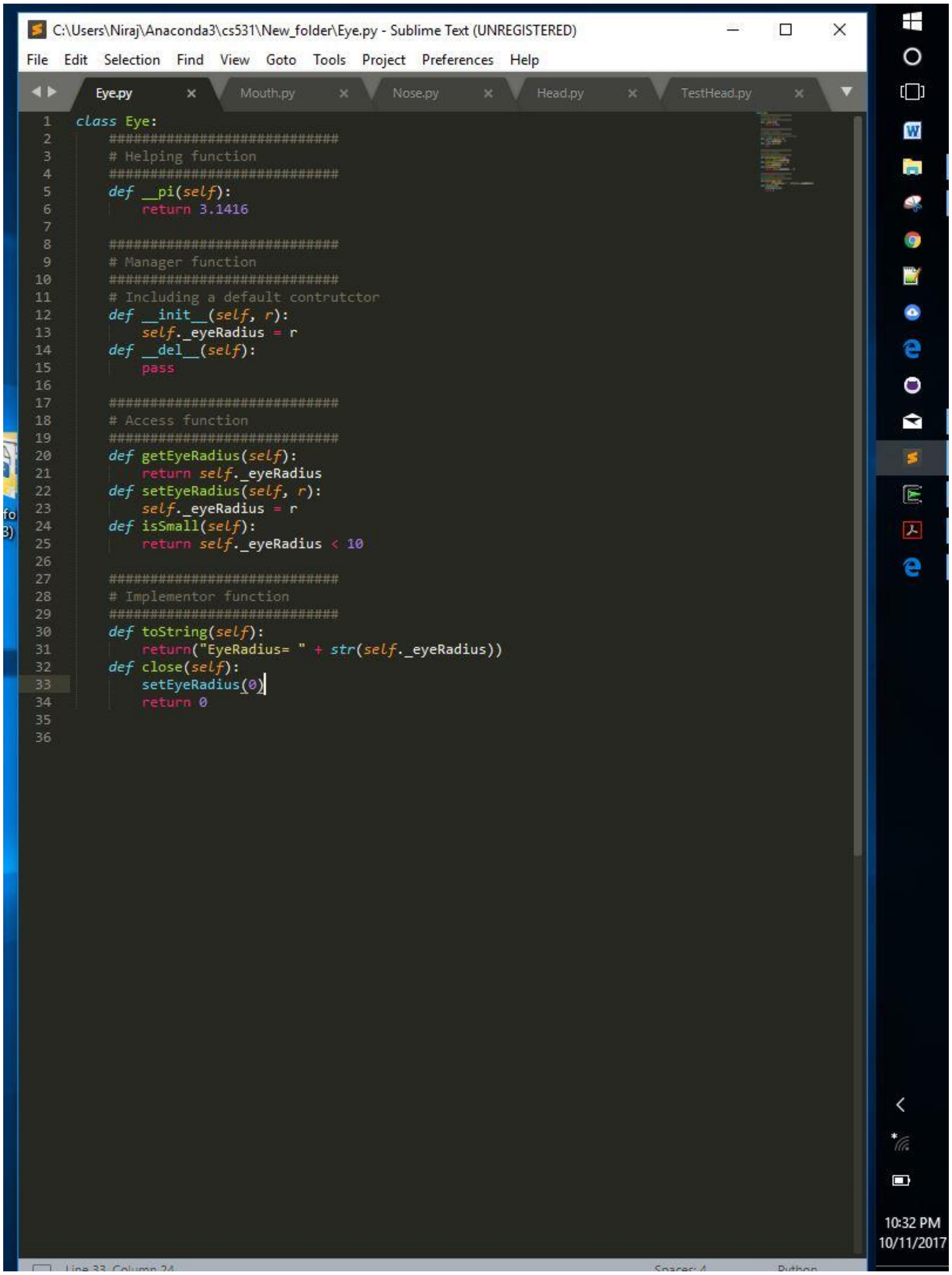
Name : Niraj Thanki   SID : 19376 CLASS : CS531

