Assignment
Module – 4 (Advance python programming)

1. What is File function in python? What is keywords to create and write file.

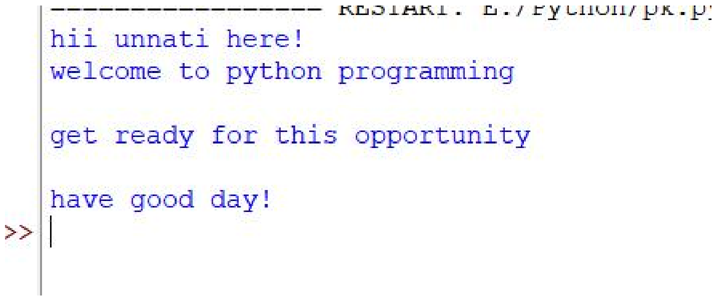
-> Files in Python can be opened with a built-in open() function. To create a file in Python, you can use the open() function with the 'a' mode such as file = open('myfile. txt', 'a') .

2. Write a Python program to read an entire text file.

Code:

```
def file_read(fname):
   txt = open(fname)
   print(txt.read())
file_read("test.txt")
```

Output:



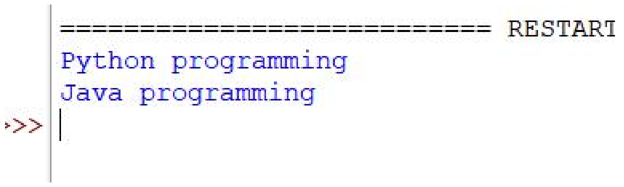
```
----------------- RESTART: E:/Python/pk.p
hii unnati here!
welcome to python programming

get ready for this opportunity

have good day!
>>
```

3. Write a Python program to append text to a file and display the text.

Code:

```
def file_read(fname):
        from itertools import islice
        with open(fname, "w") as myfile:
                myfile.write("Python programming\n")
                myfile.write("Java programming")
        txt = open(fname)
        print(txt.read())
file_read('abc.txt')
```

Output:



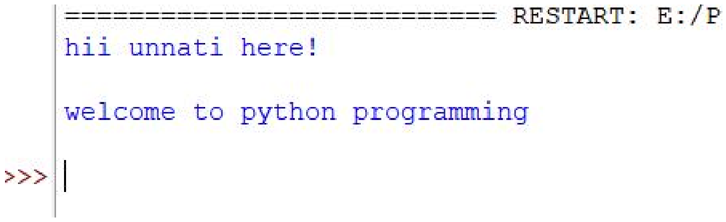
```
=========================== RESTART
Python programming
Java programming
>>> |
```

4. Write a Python program to read first n lines of a file.

Code:

```
def file_read_from_head(fname, nlines):
        from itertools import islice
        with open(fname) as f:
                for line in islice(f, nlines):
                        print(line)
file_read_from_head('test.txt',2)
```

Output:



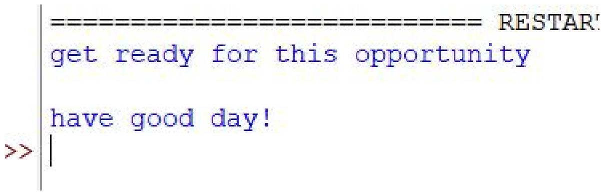
```
=========================== RESTART: E:/P
hii unnati here!

welcome to python programming

>>> |
```

5. Write a Python program to read last n lines of a file.

Code:

```
def LastNlines(fname, N):

    with open(fname) as file:

        for line in (file.readlines() [-N:]):
            print(line, end ='')
if __name__ == '__main__':
    fname = 'test.txt'
    N = 3
    try:
        LastNlines(fname, N)
    except:
        print('File not found')
```

Output:



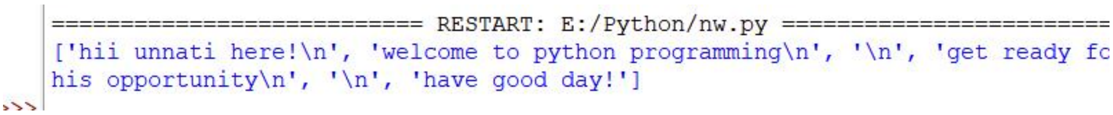
```
=========================== RESTAR
get ready for this opportunity

have good day!
>>
```

6. Write a Python program to read a file line by line and store it into a list

Code:

```
def file_read(fname):
        with open(fname) as f:
                #Content_list is the list that contains the read lines.
                content_list = f.readlines()
                print(content_list)

file_read('test.txt')
```

Output:



