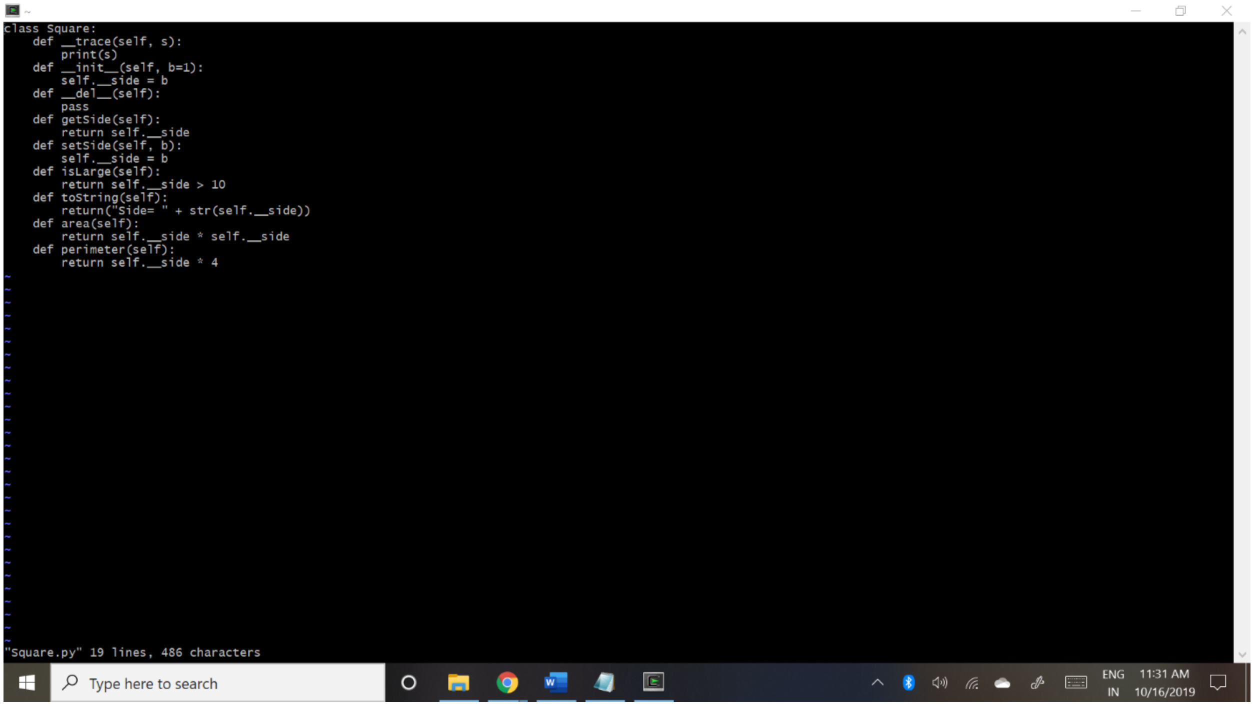
ClassicalChinaCoin

1. Metal.py

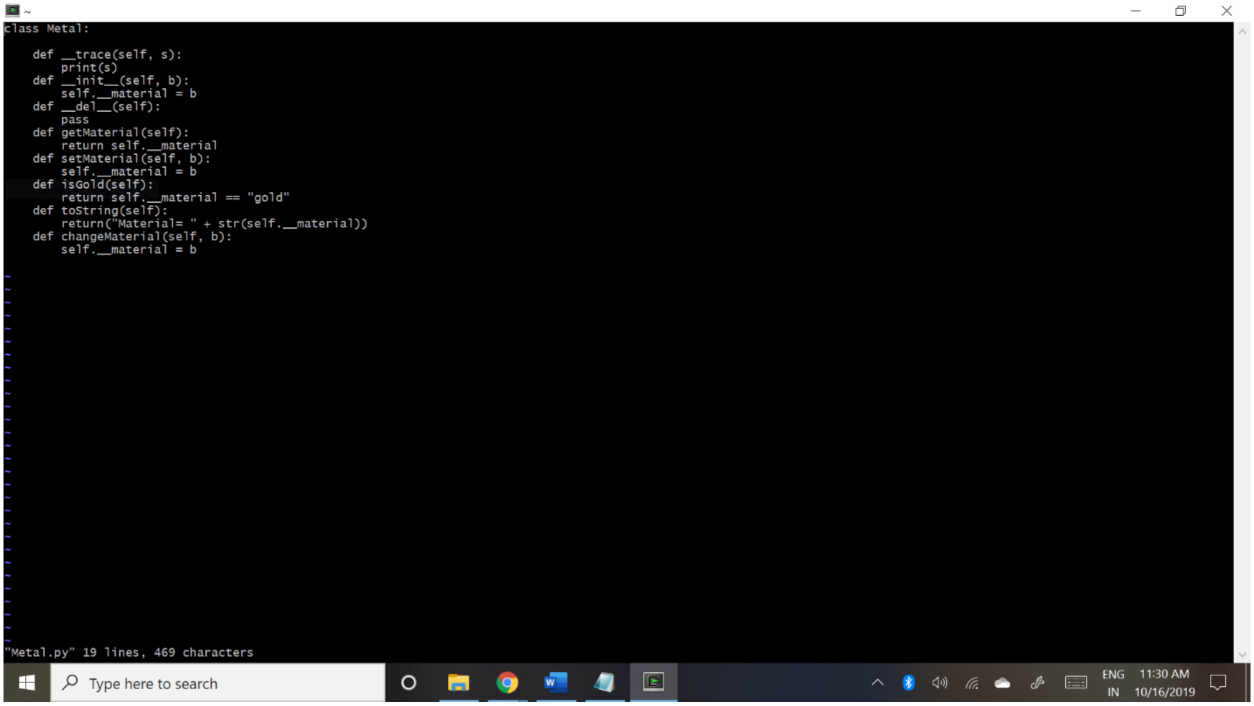
```
class Metal:
  def __trace(self, s):
    print(s)
  def __init__(self, b):
    self.__material = b
  def __del__(self):
    pass
  def getMaterial(self):
    return self.__material
  def setMaterial(self, b):
    self.__material = b
  def isGold(self):
    return self.__material == "gold"
  def toString(self):
    return("Material= " + str(self.__material))
  def changeMaterial(self, b):
    self.__material = b
```



```
class Square:
    def __trace(self, s):
        print(s)
    def __init__(self, b=1):
        self.__side = b
    def __del__(self):
        pass
    def getSide(self):
        return self.__side
    def setSide(self, b):
        self.__side = b
    def isLarge(self):
        return self.__side > 10
    def toString(self):
        return("Side= " + str(self.__side))
    def area(self):
        return self.__side * self.__side
    def perimeter(self):
        return self.__side * 4
"Square.py" 19 lines, 486 characters
```

2. Square.py

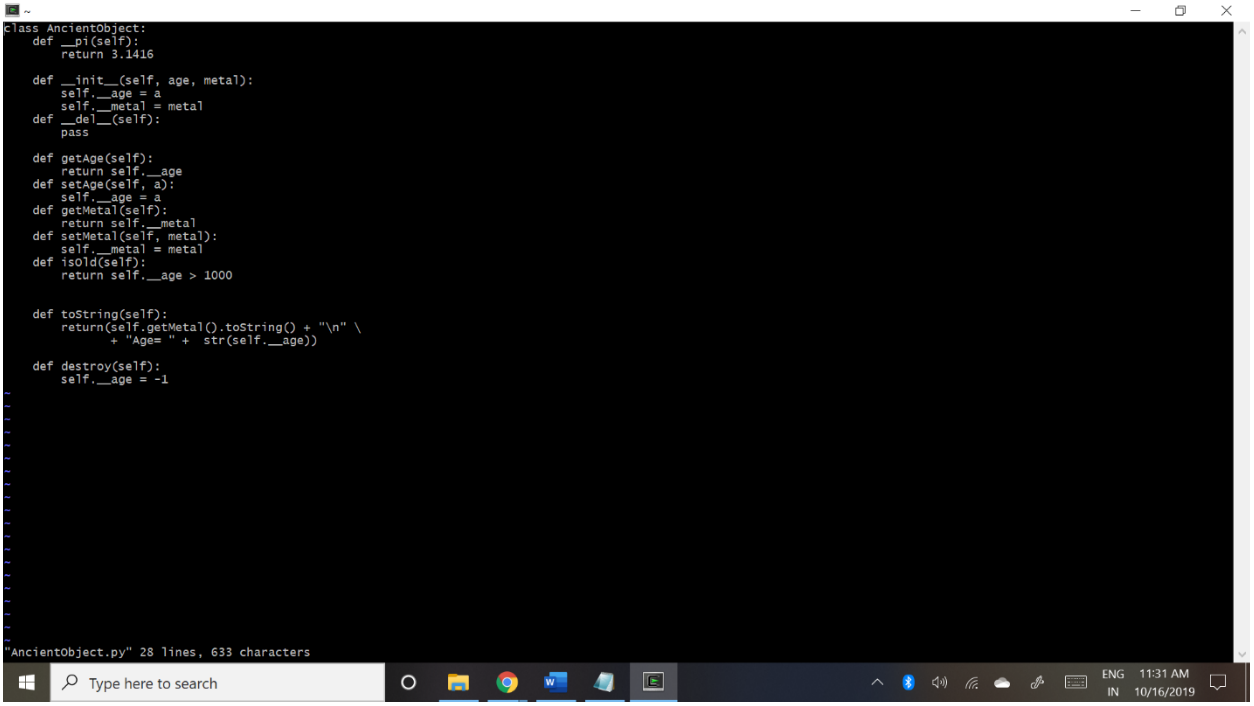
```
class Square:
   def __trace(self, s):
      print(s)
   def __init__(self, b=1):
      self.__side = b
   def __del__(self):
      pass
   def getSide(self):
      return self.__side
