsTIF("Deni", 13, "Klaten", 245000)
c7 = MhsTIF("Galuh", 5, "Wonogiri", 245000)
c8 = MhsTIF("Janto", 23, "Klaten", 245000)
c9 = MhsTIF("Hasan", 64, "Karanganyar", 270000)
c10 = MhsTIF("Khalid", 29, "Purwodadi", 265000)

Daftar = [c0, c1, c2, c3, c4, c5, c6, c7, c8, c9, c10]
A = []
for i in Daftar:
    A.append(i.nama)

def cetak():
    for i in A:
        print(i)

def quickSort(arr):
    kurang = []
    pivotList = []
    lebih = []
    if len(arr) <= 1:
        return arr
    else:
        pivot = arr[0]
        for i in arr:
            if i < pivot:
                kurang.append(i)
            elif i > pivot:
                lebih.append(i)
            else:
                pivotList.append(i)
        kurang = quickSort(kurang)
        lebih = quickSort(lebih)
        return kurang + pivotList + lebih

print("Sebelum diurutkan")
cetak()
print("\nSetelah diurutkan")
quickSort(A)
cetak()

```

7.

Modul8No7.py - C:\Users\viola\Documents\Modul8No7.py (2.7.10)
File Edit Format Run Options Window Help

```

mergeSort2(A, 0, len(A)-1)

def quickSortNew(arr):
    kurang = []
    pivotList = []
    lebih = []
    if len(arr) <= 1:
        return arr
    else:
        pivot = arr[0]
        for i in arr:
            if i < pivot:
                kurang.append(i)
            elif i > pivot:
                lebih.append(i)
            else:
                pivotList.append(i)
        kurang = quickSortNew(kurang)
        lebih = quickSortNew(lebih)
        return kurang + pivotList + lebih

k = list(range(6000))
kacak(k)
u_bub = k[:]
u_sel = k[:]
u_ins = k[:]
u_mrg = k[:]
u_qck = k[:]
u_mrgNew = k[:]
u_qckNew = k[:]

aw = detak();bubbleSort(u_bub);ak = detak();print('bubble : %g detik' %(ak-aw));
aw = detak();selectionSort(u_sel);ak = detak();print('selection : %g detik' %(ak-aw));
aw = detak();insertionSort(u_ins);ak = detak();print('insertion : %g detik' %(ak-aw));
aw = detak();mergeSort(u_mrg);ak = detak();print('merge: %g detik' %(ak-aw));
aw = detak();quickSort(u_qck);ak = detak();print("quick: %g detik" %(ak-aw));
aw = detak();mergeSortNew(u_mrgNew);ak = detak();print("merge New: %g detik" %(ak-aw));
aw = detak();quickSortNew(u_qckNew);ak = detak();print("quick New: %g detik" %(ak-aw));

```

Python 2.7.10 Shell
File Edit Shell Debug Options Window Help
Python 2.7.10 (default, May 23 2015, 09:40:32) [MSC v.1500 32 bit (Intel)] on wi
n32
Type "copyright", "credits" or "license()" for more information.
>>> ===== RESTART =====
>>>
bubble : 9.427 detik
selection : 2.673 detik
insertion : 3.889 detik
merge: 0.102 detik
quick: 0.0510001 detik
merge New: 0.121 detik
quick New: 0.04 detik
>>>

8.

Modul8No8.py - C:\Users\viola\Documents\Modul8No8.py (2.7.10)
File Edit Format Run Options Window Help

```

class Node():
    def __init__(self, data, tautan=None):
        self.data = data
        self.tautan = tautan

def cetak(head):
    curr = head
    while curr is not None:
        try:
            print (curr.data)
            curr = curr.tautan
        except:
            pass

a = Node(1)
b = Node(3)
c = Node(5)
d = Node(7)
e = Node(2)
f = Node(4)
g = Node(6)

a.tautan = b
b.tautan = c
c.tautan = d
d.tautan = e
e.tautan = f
f.tautan = g

def mergeSortLL(A):
    linked = A
    try:
        daftar = []
        curr = A
        while curr:
            daftar.append(curr.data)
            curr = curr.tautan
        A = daftar
    except:
        A = A

```

Python 2.7.10 Shell
File Edit Shell Debug Options Window Help
Python 2.7.10 (default, May 23 2015, 09:40:32) [MSC v.1500 32 bit (Intel)] on wi
n32
Type "copyright", "credits" or "license()" for more information.
>>> ===== RESTART =====
>>>
1
2
3
4
5
6
7
>>>


```
except:
    A = A

if len(A) > 1:
    mid = len(A) // 2
    separuhkiri = A[:mid]
    separuhkanan = A[mid:]

    mergeSortLL(separuhkiri)
    mergeSortLL(separuhkanan)

    i = 0; j = 0; k = 0
    while i < len(separuhkiri) and j < len(separuhkanan):
        if separuhkiri[i] < separuhkanan[j]:
            A[k] = separuhkiri[i]
            i = i + 1
        else:
            A[k] = separuhkanan[j]
            j = j + 1
        k = k + 1

    while i < len(separuhkiri):
        A[k] = separuhkiri[i]
        i = i + 1
        k = k + 1

    while j < len(separuhkanan):
        A[k] = separuhkanan[j]
        j = j + 1
        k = k + 1

for x in A:
    try:
        linked.data = x
        linked = linked.tautan
    except:
        pass

mergeSortLL(a)
cetak(a)
```