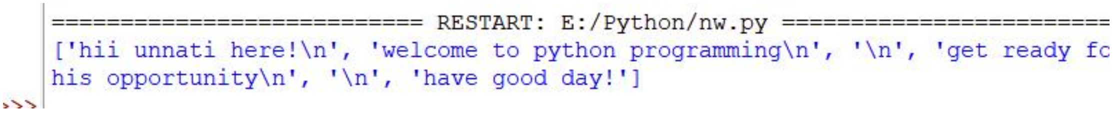
```
=========================== RESTART: E:/Python/nw.py ========================
['hii unnati here!\n', 'welcome to python programming\n', '\n', 'get ready fc
his opportunity\n', '\n', 'have good day!']
>>>
```

7. Write a Python program to read a file line by line store it into a variable.

Code:

```
def file_read(fname):
        with open (fname, "r") as myfile:
                data=myfile.readlines()
                print(data)
file_read('test.txt')
```

Output:



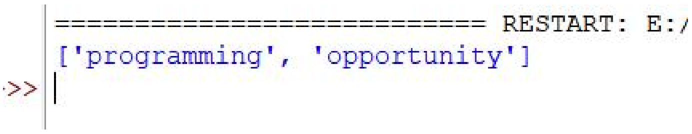
```
=========================== RESTART: E:/Python/nw.py ========================
['hii unnati here!\n', 'welcome to python programming\n', '\n', 'get ready fc
his opportunity\n', '\n', 'have good day!']
>>>
```

8. Write a python program to find the longest words.

Code:

```
def longest_word(filename):
   with open(filename, 'r') as infile:
          words = infile.read().split()
   max_len = len(max(words, key=len))
   return [word for word in words if len(word) == max_len]

print(longest_word('test.txt'))
```

Output:



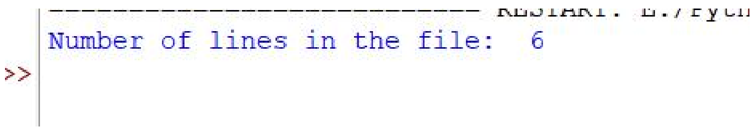
```
========================== RESTART: E:/
['programming', 'opportunity']
>>> |
```

9. Write a Python program to count the number of lines in a text file.

Code:

```
def file_lengthy(fname):
      with open(fname) as f:
            for i, l in enumerate(f):
                  pass
      return i + 1
print("Number of lines in the file: ",file_lengthy("test.txt"))
```

Output:



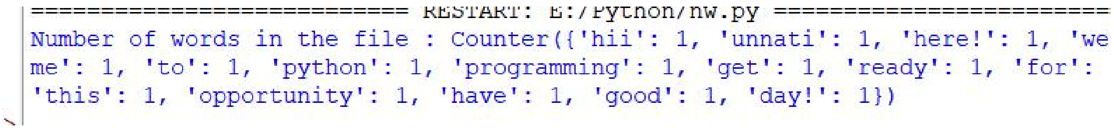
```
=========================== RESTART: E:/Pyth
Number of lines in the file:  6
>>>
```

10. Write a Python program to count the frequency of words in a file.

Code:

```
from collections import Counter
def word_count(fname):
        with open(fname) as f:
                return Counter(f.read().split())

print("Number of words in the file :",word_count("test.txt"))
```

Output:



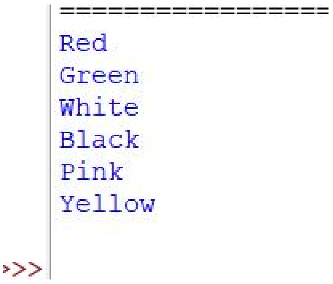
```
========================== RESTART: E:/Python/nw.py ========================
Number of words in the file : Counter({'hii': 1, 'unnati': 1, 'here!': 1, 'we
me': 1, 'to': 1, 'python': 1, 'programming': 1, 'get': 1, 'ready': 1, 'for':
'this': 1, 'opportunity': 1, 'have': 1, 'good': 1, 'day!': 1})
```

11. Write a Python program to write a list to a file.

Code:

```
color = ['Red', 'Green', 'White', 'Black', 'Pink', 'Yellow']
with open('abc.txt', "w") as myfile:
        for c in color:
                myfile.write("%s\n" % c)

content = open('abc.txt')
print(content.read())
```

Output:



```
==================
Red
Green
White
Black
Pink
Yellow

>>>
```

12. Explain Exception handling? What is an Error in Python?.

-> An error is an issue in a program that prevents the program from completing its task.
an exception is a condition that interrupts the normal flow of the program. Both errors and exceptions are a type of runtime error, which means they occur during the execution of a program.

13. How many except statements can a try-except block have? Name Some built-in exception classes:

-> Have multiple try blocks in a program but only one except statement with each try block.

14. When will the else part of try-except-else be executed?

-> The else part is executed when no exception occurs.

15. Can one block of except statements handle multiple exception?

-> Yes, a single block of except statements in Python can handle multiple exceptions

16. When is the finally block executed?

-> The finally block always executes when the try block exits. This ensures that the finally block is executed even if an unexpected exception occurs.

17. What happens when „1"== 1 is executed?

-> It simply evaluates to False and does not raise any exception.

