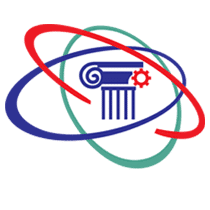
**Acropolis Institute of Technology and Research**

****

**Dept. of C.S.I.T**

**PYTHON LAB WORK**

**Submitted By:- Submitted To:**

**Piyush Dave Prof. NIDHI NIGAM MAM**

**0827CI191042**

INDEX

* write a program in Python to find mean median mode for the given set of number in the list.
* Python program to find first n prime numbers
* write a code to find even and odd numbers
* write a program to find square root of a number
* 5.write a python program exponentiation (power of a number)
* 6.write a python program fibonacci series
* 7.write a python program showing inheritance
* 8.write a python program function polymorphism
* 9.Write a python program to print each line of a file in reverse order.

1. write a program in Python to find mean median mode for the given set of number in the list.

numb = [2, 4, 5, 8, 9]

no = len(numb)

numb.sort()

if no % 2 == 0:

median1 = numb[no//2]

median2 = numb[no//2 - 1]

median = (median1 + median2)/2

else:

median = numb[no//2]

print("The median of the given numbers (", numb, ") is", str(median))

numb = [2, 3, 5, 7, 8]

no = len(numb)

summ = sum(numb)

mean = summ / no

print("The mean or average of all these numbers (", numb, ") is", str(mean))

from collections import Counter

numb = [2, 3, 4, 5, 7, 2]

no = len(numb)

val = Counter(numb)

findMode = dict(val)

mode = [i for i, v in findMode.items() if v == max(list(val.values()))]

if len(mode) == no:

findMode = "The group of number do not have any mode"

else:

findMode = "The mode of a number is / are: " + ', '.join(map(str, mode))

print(findMode)

2.Python program to find first n prime numbers

numr=int(input("Enter range:"))

print("Prime numbers:",end=' ')

for n in range(1,numr):

for i in range(2,n):

if(n%i==0):

break

else:

print(n,end=' ')

3. write a code to find even and odd numbers

num = int(input("Enter a number: "))

if (num % 2) == 0:

print("{0} is Even".format(num))

else:

print("{0} is Odd".format(num))

Run Code

4. write a program to find square root of a number

num\_sqrt = num \*\* 0.5

print('The square root of %0.3f is %0.3f'%(num ,num\_sqrt))

5.write a python program exponentiation (power of a number)

base = 3

exponent = 4

result = 1

while exponent != 0:

result \*= base

exponent-=1

print("Answer = " + str(result))

6.write a python program fibonacci series

# Program to display the Fibonacci sequence up to n-th term

nterms = int(input("How many terms? "))

# first two terms

n1, n2 = 0, 1

count = 0

# check if the number of terms is valid

if nterms <= 0:

print("Please enter a positive integer")

# if there is only one term, return n1

elif nterms == 1:

print("Fibonacci sequence upto",nterms,":")

print(n1)

# generate fibonacci sequence

else:

print("Fibonacci sequence:")

while count < nterms:

print(n1)

nth = n1 + n2

# update values

n1 = n2

n2 = nth

count += 1

7.write a python program showing inheritance

# Python program to show

# multilevel inheritance

# The base class

class grandma:

def \_\_init\_\_(self, grandmaname):

self.grandmaname = grandmaname

# Middle class

class mother(grandma):

def \_\_init\_\_(self, mothername, grandmaname):

self.mothername = mothername

# invoke a constructor of grandma class

grandma.\_\_init\_\_(self, grandmaname)

# last class

class son(mother):

def \_\_init\_\_(self,sonname, mothername, grandmaname):

self.sonname = sonname

# invoke a constructor of mother class

father.\_\_init\_\_(self, mothername, grandmaname)

def print\_name(self):

print('Grandma name :', self.grandmaname)

print("Mother name :", self.mothername)

print("Son name :", self.sonname)

8.write a python program function polymorphism

class Cat:

def \_\_init\_\_(self, name, age):

self.name = name

self.age = age

def info(self):

print(f"I am a cat. My name is {self.name}. I am {self.age} years old.")

def make\_sound(self):

print("Meow")

class Dog:

def \_\_init\_\_(self, name, age):

self.name = name

self.age = age

def info(self):

print(f"I am a dog. My name is {self.name}. I am {self.age} years old.")

def make\_sound(self):

print("Bark")

cat1 = Cat("Kitty", 2.5)

dog1 = Dog("Fluffy", 4)

for animal in (cat1, dog1):

animal.make\_sound()

animal.info()

animal.make\_sound()

9.Write a python program to print each line of a file in reverse order.

input\_file=open('D:/a.txt','r')

for line in input\_file:

l=len(line)

s=' '

while(l>=1):

s=s+line[l-1]

l=l-1

print(s)

input\_file.close()