   def setSide(self, b):
      self.__side = b
   def isLarge(self):
      return self.__side > 10
   def toString(self):
      return("Side= " + str(self.__side))
   def area(self):
      return self.__side * self.__side
   def perimeter(self):
      return self.__side * 4
```



```
class Metal:

    def __trace(self, s):
        print(s)
    def __init__(self, b):
        self.__material = b
    def __del__(self):
        pass
    def getMaterial(self):
        return self.__material
    def setMaterial(self, b):
        self.__material = b
    def isGold(self):
        return self.__material == "gold"
    def toString(self):
        return("Material= " + str(self.__material))
    def changeMaterial(self, b):
        self.__material = b

"Metal.py" 19 lines, 469 characters
```

3. AncientObject.py

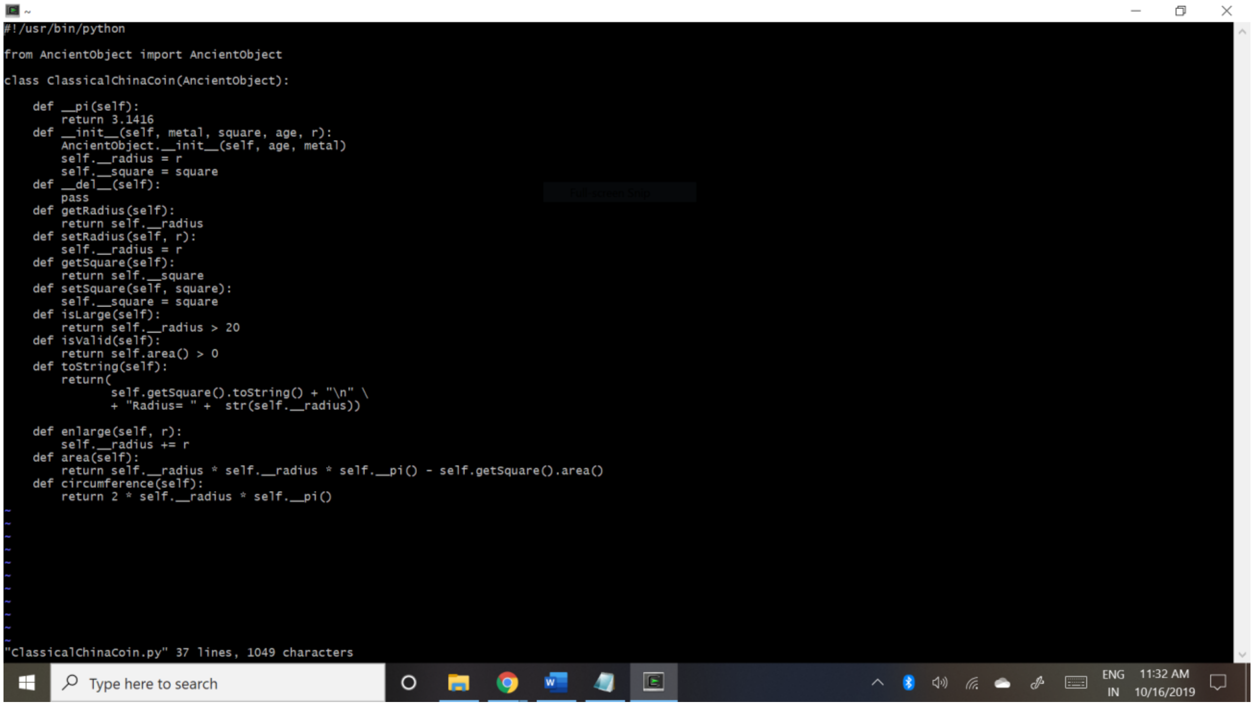
```
class AncientObject:
    def __pi(self):
        return 3.1416
    def __init__(self, age, metal):
        self.__age = a
        self.__metal = metal
    def __del__(self):
        pass
    def getAge(self):
        return self.__age
    def setAge(self, a):
        self.__age = a
    def getMetal(self):
        return self.__metal
    def setMetal(self, metal):
        self.__metal = metal
    def isOld(self):
        return self.__age > 1000