18. How Do You Handle Exceptions With Try/Except/Finally In Python? Explain with coding snippets.

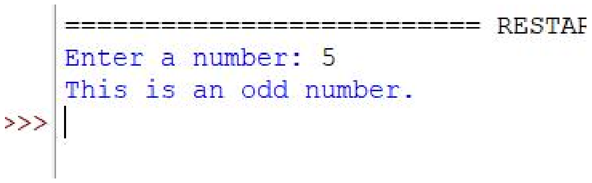
-> If an exception occurs during execution of the try clause, the exception may be handled by an except clause. If the exception is not handled by an except clause, the exception is re-raised after the finally clause has been executed.

19. Write python program that user to enter only odd numbers, else will raise an exception.

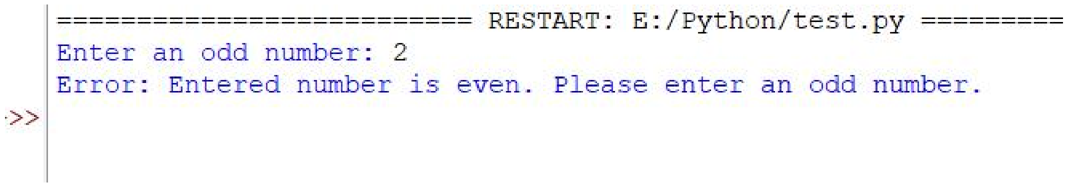
Code:

```
class EvenNumberException(Exception):
    pass

def get_odd_number():
    while True:
        try:
            user_input = int(input("Enter an odd number: "))
            if user_input % 2 == 1:
                return user_input
            else:
                raise EvenNumberException("Entered number is even. Please
enter an odd number.")
        except ValueError:
            print("Invalid input. Please enter a valid integer.")

try:
    odd_number = get_odd_number()
    print(f"You entered an odd number: {odd_number}")
except EvenNumberException as e:
    print(f"Error: {e}")
```

Output:



```
    ========================== RESTAF
    Enter a number: 5
    This is an odd number.
>>> |
```



```
========================= RESTART: E:/Python/test.py =========
Enter an odd number: 2
Error: Entered number is even. Please enter an odd number.
>>
```

20. What are oops concepts? Is multiple inheritance supported in java

-> OOP concepts include abstraction, encapsulation, inheritance and polymorphism. Basically, Java OOP concepts let us create working methods and variables, then re-use all or part of them without compromising security

->Java doesn't support Multiple Inheritance, but we can use interfaces (instead of classes) to achieve the same purpose.

21. How to Define a Class in Python? What Is Self? Give An Example Of A Python Class

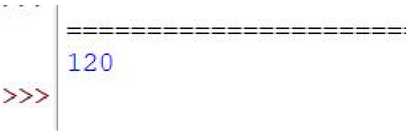
-> Objects are instances of a class, and they can have attributes (characteristics) and methods (functions) associated with them. The self parameter is used to refer to the instance of the class itself within its methods.

22. Write a Python class named Rectangle constructed by a length and width and a method which will compute the area of a rectangle.

Code:

```
class Rectangle():
    def __init__(self, l, w):
        self.length = l
        self.width  = w

    def rectangle_area(self):
        return self.length*self.width

newRectangle = Rectangle(12, 10)
print(newRectangle.rectangle_area())
```

Output:



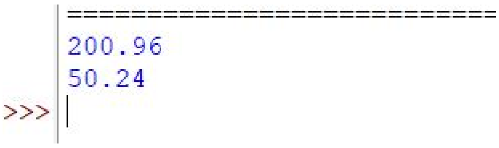
```
=====================
120
>>>
```

23. Write a Python class named Circle constructed by a radius and two methods which will compute the area and the perimeter of a circle.

Code:

```
class Circle():
   def __init__(self, r):
      self.radius = r

   def area(self):
      return self.radius**2*3.14

   def perimeter(self):
      return 2*self.radius*3.14

NewCircle = Circle(8)
print(NewCircle.area())
print(NewCircle.perimeter())
```

Output:



```
==========================
200.96
50.24
>>> |
```

24. Explain Inheritance in Python with an example? What is init? Or What Is A Constructor In Python?

-> Inheritance means creating new class from existing class is called inheritance.

-> In Python, __init__ is a special method, It is a constructor method that is automatically called when an object is created from a class. The primary purpose of the __init__ method is to initialize the attributes of the object.

-> a constructor is a special method in a class that is automatically called when an object of the class is created. The purpose of a constructor is to initialize the attributes or properties of the object

25. What is Instantiation in terms of OOP terminology?

->Instantiation describes the processes of creating a new object for a class using a new keyword.

26. What is used to check whether an object o is an instance of class A?

-> The isinstance () function checks whether an object is an instance of the class mentioned.

27. What relationship is appropriate for Course and Faculty?

-> Association

28. What relationship is appropriate for Student and Person?

-> Inheritance