Eye.py



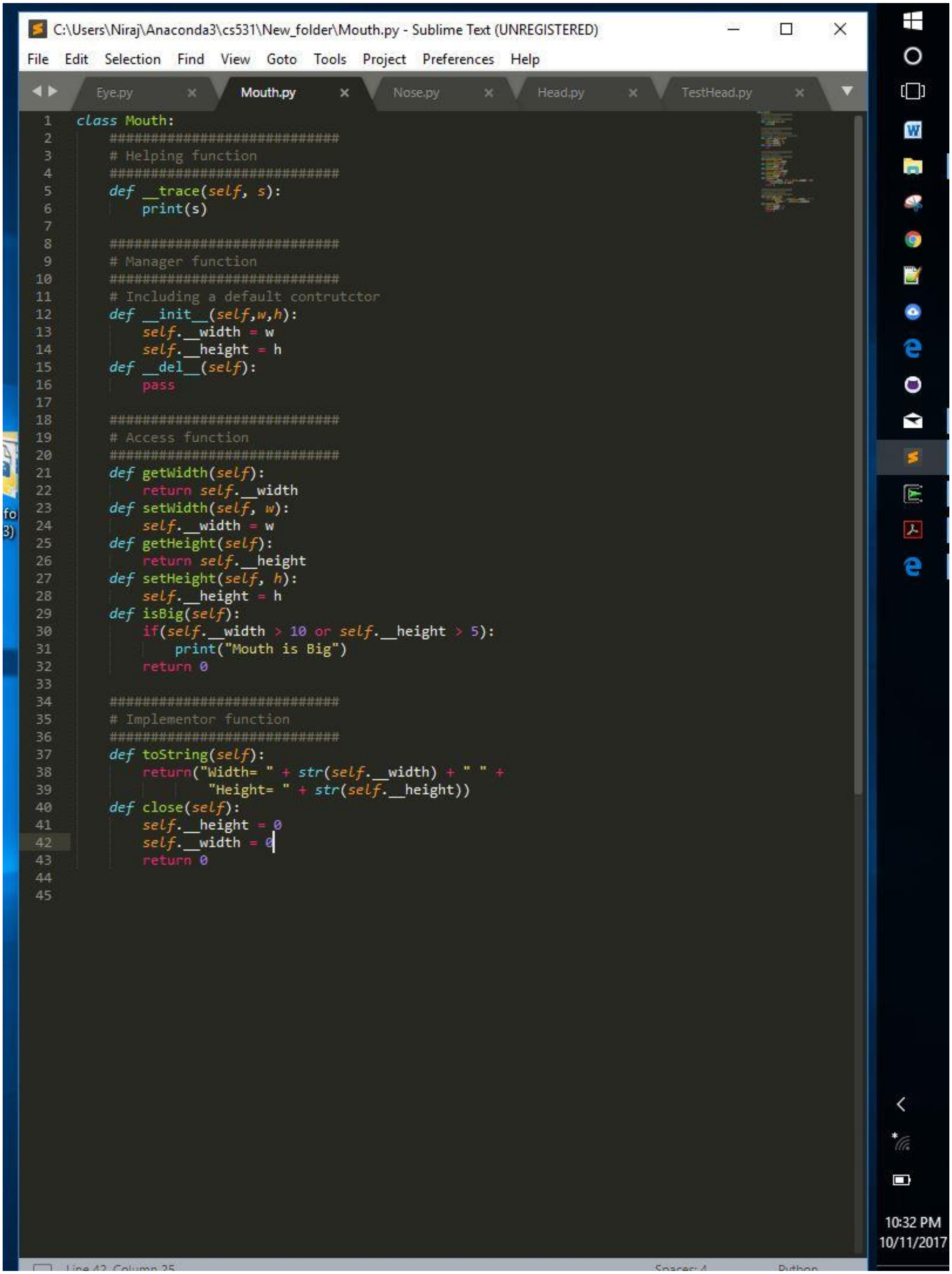
```
class Eye:
    ############################
    # Helping function
    ############################
    def __pi(self):
        return 3.1416

    ############################
    # Manager function
    ############################
    # Including a default contrutctor
    def __init__(self, r):
        self._eyeRadius = r
    def __del__(self):
        pass

    ############################
    # Access function
    ############################
    def getEyeRadius(self):
        return self._eyeRadius
    def setEyeRadius(self, r):
        self._eyeRadius = r
    def isSmall(self):
        return self._eyeRadius < 10

    ############################
    # Implementor function
    ############################
    def toString(self):
        return("EyeRadius= " + str(self._eyeRadius))
    def close(self):
        setEyeRadius(0)
        return 0
```

Eye.py   Soure Code

```
class Eye:
    ############################
    # Helping function
    ############################
    def __pi(self):
        return 3.1416

    ############################
    # Manager function
    ############################
    # Including a default contrutctor
    def __init__(self, r):
        self._eyeRadius = r
    def __del__(self):
        pass

    ############################
    # Access function
    ############################
    def getEyeRadius(self):
        return self._eyeRadius
    def setEyeRadius(self, r):
        self._eyeRadius = r
    def isSmall(self):
        return self._eyeRadius < 10
```

```
############################
# Implementor function
############################
def toString(self):
    return("EyeRadius= " + str(self._eyeRadius))
def close(self):
    setEyeRadius(0)
    return 0
```

