Assignment - 19

1A.

```
class Thing:
   pass
print(Thing)
example = Thing()
print(example)
```

No they are not same

2A.

```
class Thing2:
   letters="abc"


print(Thing2.letters)
```

3A.

```
class Thing3:
   letters="xyz"
print(Thing3.letters)
```

Object not required

4A.

```
class Element:
   def __init__(self,name,symbol,number):
      self.name = name
      self.symbol = symbol
      self.number = number
```

```
        def __str__(self):
            return "details: name={}, symbol={}, number={}
            ".format(self.name,self.symbol,self.number)
ele = Element('Hydrogen','H',1)
print(ele)
```

5A.

```
dict12={'name':'Hydrogen1','symbol':'H','number':1}
ele1 = Element(dict12['name'],dict12['symbol'],dict12['number'])
print(ele1)
```

6A.

```
class Element1:
    def __init__(self,name,symbol,number):
        self.name = name
        self.symbol = symbol
        self.number = number
    def dump(self):
        print(self.name,',',self.symbol,',',self.number)

hydrogen=Element1('Hydrogen','H',1)
hydrogen.dump()
```

7A.

```
# using __str__
class Element2:
    def __init__(self,name,symbol,number):
```

```
        self.name = name
        self.symbol = symbol
        self.number = number
    def __str__(self):
        return "details: name={}, symbol={}, number={}
".format(self.name,self.symbol,self.number)


hydrogen1=Element2('Hydrogen','H',1)
print(hydrogen1)
```

8A.

```
# private variable and getter method - property() function
class Element3:
    def __init__(self,name,symbol,number):
        self._name = name
        self._symbol = symbol
        self._number = number
    def getname(self):
        return self._name
    def getsymbol(self):
        return self._symbol
    def getnumber(self):
        return self._number


    name=property(getname)
```

```
    symbol=property(getsymbol)
    number=property(getnumber)

hydrogen1=Element3('Hyd','H',1)
print(hydrogen1.name)
print(hydrogen1.symbol)
print(hydrogen1.number)
```

9A.

```
class Bear:
    def eats(self):
        return 'berries'
class Rabbit:
    def eats(self):
        return 'clover'
class Octothrope:
    def eats(self):
        return 'campers'

b = Bear()
print(b.eats())
r = Rabbit()
print(r.eats())
o = Octothrope()
print(o.eats())
```

10A.

```
class Laser():
    def does(self):
        return 'disintegrate'


class Claw():
    def does(self):
        return 'crush'


class SmartPhone():
    def does(self):
        return 'ring'


class Robot():
    def __init__(self, laser, claw, smartphone):
        self.laser = laser
        self.claw = claw
        self.smartphone = smartphone

    def does(self):
        print(self.laser.does(), self.claw.does(), self.smartphone.does())


laser = Laser()
claw = Claw()
```

```
smartphone = SmartPhone()

robot = Robot(laser, claw, smartphone)

robot.does()
```