    def toString(self):
        return(self.getMetal().toString() + "\n" \
                + "Age= " +  str(self.__age))
    def destroy(self):
                self.__age = -1
```



```
class AncientObject:
    def __pi(self):
        return 3.1416

    def __init__(self, age, metal):
        self.__age = a
        self.__metal = metal
    def __del__(self):
        pass

    def getAge(self):
        return self.__age
    def setAge(self, a):
        self.__age = a
    def getMetal(self):
        return self.__metal
    def setMetal(self, metal):
        self.__metal = metal
    def isOld(self):
        return self.__age > 1000


    def toString(self):
        return(self.getMetal().toString() + "\n" \
               + "Age= " +  str(self.__age))

    def destroy(self):
        self.__age = -1
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
"AncientObject.py" 28 lines, 633 characters
```

4. ClassicalChinaCoin.py

```
#!/usr/bin/python
from AncientObject import AncientObject

class ClassicalChinaCoin(AncientObject):
   def __pi(self):
      return 3.1416
   def __init__(self, metal, square, age, r):
      AncientObject.__init__(self, age, metal)
      self.__radius = r
      self.__square = square
   def __del__(self):
      pass
   def getRadius(self):
      return self.__radius
   def setRadius(self, r):
      self.__radius = r
   def getSquare(self):
      return self.__square
   def setSquare(self, square):
      self.__square = square
   def isLarge(self):
      return self.__radius > 20
   def isValid(self):
      return self.area() > 0
   def toString(self):
      return(
            self.getSquare().toString() + "\n" \
            + "Radius= " +  str(self.__radius))
   def enlarge(self, r):
      self.__radius += r
   def area(self):
      return self.__radius * self.__radius * self.__pi() - self.getSquare().area()
   def circumference(self):
      return 2 * self.__radius * self.__pi()
```

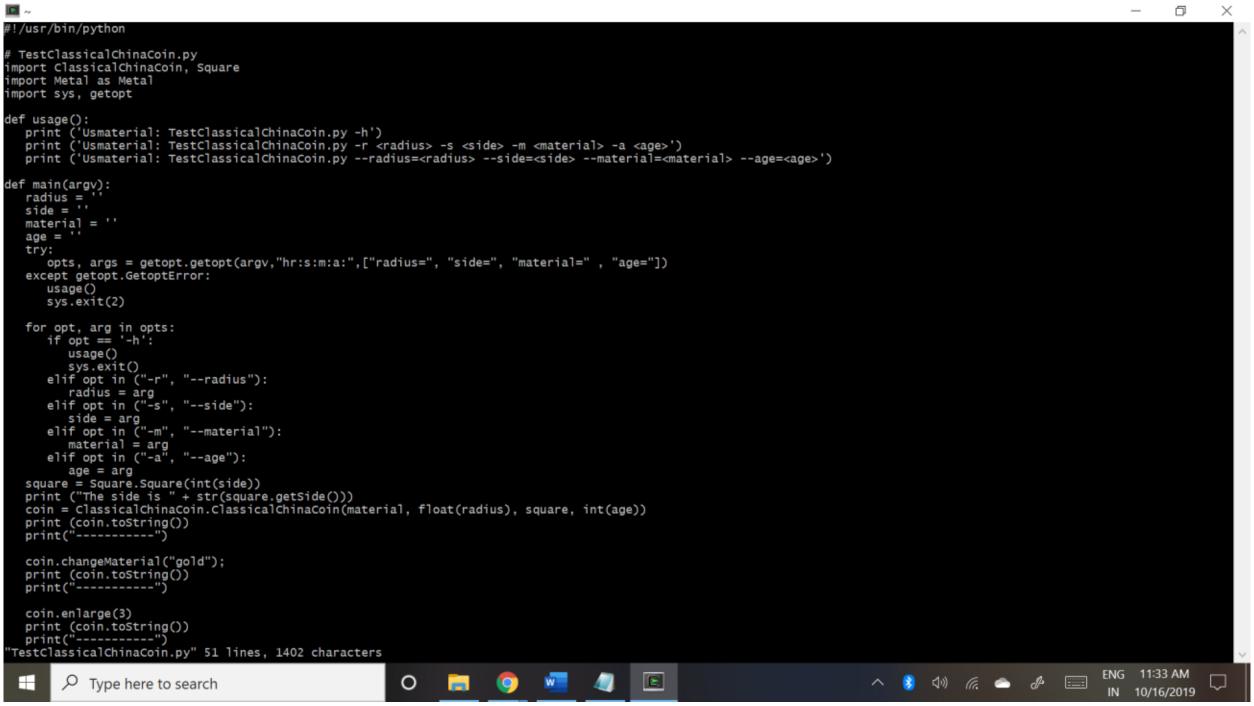


```
#!/usr/bin/python

from AncientObject import AncientObject

class ClassicalChinaCoin(AncientObject):

    def __pi(self):
        return 3.1416
    def __init__(self, metal, square, age, r):
        AncientObject.__init__(self, age, metal)
        self.__radius = r
        self.__square = square
    def __del__(self):
        pass
    def getRadius(self):
        return self.__radius
    def setRadius(self, r):
        self.__radius = r
    def getSquare(self):
        return self.__square
    def setSquare(self, square):
        self.__square = square
    def isLarge(self):
        return self.__radius > 20
