In [2]:
```
#"I affirm that I have not given or received any unauthorized help on this assigr
#Christian Ed B. Efa BSCPE12S1
```

In [6]:
```
class Circuit:

    def setName(self,name):
        self._name = name
    def getName(self):
        return self._name

    def setDescription(self,description):
        self._description = description
    def getDescription(self):
        return self._description

    def setValue(self,value):
        self._value = value
    def getValue(self,value):
        return self._value

    def display(self):
        print('Name: ', self.getName())
        print('What is A', self.getDescription())

    def displayvalue(self):
        print(self.getValue())

component = Circuit()
component.setName('Component')
component.setDescription('Component?\nIt comprises several different components s
component.display()

resistor = component
resistor.setName('Resistor')
resistor.setDescription('Resistor?\nresistor is a passive two-terminal electrical
resistor.display()
```

```
Name:  Component
What is A Component?
It comprises several different components such as resistors, transistors, capac
itors, inductors, and diodes. Conductive wires or traces are used to connect th
e components to each other. However, a circuit is complete only if it starts an
d ends at the same point, forming a loop.

Name:  Resistor
What is A Resistor?
resistor is a passive two-terminal electrical component that implements electri
cal resistance as a circuit element. In electronic circuits, resistors are used
to reduce current flow, adjust signal levels, to divide voltages, bias active e
lements, and terminate transmission lines, among other uses..
```

In [20]:
```
class Series(Circuit):
    def setValue(self, s1, s2, s3, s4):
        self._s1 = s1
        self._s2 = s2
        self._s3 = s3
        self._s4 = s4
        self._srreq = s1 + s2 + s3 + s4
    def getValue(self):
        return self._s1, self._s2, self._s3, self._s4
series = Series()
series.setValue(22,10,5,2)
print(series.getValue)
print("The Resistance is", round(series._srreq,1), "Ω.")

class Series:
    def setValue(self, r1, r2, r3, r4):
        self._r1 = r1
        self._r2 = r2
        self._r3 = r3
        self._r4 = r4
        self._srreq = r1 + r2 + r3 + r4
    def getValue(self):
            return self._r1, self._r2, self._r3, self._r4
series = Series()
series.setValue(22,10,5,2)
print(series.getValue)
print("The Resistance is", round(series._srreq,1), "Ω.")
```

```
<bound method Series.getValue of <__main__.Series object at 0x0000028B423F3F70>
>
The Resistance is 39 Ω.
<bound method Series.getValue of <__main__.Series object at 0x0000028B424E60D0>
>
The Resistance is 39 Ω.
```

In [21]:
```
class Parallel:
    def setValue(self, r1, r2, r3):
        self._r1 = r1
        self._r2 = r2
        self._r3 = r3
        self._prreq = (1/r1 + 1/r2 + 1/r3)**-1
    def getValue(self):
        return self._pr1, self._pr2, self._pr3
parallel = Parallel()
parallel.setValue(7,5,12)
print(parallel.getValue)
print("The Equivalent Resistance is", round(parallel._prreq,1), "Ω.")

class Parallel:
    def setValue(self, r1, r2, r3):
        self._r1 = r1
        self._r2 = r2
        self._r3 = r3
        self._prreq = (1/r1 + 1/r2 + 1/r3)**-1
    def getValue(self):
        return self._pr1, self._pr2, self._pr3
parallel = Parallel()
parallel.setValue(7,5,12)
print(parallel.getValue)
print("The Equivalent Resistance is", round(parallel._prreq,1), "Ω.")
```

```
<bound method Parallel.getValue of <__main__.Parallel object at 0x0000028B424E6
430>>
The Equivalent Resistance is 2.3 Ω.
<bound method Parallel.getValue of <__main__.Parallel object at 0x0000028B424E6
5B0>>
The Equivalent Resistance is 2.3 Ω.
```

In [ ]: