FILE HANDLING AND ERROR HANDALING In python

FILE HANDALING

```
#file operation
#opening a file
file=open("/content/megha.txt", "r")
```

#**reading** file

OPENING

```
file=open("/content/megha.txt","r")
content = file.read()
print(content)
file.close()



file=open("/content/megha.txt","w")
file.write("hello,everyone!\n")
file.close()

file=open("/content/megha.txt","a")
file.write("my name is megha i from siet.\n")
file.close()

file=open("/content/megha.txt","a")
file.write("my name is megha i from siet.\n")
file.close()
```

```
hello,everyone!
my name is megha i from siet
```

```


#file handaling modules
with open("/content/download (1).jpg","rb")as file:
  data=file.read()
```

ERROR HANDALING

```
#TRY EXCEPT BLOCK
try:
  num=int(input("enter a number:"))
  print(10/num)
except ZeroDivisionError:
  print("you cannot divide by zero.")
except valueerror:
```

```
  print("invalid input!please enter a number")


enter a number:0
you cannot divide by zero.

#finallly block
try:
  file=open("/content/megha.txt","r")
except filenotfounderror:
  print("filenotfound.")
finally:
  print("execution complete.")

execution complete.

#raising exeptions
def check_age(age):
  if age<18:
    raise ValueError('age must be 18 or older.')
  return true

try:
  check_age(16)
except ValueError as e:
  print(e)

age must be 18 or older.
```

HAND ON PRACTICE

READING AND WRITEING A FILE

```
with open("/content/megha.txt","w")as file:
  file.write("python is awesome!\n")
with open("/content/megha.txt","w")as file:
  print(file.read())

---------------------------------------------------------------------------
-----
UnsupportedOperation                      Traceback (most recent call
last)
<ipython-input-38-bea2b2f5cac1> in <cell line: 3>()
      2   file.write("python is awesome!\n")
      3 with open("/content/megha.txt","w")as file:
----> 4   print(file.read())

UnsupportedOperation: not readable

#appending data to afile
with open("/content/megha.txt","a") as file:  # Changed mode to 'a'
```

```
for appending
  file.write("Let's a learn file handaling.\n")
with open("/content/megha.txt","r") as file:
  print(file.read())
```

```
Let's a learn file handaling.
```

```
try:
  num1=int(input("enter numerator:"))
  num2=int(input("enter denomenator:"))
  result=num1/num2
  print("Result",result)
except ZeroDivisionError:
  print("cannot devide a zero.")
except ValueError:
  print("invalid input! enter a numerical value")
```

```
enter numerator:5
enter denomenator:10
Result 0.5
```

```
#creating a custom exception
class NegativeNumberError(Exception):
  def check_positive(number): # Indented the function definition
    if number < 0:  # Corrected the condition to check the 'number'
variable
      raise NegativeNumberError("negative number entered.")
try:
  num = int(input("enter the positive nyumber"))
  NegativeNumberError.check_positive(num)  # Call check_positive using
the class name
  print("you entered a positive number")
except NegativeNumberError as e:
  print(e)  # Indented the print statement within the except block
```

```
enter the positive nyumber6
you entered a positive number
```

```
import ramdom
random_number=random.radient(1,6):
 print("random number is:" radident)
```

```
  File "<ipython-input-59-e7600a6fe6cf>", line 2
    random_number=random.radient(1,6):
                                      ^
SyntaxError: invalid syntax
```

pdf question

```
# 1. Create and Write to a File
with open("/content/megha.txt", "w") as file:
    file.write("Hello, World!\n")

# 2. Read from a File
with open("/content/megha.txt", "r") as file:
    content = file.read()
    print("File content:\n", content)
```

```
File content:
 Hello, World!
```

```
# 3. Append to a File
with open("/content/megha.txt", "a") as file:
    file.write("Welcome to Python programming!\n")

# 4. Count Lines in a File
with open("/content/megha.txt", "r") as file:
    lines = file.readlines()
    line_count = len(lines)
    print("Number of lines:", line_count)
```

```
Number of lines: 2
```

```
# 6. Copy File Contents
with open("/content/megha.txt", "r") as source_file, open("copy.txt",
"w") as dest_file:
    dest_file.write(source_file.read())
    print("File copied successfully!")
```

```
File copied successfully!
```

```
# 5. Count Words in a File
with open("/content/megha.txt", "r") as file:
    content = file.read()
    words = content.split()
    word_count = len(words)
    print("Number of words:", word_count)
```

```
Number of words: 6
```

```
# 7. Check if File Exists
import os
if os.path.exists("/content/megha.txt"):
    print("/content/megha.txt exists")
else:
    print("/content/megha.txt does not exist")
```

```
/content/megha.txt exists
```

```
# 8. Read File Line by Line
with open("/content/megha.txt", "r") as file:
    for line in file:
        print(line, end="")  # end="" to avoid extra newline
```

```
Hello, World!
Welcome to Python programming!
```

```
# 9. Search for a Word in a File
with open("/content/megha.txt", "r") as file:
    for line in file:
        if "Python" in line:
            print(line, end="")
```

```
Welcome to Python programming!
```

```
# 10. Write a List to a File
numbers = [1, 2, 3, 4, 5]
with open("numbers.txt", "w") as file:
    for number in numbers:
        file.write(str(number) + "\n")
    print("List written to numbers.txt")
```

```
List written to numbers.txt
```

```
import os
from collections import Counter
# 11. Reverse File Contents
with open("/content/megha.txt", "r") as file, open("reverse.txt", "w")
as reversed_file:
    lines = file.readlines()
    reversed_file.writelines(reversed(lines))
    print("File reversed and saved to reverse.txt")
```

```
File reversed and saved to reverse.txt
```

```
# 11. Reverse File Contents
with open("/content/megha.txt", "r") as file, open("reverse.txt", "w")
as reversed_file:
    lines = file.readlines()
    reversed_file.writelines(reversed(lines))
    print("File reversed and saved to reverse.txt")
```

```
File reversed and saved to reverse.txt
```