    def isValid(self):
        return self.area() > 0
    def toString(self):
        return(
               self.getSquare().toString() + "\n" \
               + "Radius= " +  str(self.__radius))

    def enlarge(self, r):
        self.__radius += r
    def area(self):
        return self.__radius * self.__radius * self.__pi() - self.getSquare().area()
    def circumference(self):
        return 2 * self.__radius * self.__pi()
~
~
~
~
~
~
~
~
~
~
~
"ClassicalChinaCoin.py" 37 lines, 1049 characters
```

5. TestClassicalChinaCoin.py

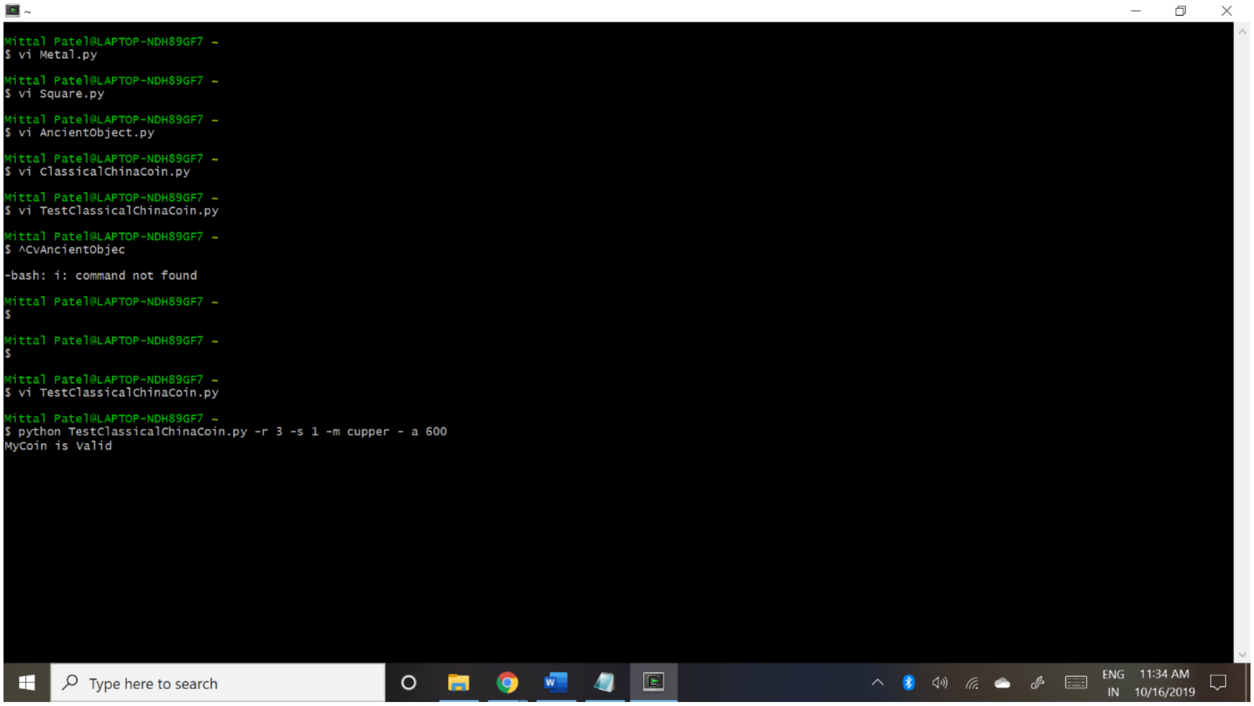
```
#!/usr/bin/python
# TestClassicalChinaCoin.py

import ClassicalChinaCoin, Square
import Metal as Metal
import sys, getopt

def usage():
  print ('Usmaterial: TestClassicalChinaCoin.py -h')
  print ('Usmaterial: TestClassicalChinaCoin.py -r <radius> -s <side> -m <material> -a <age>')
  print ('Usmaterial: TestClassicalChinaCoin.py --radius=<radius> --side=<side> --material=<material>
--age=<age>')

def main(argv):
  radius = ''
  side = ''
  material = ''
  age = ''
  try:
    opts, args = getopt.getopt(argv,"hr:s:m:a:",["radius=", "side=", "material=" , "age="])
  except getopt.GetoptError:
    usage()
    sys.exit(2)

  for opt, arg in opts:
    if opt == '-h':
      usage()
      sys.exit()
    elif opt in ("-r", "--radius"):
