

## BCA – 502: Python Programming

Rahul Kumar Singh

In today's Class we have discussed on different file read and write functions used in Python.

Understanding read functions, `read()`, `readline()` and `readlines()`

**`read()` method:-**

The `read()` method returns the specified number of bytes from the file. Default is -1 which means the whole file.

**Syntax:-**

```
file.read()
```

Here size is optional. The number of bytes to return. Default -1, which means the whole file.

**Example:-1**

```
f = open("demofile.txt", "r")
```

```
print(f.read())
```

**Output:-**

Hello! Welcome to demofile.txt

This file is for testing purposes.

Good Luck!

## **Example:-2**

```
f = open("demofile.txt", "r")
```

```
print(f.read(33))
```

### **Output:-**

Hello! Welcome to demofile.txt

Th

## **readline() Method:-**

The readline() method returns one line from the file. You can also specified how many bytes from the line to return, by using the size parameter.

### **Syntax:-**

```
file.readline(size)
```

Here size is optional. The number of bytes from the line to return. Default -1, which means the whole line.

## **Example:-1**

```
f = open("demofile.txt", "r")
```

```
print(f.readline())
```

### **Output:-**

Hello! Welcome to demofile.txt

### **Example:-2**

```
f = open("demofile.txt", "r")  
print(f.readline())  
print(f.readline())
```

### **Output:-**

Hello! Welcome to demofile.txt  
This file is for testing purposes.

### **Example:-3**

```
f = open("demofile.txt", "r")  
print(f.readline(5))
```

### **Output:-**

Hello

### **readlines() Method:-**

The readlines() method returns a list containing each line in the file as a list item.

Use the hint parameter to limit the number of lines returned. If the total number of bytes returned exceeds the specified number, no more lines are returned.

### **Syntax:-**

```
file.readlines(hint)
```

Here hint is Optional. If the number of bytes returned

exceed the hint number, no more lines will be returned. Default value is -1, which means all lines will be returned.

### **Example:-1**

```
f = open("demofile.txt", "r")  
print(f.readlines())
```

### **Output:-**

```
['Hello! Welcome to demofile.txt\n', 'This file is for testing  
purposes.\n', 'Good Luck!']
```

### **Example:-2**

```
f = open("demofile.txt", "r")  
print(f.readlines(33))
```

### **Output:-**

```
['Hello! Welcome to demofile.txt\n', 'This file is for testing  
purposes.\n']
```

### **readable() Method:-**

The readable() method returns True if the file is readable, False if not.

### **Syntax:-**

```
file.readable()
```

### **Example:-**

```
f = open("demofile.txt", "r")  
print(f.readable())
```

### **Output:-**

True

## **Understanding write functions, write() and writelines()**

### **write() Method:-**

The write() method writes a specified text to the file. Where the specified text will be inserted depends on the file mode and stream position.

**"a":** The text will be inserted at the current file stream position, default at the end of the file.

**"w":** The file will be emptied before the text will be inserted at the current file stream position, default 0.

### **Syntax:-**

```
file.write(byte)
```

Here byte is The text or byte object that will be inserted.

### **Example:-1**

```
f = open("demofile2.txt", "a")
```

```
f.write("\nSee you soon!")
```

```
f.close()
```

#open and read the file after the appending:

```
f = open("demofile2.txt", "r")
```

```
print(f.read())
```

### **Output:-**

Hello! Welcome to demofile2.txt

This file is for testing purposes.

Good Luck!

See you soon!

### **Example:-2**

```
f = open("demofile2.txt", "w")
```

```
f.write("\nSee you soon!")
```

```
f.close()
```

#open and read the file after the appending:

```
f = open("demofile2.txt", "r")
```

```
print(f.read())
```

## **Output:-**

See you soon

## **writelines() Method:-**

The writelines() method writes the items of a list to the file.

Where the texts will be inserted depends on the file mode and stream position.

**"a":** The texts will be inserted at the current file stream position, default at the end of the file.

**"w":** The file will be emptied before the texts will be inserted at the current file stream position, default 0.

## **Syntax:-**

```
file.writelines(list)
```

Here list is The list of texts or byte objects that will be inserted.

## **Example:-**

```
f = open("demofile3.txt", "a")
```

```
f.writelines(["See you soon!", "Over and out."])
```

```
f.close()
```

#open and read the file after the appending:

```
f = open("demofile3.txt", "r")
```

```
print(f.read())
```

### **Output:-**

Hello! Welcome to demofile2.txt

This file is for testing purposes.

Good Luck!See you soon!Over and out.

### **Example:-2**

```
f = open("demofile3.txt", "a")
```

```
f.writelines(["\nSee you soon!", "\nOver and out."])
```

```
f.close()
```

#open and read the file after the appending:

```
f = open("demofile3.txt", "r")
```

```
print(f.read())
```

### **Output:-**

Hello! Welcome to demofile3.txt

This file is for testing purposes.

Good Luck!See you soon!Over and out.

### **writable() Method:-**

The writable() method returns True if the file is writable, False if not.



A file is writable if it is opened using "a" for append or "w" for write.

**Syntax:-**

```
file.writable()
```

**Example:-**

```
f = open("demofile.txt", "a")
```

```
print(f.writable())
```

**output:-**

```
True
```