Mouth.py



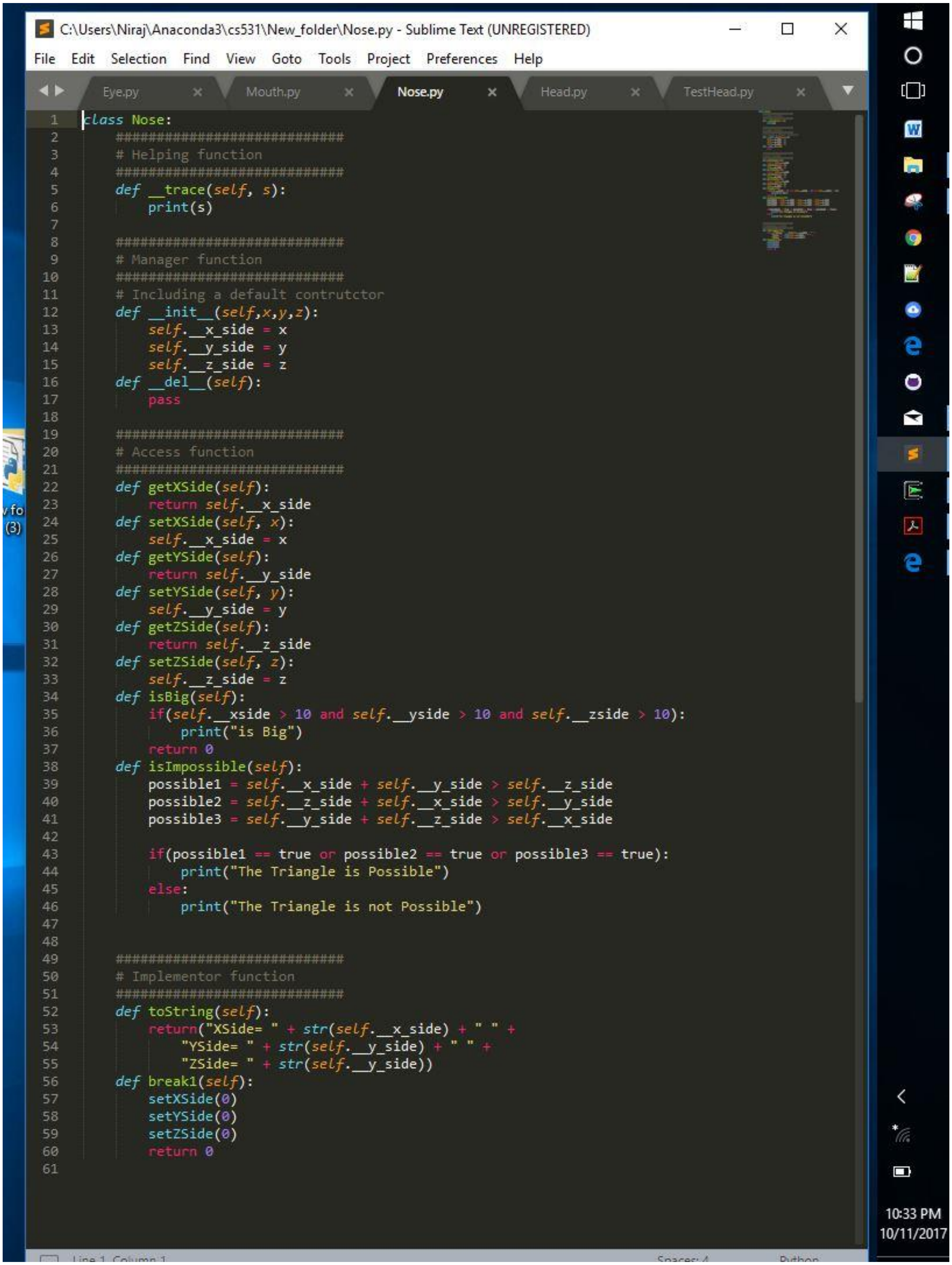
```python
class Mouth:
    ############################
    # Helping function
    ############################
    def __trace(self, s):
        print(s)

    ############################
    # Manager function
    ############################
    # Including a default contrutctor
    def __init__(self,w,h):
        self.__width = w
        self.__height = h
    def __del__(self):
        pass

    ############################
    # Access function
    ############################
    def getWidth(self):
        return self.__width
    def setWidth(self, w):
        self.__width = w
    def getHeight(self):
        return self.__height
    def setHeight(self, h):
        self.__height = h
    def isBig(self):
        if(self.__width > 10 or self.__height > 5):
            print("Mouth is Big")
        return 0

    ############################
    # Implementor function
    ############################
    def toString(self):
        return("Width= " + str(self.__width) + " " +
                "Height= " + str(self.__height))
    def close(self):
        self.__height = 0
        self.__width = 0
        return 0


