1. Write a Python program to check if the given number is a Disarium Number?

Ans.

import logging as lg

# importing logging so every function call of

lg.basicConfig(filename ='C:\\Users\\Home\\Johns python talent\\logging\\testlog1.log', level =lg.INFO , format = '%(asctime)s %(message)s')

def check\_disarium(n):

k = 1

disarium = 0

for i in str(n):

disarium += int(i)\*\*k

k += 1

if disarium == n:

print(n,"is a disarium number")

else:

print(n,"is not a disarium number")

try:

check\_disarium(175)

lg.info("Function check\_disarium(175) has been called")

except exception as e:

print("There was an error called: ",e)

else:

pass

finally:

pass

1. Write a Python program to print all disarium numbers between 1 to 100?

Ans.

import logging as lg

# importing logging so every function call of

lg.basicConfig(filename ='C:\\Users\\Home\\Johns python talent\\logging\\testlog1.log', level =lg.INFO , format = '%(asctime)s %(message)s')

def disarium\_in\_range():

start = int(input("Enter your starting number : "))

ending = int(input("Enter your ending number number : "))

for j in range(start,ending+1):

k = 1

disarium = 0

for i in str(j):

disarium += int(i)\*\*k

k += 1

if disarium == int(j):

print(disarium,"is a disarium number")

try:

disarium\_in\_range()

lg.info("Function disarium\_in\_range() has been called")

except exception as e:

print("There was an error called: ",e)

else:

pass

finally:

pass

1. Write a Python program to check if the given number is Happy Number?

Ans.

import logging as lg

# importing logging so every function call of

lg.basicConfig(filename ='C:\\Users\\Home\\Johns python talent\\logging\\testlog1.log', level =lg.INFO , format = '%(asctime)s %(message)s')

def check\_happy(n):

empty\_set = []

while True:

happy = 0

for i in str(n):

happy += int(i)\*\*2

empty\_set.append(happy)

if empty\_set.count(1) == 1:

print(n, "This is a Happy Number")

break

if empty\_set.count(happy) == 2:

print("This is not a happy number")

break

else:

n = happy

try:

check\_happy(14)

lg.info("Function check\_happy(78) has been called")

except exception as e:

print("There was an error called: ",e)

else:

pass

finally:

pass

1. Write a Python program to print all happy numbers between 1 and 100?

Ans.

import logging as lg

# importing logging so every function call of

lg.basicConfig(filename ='C:\\Users\\Home\\Johns python talent\\logging\\testlog1.log', level =lg.INFO , format = '%(asctime)s %(message)s')

def check\_happy\_range(start, end):

for p in range(start,end+1):

num1 = p

empty\_set = []

while True:

happy = 0

for i in str(num1):

happy += int(i)\*\*2

empty\_set.append(happy)

if empty\_set.count(1) == 1:

print(p, "is a Happy Number")

break

if empty\_set.count(happy) == 2:

print(p,"is not a happy number")

break

else:

num1 = happy

try:

check\_happy\_range(10,100)

lg.info("Function check\_happy(78) has been called")

except exception as e:

print("There was an error called: ",e)

else:

pass

finally:

pass

1. Write a Python program to determine whether the given number is a Harshad Number?

Ans.

import logging as lg

# importing logging so every function call of

lg.basicConfig(filename ='C:\\Users\\Home\\Johns python talent\\logging\\testlog1.log', level =lg.INFO , format = '%(asctime)s %(message)s')

def check\_harshad(n):

# program for harshad number

num = 0

for i in str(n):

num += int(i)

if n%num == 0:

print(n," is harshad number")

if n%num != 0:

print(n," is not a harshad number")

try:

check\_harshad(378)

lg.info("Function check\_harshad(378) has been called")

except exception as e:

print("There was an error called: ",e)

else:

pass

finally:

pass

1. Write a Python program to print all pronic numbers between 1 and 100?

Ans.

import logging as lg

# importing logging so every function call of

lg.basicConfig(filename ='C:\\Users\\Home\\Johns python talent\\logging\\testlog1.log', level =lg.INFO , format = '%(asctime)s %(message)s')

def check\_pronic\_range(start, end):

for j in range(start,end+1):

for i in range(2,j):

if j%i == 0:

if ((j/i)\*((j/i)+1)) == int(j):

print(j, "is a pronic number")

print(i ,"x", i-1, "=", j)

break

try:

check\_pronic\_range(1,100)

lg.info("Function check\_pronic\_range(1,100) has been called")

except exception as e:

print("There was an error called: ",e)

else:

pass

finally:

pass