
System Requirements Specification Index

For

Core Python File handling in python

Version 1.0

Problem Statement : To perform file operations and file manipulation .

Description : Use relevant methods of file operations to perform specified activities which are given in the instructions.

The Template contains the following folder structure.

```
File handling |
               |--file_ex.py
               |--file_read.py
               |--test (contains unit test case files for the solution)
```

File_ex.py:

Implement all the methods as specified in the following instructions.

1. Display the root name and the extension of the given file?
2. What is the size of the file inventory.txt in bytes?
3. How many words are in the file inventory.txt?
4. Does the current user have write permission for the file inventory.txt?
5. What is the size of the image file flower.jpg in kilobytes?

File_read.py

1. How can you display the names from emp.txt that start with the letter 'S'?
2. How can you display the names from emp.txt with a salary above 94000?
3. How can you find and display the names from emp.txt that have a Yahoo email address?
4. How can you print the seventh line from the file emp.txt?
5. How can you count the total number of names in the file emp.txt?

Execution Steps to Follow:

1. All actions like build, compile, running application, running test cases will be through Command Terminal.
2. To open the command terminal the test takers, need to go to Application menu (Three horizontal lines at left top) -> Terminal -> New Terminal
3. This editor Auto Saves the code
4. If you want to exit(logout) and continue the coding later anytime (using Save & Exit option on Assessment Landing Page) then you need to use CTRL+Shift+B-command compulsorily on code IDE. This will push or save the updated contents in the internal git/repository. Else the code will not be available in the next login.
5. These are time bound assessments the timer would stop if you logout and while logging in back using the same credentials the timer would resume from the same time it was stopped from the previous logout.
6. To setup environment:

You can run the application without importing any packages

7. To launch application:

```
python3 file_ex.py
```

```
python3 file_ready.py
```

8. To run Test cases:

```
python3 -m unittest
```

9. Before Final Submission also, you need to use CTRL+Shift+B-command compulsorily on code IDE. This will push or save the updated contents in the internal git/repository for code quality analysis graph.

-----X-----