```

<u>Mouth.py   Soure Code</u>

```
class Mouth:
    ############################
    # Helping function
    ############################
    def __trace(self, s):
        print(s)

    ############################
    # Manager function
    ############################
    # Including a default contrutctor
    def __init__(self,w,h):
        self.__width = w
        self.__height = h
    def __del__(self):
        pass

    ############################
    # Access function
    ############################
    def getWidth(self):
        return self.__width
    def setWidth(self, w):
        self.__width = w
```

```
def getHeight(self):

    return self.__height

def setHeight(self, h):

    self.__height = h

def isBig(self):

    if(self.__width > 10 or self.__height > 5):

        print("Mouth is Big")

    return 0


############################

# Implementor function

############################

def toString(self):

    return("Width= " + str(self.__width) + " " +

            "Height= " + str(self.__height))

def close(self):

    self.__height = 0

    self.__width = 0

    return 0
```

Nose.py



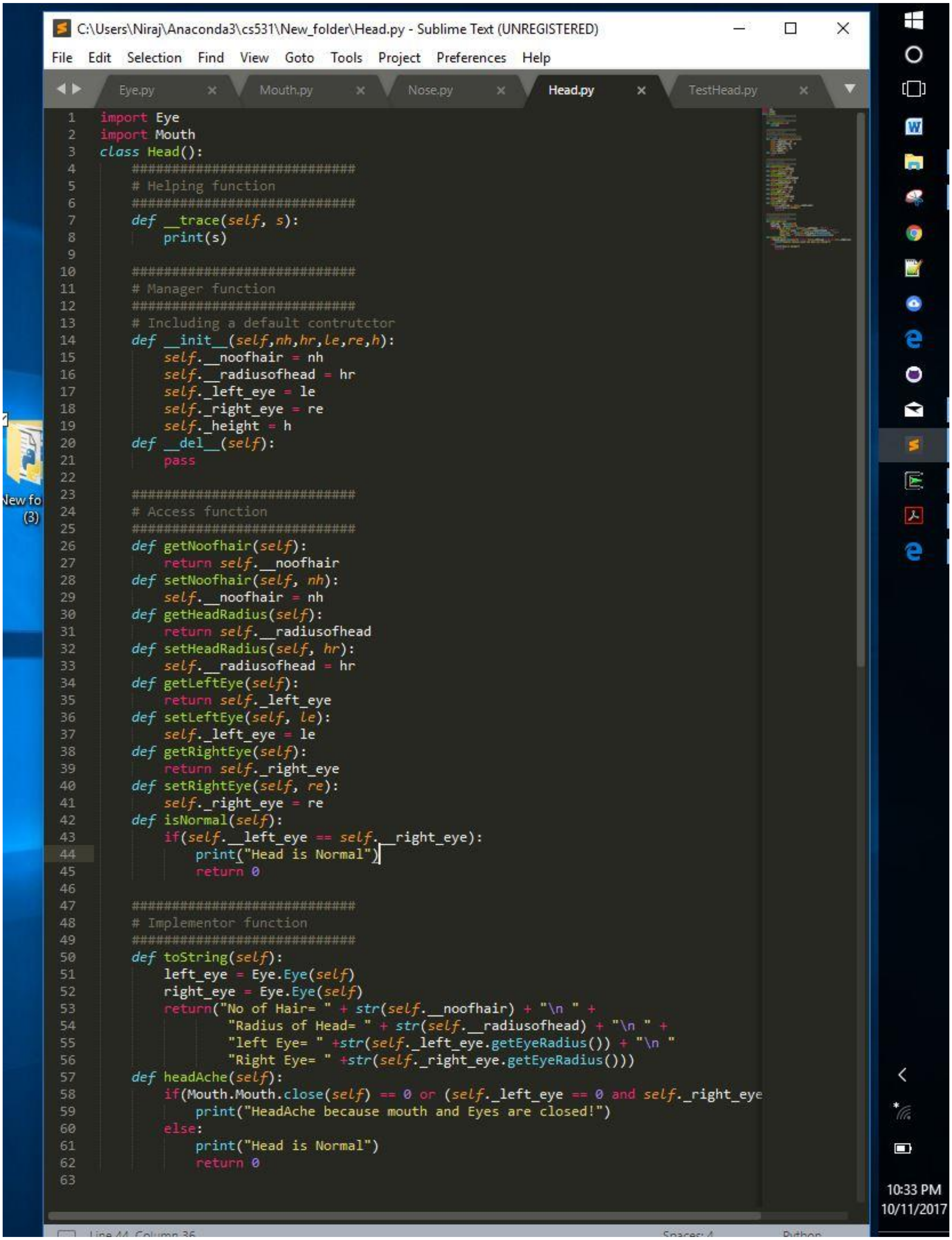
```python
class Nose:
    ############################
    # Helping function
    ############################
    def __trace(self, s):
        print(s)

    ############################
    # Manager function
    ############################
    # Including a default contrutctor
    def __init__(self,x,y,z):
        self.__x_side = x
        self.__y_side = y
        self.__z_side = z
    def __del__(self):
        pass

    ############################
    # Access function
    ############################
    def getXSide(self):
        return self.__x_side
    def setXSide(self, x):
        self.__x_side = x
    def getYSide(self):
        return self.__y_side
    def setYSide(self, y):
        self.__y_side = y
    def getZSide(self):
        return self.__z_side
    def setZSide(self, z):
        self.__z_side = z
    def isBig(self):
        if(self.__xside > 10 and self.__yside > 10 and self.__zside > 10):
            print("is Big")
        return 0
    def isImpossible(self):
        possible1 = self.__x_side + self.__y_side > self.__z_side
        possible2 = self.__z_side + self.__x_side > self.__y_side
        possible3 = self.__y_side + self.__z_side > self.__x_side

        if(possible1 == true or possible2 == true or possible3 == true):
            print("The Triangle is Possible")
        else:
            print("The Triangle is not Possible")


    ############################
    # Implementor function
    ############################
    def toString(self):
        return("XSide= " + str(self.__x_side) + " " +
            "YSide= " + str(self.__y_side) + " " +
            "ZSide= " + str(self.__y_side))
    def break1(self):
        setXSide(0)
        setYSide(0)
        setZSide(0)
        return 0
```

