1DA2223 - Python programming and data analysis

Exercise 2 - Self done task [complete 2022L] *Ostatnia modyfikacja: R. Szmurło 22.03.2022 05:50*

Object Oriented Programming

Exercises

Problem 1

Construct a generator which returns a sequence of prime numbers as a function, class and expression ().

```
def myprimef():
  (...)

class MyPrimeC(object):
  (...)

primeg = (...)
```

Problem 2

Fill in the Line class methods to accept coordinate as a pair of tuples and return the slope and distance of the line.

```
class Line(object):

    def __init__(self,coor1,coor2):
        """Initialize instance attributes with tuples (x1,y1)  and (x2,y2)
        """
        (...)

    def distance(self):
        """Calculate the length of the segment (line)
        """
        (...)

    def slope(self):
        """ Return the slope of a line going through the ends ( the 'a' in y=ax+b)
        """
        (...)
```

EXAMPLE OUTPUT

```
coordinate1 = (3,2)
coordinate2 = (8,10)
```

```
li = Line(coordinate1,coordinate2)

li.distance()
9.433981132056603

li.slope()
1.6
```

####Problem 3

Fill in the class

```
class Cylinder(object):

    def __init__(self,height=1,radius=1):
        (...)

    def volume(self):
        (...)

    def surface_area(self):
        (...)
```

EXAMPLE OUTPUT

```
c = Cylinder(2,3)

c.volume()
56.52

c.surface_area()
94.2
```

####Problem 4
Write a class that takes a filename as an argument, opens it and reads its content into a 2D matrix of values (an array of arrays: [[1,2],[3,4]]). The class should define a function info() which prints out statistical information about values in columns. Assume that the first row in the file specifies the column names. Assume that the cells can be both numbers or strings (you should plan a conversion strategy for statistical data analysis).

```
class DataFile(object):

  def __init__(self, filename='undef'):
    (...)

  def info(self):
    (...)

  def avg(self, colnum=0, colname=''):
    """ The column name or colnum can be provided alternatively
    """
    (...)

  def min(self, colnum=0, colname=''):
      (...)

  def max(self, colnum=0, colname=''):
      (...)

  def distinc(self, colnum=0, colname=''):
     "Counts distinct number of values in a given column."
      (...)
```

HINTS:
For pretty printing values in the table use formatted string literals (from python 3.6+):
print(f'''{(123+98+138)/3:^20.2f}''');

EXAMPLE INPUT

Given the file content is:

```
Name;Age;Weight;Height
John;6;25;123
Mary;4;18;98
Jack;8;32;138
```

```
df = DataFile('myfile.csv')
df.info()
```

EXAMPLE OUTPUT

The info() function should produce the following Statistical Data information:

```
           Min      Max      Avg    Distinct
Name:       -        -        -         3
Age:        4        8        6         3
Weight:     18       32       25        3
Height:     98      138     119.66      3
```