      radius = arg
    elif opt in ("-s", "--side"):
      side = arg
    elif opt in ("-m", "--material"):
      material = arg
    elif opt in ("-a", "--age"):
      age = arg
  square = Square.Square(int(side))
  print ("The side is " + str(square.getSide()))
  coin = ClassicalChinaCoin.ClassicalChinaCoin(material, float(radius), square, int(age))
  print (coin.toString())
  print("-----------")
```

```
coin.changeMaterial("gold");
print (coin.toString())
print("-----------")

coin.enlarge(3)
print (coin.toString())
print("-----------")

if __name__ == '__main__':
   main(sys.argv[1:])
```



```
#!/usr/bin/python

# TestClassicalChinaCoin.py
import ClassicalChinaCoin, Square
import Metal as Metal
import sys, getopt

def usage():
   print ('Usmaterial: TestClassicalChinaCoin.py -h')
   print ('Usmaterial: TestClassicalChinaCoin.py -r <radius> -s <side> -m <material> -a <age>')
   print ('Usmaterial: TestClassicalChinaCoin.py --radius=<radius> --side=<side> --material=<material> --age=<age>')

def main(argv):
   radius = ''
   side = ''
   material = ''
   age = ''
   try:
      opts, args = getopt.getopt(argv,"hr:s:m:a:",["radius=", "side=", "material=" , "age="])
   except getopt.GetoptError:
      usage()
      sys.exit(2)

   for opt, arg in opts:
      if opt == '-h':
         usage()