Nose.py   Soure Code

```
class Nose:
    ############################
    # Helping function
    ############################
    def __trace(self, s):
        print(s)

    ############################
    # Manager function
    ############################
    # Including a default contrutctor
    def __init__(self,x,y,z):
        self.__x_side = x
        self.__y_side = y
        self.__z_side = z
    def __del__(self):
        pass

    ############################
    # Access function
    ############################
    def getXSide(self):
        return self.__x_side
    def setXSide(self, x):
        self.__x_side = x
    def getYSide(self):
        return self.__y_side
    def setYSide(self, y):
        self.__y_side = y
    def getZSide(self):
        return self.__z_side
    def setZSide(self, z):
        self.__z_side = z
    def isBig(self):
        if(self.__xside > 10 and self.__yside > 10 and self.__zside > 10):
            print("is Big")
        return 0
    def isImpossible(self):
        possible1 = self.__x_side + self.__y_side > self.__z_side
        possible2 = self.__z_side + self.__x_side > self.__y_side
        possible3 = self.__y_side + self.__z_side > self.__x_side

        if(possible1 == true or possible2 == true or possible3 == true):
            print("The Triangle is Possible")
        else:
            print("The Triangle is not Possible")
```

```
############################

# Implementor function

############################

def toString(self):

    return("XSide= " + str(self.__x_side) + " " +

        "YSide= " + str(self.__y_side) + " " +

        "ZSide= " + str(self.__y_side))

def break1(self):

    setXSide(0)

    setYSide(0)

    setZSide(0)

    return 0
```

Head.py


C:\Users\Niraj\Anaconda3\cs531\New_folder\Head.py - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
Eye.py
Mouth.py
Nose.py
Head.py
TestHead.py
New fo (3)
Line 44, Column 36
Spaces: 4
Python
10:33 PM
10/11/2017

```python
import Eye
import Mouth
class Head():
    ##############################
    # Helping function
    ##############################
    def __trace(self, s):
        print(s)

    ##############################
    # Manager function
    ##############################
    # Including a default contrutctor
    def __init__(self,nh,hr,le,re,h):
        self.__noofhair = nh
        self.__radiusofhead = hr
        self._left_eye = le
        self._right_eye = re
        self._height = h
    def __del__(self):
        pass

    ##############################
    # Access function
    ##############################
    def getNoofhair(self):
        return self.__noofhair
    def setNoofhair(self, nh):
        self.__noofhair = nh
    def getHeadRadius(self):
        return self.__radiusofhead
    def setHeadRadius(self, hr):
        self.__radiusofhead = hr
    def getLeftEye(self):
        return self._left_eye
    def setLeftEye(self, le):
        self._left_eye = le
    def getRightEye(self):
        return self._right_eye
    def setRightEye(self, re):
        self._right_eye = re
    def isNormal(self):
        if(self.__left_eye == self.__right_eye):
            print("Head is Normal")
            return 0

    ##############################
    # Implementor function
    ##############################
    def toString(self):
        left_eye = Eye.Eye(self)
        right_eye = Eye.Eye(self)
        return("No of Hair= " + str(self.__noofhair) + "\n " +
                "Radius of Head= " + str(self.__radiusofhead) + "\n " +
                "left Eye= " +str(self._left_eye.getEyeRadius()) + "\n "
                "Right Eye= " +str(self._right_eye.getEyeRadius()))
    def headAche(self):
        if(Mouth.Mouth.close(self) == 0 or (self._left_eye == 0 and self._right_eye
            print("HeadAche because mouth and Eyes are closed!")
        else:
            print("Head is Normal")
            return 0

