**Q1: Which function is used to open a file? What are the different modes of opening a file? Explain each mode of file opening.**

The open() function is used to open a file in Python. The syntax is:

file\_object = open(file\_name, mode)

**Modes of Opening a File:**

1. **r (Read)**: Opens a file for reading (default mode). The file must exist; otherwise, an error is raised.
   * Example: open('file.txt', 'r')
2. **w (Write)**: Opens a file for writing. If the file exists, it is truncated to zero length. If it doesn’t exist, a new file is created.
   * Example: open('file.txt', 'w')
3. **x (Exclusive Creation)**: Creates a new file. If the file already exists, an error is raised.
   * Example: open('file.txt', 'x')
4. **a (Append)**: Opens a file for appending. Data is added at the end of the file. If the file doesn’t exist, it is created.
   * Example: open('file.txt', 'a')
5. **b (Binary Mode)**: Opens a file in binary mode.
   * Example: open('file.txt', 'rb')
6. **t (Text Mode)**: Opens a file in text mode (default).
   * Example: open('file.txt', 'rt')
7. **+ (Read and Write)**: Opens a file for both reading and writing.
   * Example: open('file.txt', 'r+')

**Q2: Why is the close() function used? Why is it important to close a file?**

The close() function is used to close a file that has been opened. It is important because:

* It releases the resources associated with the file.
* It ensures all buffered data is written to the file.
* It prevents potential data corruption or loss.
* It avoids file access issues or errors during subsequent operations.

**Q3: Write a Python program to create a text file, write a string to it, close it, and then read it.**

# Create and write to the file

file = open("data\_scientist.txt", "w")

file.write("I want to become a Data Scientist")

file.close()

# Open and read the file

file = open("data\_scientist.txt", "r")

content = file.read()

file.close()

print(content)

**Q4: Explain read(), readline(), and readlines() with Python code.**

1. **read()**: Reads the entire file or specified number of characters.
2. file = open("example.txt", "r")
3. print(file.read()) # Reads the whole file
4. file.close()
5. **readline()**: Reads one line at a time.
6. file = open("example.txt", "r")
7. print(file.readline()) # Reads the first line
8. file.close()
9. **readlines()**: Reads all lines and returns them as a list.
10. file = open("example.txt", "r")
11. print(file.readlines()) # Returns a list of lines
12. file.close()

**Q5: Why is the with statement used with open()?**

The with statement is used to handle file operations more efficiently by ensuring the file is automatically closed after its block is executed, even if an exception occurs.

**Advantages:**

* Automatic file closure.
* Cleaner and more readable code.
* Reduces the risk of forgetting to close the file.

with open("example.txt", "r") as file:

content = file.read()

print(content)

# File is automatically closed here.

**Q6: Explain the write() and writelines() functions with examples.**

1. **write()**: Writes a single string to a file.
2. with open("example.txt", "w") as file:
3. file.write("Hello, World!")
4. **writelines()**: Writes a list of strings to a file.
5. lines = ["Hello, World!\n", "Python is great.\n", "File handling is easy."]
6. with open("example.txt", "w") as file:
7. file.writelines(lines)