```
# Starting at 9:05 pm
# Enjoy the song :)
```

```
'''
Agenda
    •   Why we need file handling
    •   How files are stored in a computer (basic idea)
    •   File modes (r, w, a, etc.)
    •   Opening and closing files
    •   Reading from a file
    •   Writing to a file
    •   Introduction to exception handling (try / except)
    •   Handling file errors safely
    •   The with statement (automatic closing)
    •   Practice quizzes
'''
```

```
#File Handling
-> Python program ends, all variables vanish from the memory (RAM)
-> How to store data permanently? Using files

#How Python stores files
-> Hard Disk, SSD they store a sequence of bytes
-> Every file has a name and path
-> Text is stored using encoding (eg ASCII, UTF-8)
```

```
import sys
encoding=sys.getdefaultencoding()
print(encoding)
```

```
utf-8
```

Computers dont understand letters - only numbers -> 0 and 1

The rulebook to match each character to a unique number -> encoding

ASCII

A -> 65

B -> 66

a-> 97

In UTF- 8 -> emojis, arabic characters, chinesese +ASCII

```
#File -> User A , User B -> read -> this is okay
#File -> User A -> write , User B -> read -> can cause problems
```

```
#File Access Modes
```

```
'''
Mode    Full Name           Cursor Position   what happens if no file?
r       Read Only           Start of file     Error
r+      Read and Write      Start of file     Error
(both the modes dont create a new file)

w       Write Only          Start of file     Create a new file
if the file is present, overwrites the content of the old file
```

```
w+      Write and Read      Start of file     Create a new file
if the file is present, overwrites the content of the old file

a       Append Only         End of file       Create a new file
Keeps the old content, add the new at the end

a+      Append and Read     End of file       Create a new file
Keeps the old content, add the new at the end
```

```
# Opening and Closing the file
file=open("sample1.txt",'r')
```

```
---------------------------------------------------------------------------
FileNotFoundError                         Traceback (most recent call last)
/tmp/ipython-input-4169781318.py in <cell line: 0>()
      1 # Opening and Closing the file
----> 2 file=open("sample1.txt",'r')

FileNotFoundError: [Errno 2] No such file or directory: 'sample1.txt'
```

Next steps: Explain error

```
file=open("sample1.txt",'w') # a new file named sample1 has been created
```

```
file.close()
# Always close a file to save resources and avoid corruption
```

```
# Writing in a file

'''
 write()
 -> Writes a single string to the file
 -> It places the given text in the file starting at the
 current cursor position.
 -> Returns the number of bytes(characters) written
 '''
```

```
file=open("sample1.txt","w+")
file.write("Hello Everyone from the file!")
file.close()
```

```
# writelines()
'''
-> Writes multiple lines
-> doesnt add new lines, you need to add \n
'''
```

```
file=open("sample2.txt","w+")
file.writelines(["1\n","2\n","3\n"])
file.close()
```

```
file =open("sample2.txt","r+")
print(file.read())
```

```
file.close()
```

```
1
2
3
```

```
file =open("sample1.txt","r+")
print(file.read())
file.close()
```

```
Hello Everyone from the file!
```

```
#reading from a file

'''
read() -> read the entire file
readline() -> Reads on line at a time
readlines() -> Reads all line into a list
'''
```

```
file =open("sample1.txt","r+")
content=file.read()
print(content)
c=file.read() #empty string
print(c)
file.seek(0) #reset the cursor
content1=file.read()
print(content1)
file.close()
```

```
Hello Everyone from the file!

Hello Everyone from the file!
```

```
file =open("sample2.txt","r+")
print(file.readline())
print(file.readline())
print(file.readline())
print(file.readline()) # no more lines to print
file.close()
```

```
1

2

3
```

```
file =open("sample2.txt","r+")
print(file.readlines())
file.close()
```

```
['1\n', '2\n', '3\n']
```

```
# efficient way of reading

#option 1 -> small files
```

```
file=open("sample2.txt","r+")
for line in file.readlines():
  print(line,end=" ") # 1\n 2\n 3\n
file.close()
```

```
1
 2
 3
```

```
#option 2 -> large files

file=open("sample2.txt","r+")
for line in file:
  print(line,end=" ") # 1\n 2\n 3\n
file.close()
```

```
1
 2
 3
```

```
#reading fixed number of characters
file= open("sample3.txt","w+")
file.writelines(["this is a line\n","this is some random second line\n","this is my tl
file.close()
```

```
file= open("sample3.txt","r+")
while True:
  sentence=file.read(5) #reads 5 charcters at a time
  if not sentence:
    break
  print(sentence)
file.close()
```

```
this
is a
line

this
is so
me ra
ndom
secon
d lin
e
thi
s is
my th
ird l
ine
```

BREAK: 10:07 pm - 10:17 pm

```
#Quiz 1
file = open("demo.txt", "w")
file.write("Welcome to Python.")
file.close()

file =open("demo.txt","r+")
print(file.read())
file.close()
```

```
file = open("demo.txt", "a")
file.write("Hello, World!")
file.close()

file =open("demo.txt","r+")
print(file.read())
file.close()
```

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

```
# Quiz 2

When a file is opened in r+ mode, you first read some content from it.
If you then write new data without moving the cursor,
what will happen to the existing content in the file?

•   The new data will overwrite the existing content
starting from the current cursor position.
```

```
# Quiz 3
What happens if you open a file in "r" mode but it doesn't exist?
-> Raises an error -> FILENOTFOUNDERROR
```

```
# Quiz 4
Close the file automatically
- with open
```

```
#Introduction to Exception Handling
When something goes wrong, python stops and
shows error message and this msg is exception
```

```
file =open("does_not_exist.txt","r+")
print(file.read()) #FileNotFoundError
file.close()
```

```
---------------------------------------------------------------------------
FileNotFoundError                         Traceback (most recent call last)
/tmp/ipython-input-446249956.py in <cell line: 0>()
----> 1 file =open("does_not_exist.txt","r+")
      2 print(file.read())
      3 file.close()

FileNotFoundError: [Errno 2] No such file or directory: 'does_not_exist.txt'
```

Next steps: Explain error

```
# try and catch blocks

try: #Attempts the risky code
  file =open("does_not_exist.txt","r+")
  print(file.read()) #FileNotFoundError
  file.close()

except FileNotFoundError: #if the risky code fails, do this instead
  print("Oops! The file does not exist.")
```

```
Oops! The file does not exist.
```

```
try:
  file=open("notes.txt",'w')
  file.write("This is my first saved note!")
  file.close()
  print("File Written Succesfully")
except:
  print("Something went wrong with the file")
```

```
File Written Succesfully
```

```
try:
  file=open("notes.txt","r+")
  content=file.read()
  print("File Content:",content)
  file.close()
except FileNotFoundError:
  print("File does not exist. Please recheck the name.")
except PermissionError:
  print("You dont have permission to read the file")
```

```
File Content:
```

```
file=open("notes1.txt",'w')
file.write("This is my second saved note!")
file.close()
```

```
# opening the file smartly

try:
  with open("notes1.txt","r") as file:
    print("File Content:",file.readline())
    file.seek(0)
    print("reset the cursor n read:",file.readline())

except FileNotFoundError:
  print("File does not exist. Please recheck the name.")
except PermissionError:
  print("You dont have permission to read the file")
```

```
File Content: This is my second saved note!
reset the cursor n read: This is my second saved note!
```