```

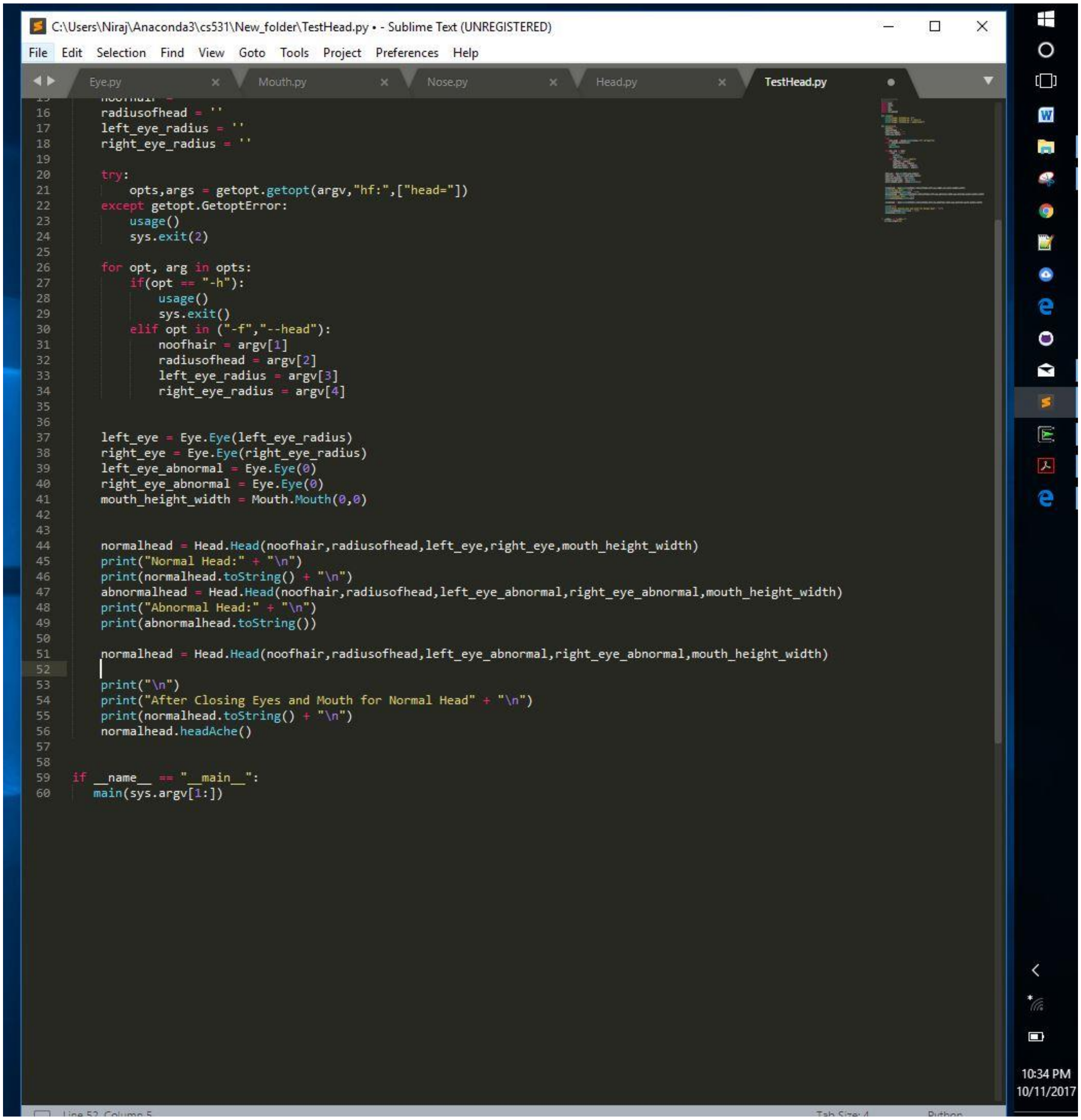
Head.py Source Code

```
import Eye
import Mouth
class Head():
    #############################
    # Helping function
    #############################
    def __trace(self, s):
        print(s)

    #############################
    # Manager function
    #############################
    # Including a default contrutctor
    def __init__(self,nh,hr,le,re,h):
        self.__noofhair = nh
        self.__radiusofhead = hr
        self._left_eye = le
        self._right_eye = re
        self._height = h
    def __del__(self):
        pass

    #############################
    # Access function
```

```
############################
def getNoofhair(self):
    return self.__noofhair
def setNoofhair(self, nh):
    self.__noofhair = nh
def getHeadRadius(self):
    return self.__radiusofhead
def setHeadRadius(self, hr):
    self.__radiusofhead = hr
def getLeftEye(self):
    return self._left_eye
def setLeftEye(self, le):
    self._left_eye = le
def getRightEye(self):
    return self._right_eye
def setRightEye(self, re):
    self._right_eye = re
def isNormal(self):
    if(self.__left_eye == self.__right_eye):
        print("Head is Normal")
        return 0

############################
# Implementor function
############################
```

```
def toString(self):
    left_eye = Eye.Eye(self)
    right_eye = Eye.Eye(self)
    return("No of Hair= " + str(self.__noofhair) + "\n " +
            "Radius of Head= " + str(self.__radiusofhead) + "\n " +
            "left Eye= " +str(self._left_eye.getEyeRadius()) + "\n "
            "Right Eye= " +str(self._right_eye.getEyeRadius()))
def headAche(self):
    if(Mouth.Mouth.close(self) == 0 or (self._left_eye == 0 and self._right_eye == 0)):
        print("HeadAche because mouth and Eyes are closed!")
    else:
        print("Head is Normal")
        return 0
```