         sys.exit()
      elif opt in ("-r", "--radius"):
         radius = arg
      elif opt in ("-s", "--side"):
         side = arg
      elif opt in ("-m", "--material"):
         material = arg
      elif opt in ("-a", "--age"):
         age = arg
   square = Square.Square(int(side))
   print ("The side is " + str(square.getSide()))
   coin = ClassicalChinaCoin.ClassicalChinaCoin(material, float(radius), square, int(age))
   print (coin.toString())
   print("-----------")

   coin.changeMaterial("gold");
   print (coin.toString())
   print("-----------")

   coin.enlarge(3)
   print (coin.toString())
   print("-----------")
"TestClassicalChinaCoin.py" 51 lines, 1402 characters
```

Output:



```
Mittal Patel@LAPTOP-NDH89GF7 ~
$ vi Metal.py

Mittal Patel@LAPTOP-NDH89GF7 ~
$ vi Square.py

Mittal Patel@LAPTOP-NDH89GF7 ~
$ vi AncientObject.py

Mittal Patel@LAPTOP-NDH89GF7 ~
$ vi ClassicalChinaCoin.py

Mittal Patel@LAPTOP-NDH89GF7 ~
$ vi TestClassicalChinaCoin.py

Mittal Patel@LAPTOP-NDH89GF7 ~
$ ^CvAncientObjec

-bash: i: command not found

Mittal Patel@LAPTOP-NDH89GF7 ~
$

Mittal Patel@LAPTOP-NDH89GF7 ~
$

Mittal Patel@LAPTOP-NDH89GF7 ~
$ vi TestClassicalChinaCoin.py

Mittal Patel@LAPTOP-NDH89GF7 ~
$ python TestClassicalChinaCoin.py -r 3 -s 1 -m cupper - a 600
MyCoin is Valid
```