TestHead.py


C:\Users\Niraj\Anaconda3\cs531\New_folder\TestHead.py • - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
Eye.py
Mouth.py
Nose.py
Head.py
TestHead.py
Line 52, Column 5
Tab Size: 4
Python
10:34 PM
10/11/2017

```python
    noofhair =
    radiusofhead = ''
    left_eye_radius = ''
    right_eye_radius = ''

    try:
        opts,args = getopt.getopt(argv,"hf:",["head="])
    except getopt.GetoptError:
        usage()
        sys.exit(2)

    for opt, arg in opts:
        if(opt == "-h"):
            usage()
            sys.exit()
        elif opt in ("-f","--head"):
            noofhair = argv[1]
            radiusofhead = argv[2]
            left_eye_radius = argv[3]
            right_eye_radius = argv[4]


    left_eye = Eye.Eye(left_eye_radius)
    right_eye = Eye.Eye(right_eye_radius)
    left_eye_abnormal = Eye.Eye(0)
    right_eye_abnormal = Eye.Eye(0)
    mouth_height_width = Mouth.Mouth(0,0)


    normalhead = Head.Head(noofhair,radiusofhead,left_eye,right_eye,mouth_height_width)
    print("Normal Head:" + "\n")
    print(normalhead.toString() + "\n")
    abnormalhead = Head.Head(noofhair,radiusofhead,left_eye_abnormal,right_eye_abnormal,mouth_height_width)
    print("Abnormal Head:" + "\n")
    print(abnormalhead.toString())

    normalhead = Head.Head(noofhair,radiusofhead,left_eye_abnormal,right_eye_abnormal,mouth_height_width)
    
    print("\n")
    print("After Closing Eyes and Mouth for Normal Head" + "\n")
    print(normalhead.toString() + "\n")
    normalhead.headAche()


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

TestHead.py Source Code

#!/usr/bin/python

#TestFace.py

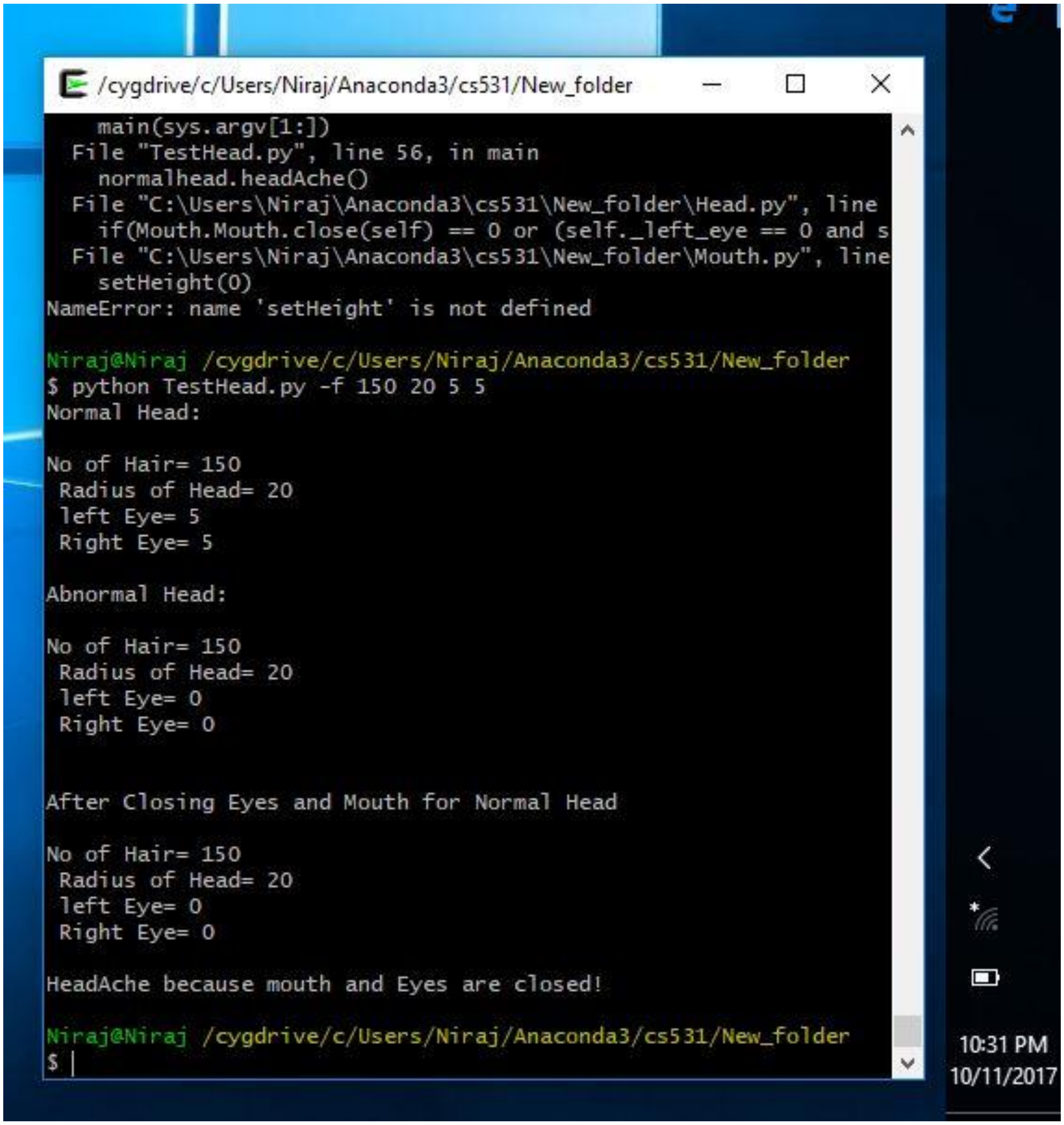
import Head

import Mouth

```
import Eye
import Nose
import sys,getopt

def usage():
    print("Usage: TestHead.py -h")
    print("Usage: TestHead.py -f <face>")
    print("Usage: TestHead.py --head=<head>")

def main(argv):
    noofhair = ''
    radiusofhead = ''
    left_eye_radius = ''
    right_eye_radius = ''

    try:
        opts,args = getopt.getopt(argv,"hf:",["head="])
    except getopt.GetoptError:
        usage()
        sys.exit(2)

    for opt, arg in opts:
        if(opt == "-h"):
            usage()
            sys.exit()
        elif opt in ("-f","--head"):
            noofhair = argv[1]
            radiusofhead = argv[2]
            left_eye_radius = argv[3]
            right_eye_radius = argv[4]



    left_eye = Eye.Eye(left_eye_radius)
    right_eye = Eye.Eye(right_eye_radius)
    left_eye_abnormal = Eye.Eye(0)
    right_eye_abnormal = Eye.Eye(0)
    mouth_height_width = Mouth.Mouth(0,0)



    normalhead = Head.Head(noofhair,radiusofhead,left_eye,right_eye,mouth_height_width)
    print("Normal Head:" + "\n")
    print(normalhead.toString() + "\n")
    abnormalhead =
Head.Head(noofhair,radiusofhead,left_eye_abnormal,right_eye_abnormal,mouth_height_width)
    print("Abnormal Head:" + "\n")
    print(abnormalhead.toString())

    normalhead =
Head.Head(noofhair,radiusofhead,left_eye_abnormal,right_eye_abnormal,mouth_height_width)

    print("\n")
```

```
        print("After Closing Eyes and Mouth for Normal Head" + "\n")

        print(normalhead.toString() + "\n")

        normalhead.headAche()



if __name__ == "__main__":

  main(sys.argv[1:])
```

OUTPUT :



```
    main(sys.argv[1:])
  File "TestHead.py", line 56, in main
    normalhead.headAche()
  File "C:\Users\Niraj\Anaconda3\cs531\New_folder\Head.py", line
    if(Mouth.Mouth.close(self) == 0 or (self._left_eye == 0 and s
  File "C:\Users\Niraj\Anaconda3\cs531\New_folder\Mouth.py", line
    setHeight(0)
NameError: name 'setHeight' is not defined

Niraj@Niraj /cygdrive/c/Users/Niraj/Anaconda3/cs531/New_folder
$ python TestHead.py -f 150 20 5 5
Normal Head:

No of Hair= 150
 Radius of Head= 20
 left Eye= 5
 Right Eye= 5

Abnormal Head:

No of Hair= 150
 Radius of Head= 20
 left Eye= 0
 Right Eye= 0


After Closing Eyes and Mouth for Normal Head

No of Hair= 150
 Radius of Head= 20
 left Eye= 0
 Right Eye= 0

HeadAche because mouth and Eyes are closed!

Niraj@Niraj /cygdrive/c/Users/Niraj/Anaconda3/cs531/New_folder
$ |
```