|  |  |  |
| --- | --- | --- |
| **LAB211 Assignment** | **Type:** | **Short Assignment** |
| **Code:** | **J1.S.P0075** |
| **LOC:** | **100** |
| **Slot(s):** | **2** |

**Title**

Handle file program (extraction from CBDT project)

**Background**

NA

**Program Specifications**

Develop a file processing program include functions:

1. .Enter the path and then check it is exist or not? If path exist, that is file path or directory path?
2. Enter a directory path and then list all file with .java extension.
3. Enter a directory path and an integer n(KB). Find all the files with size > n in the directory and print to the screen.
4. Enter file path then insert more content from the keyboard.
5. Enter a file txt path Count the number of word in the file (each word is separated by a whitespace)

***Function details:***

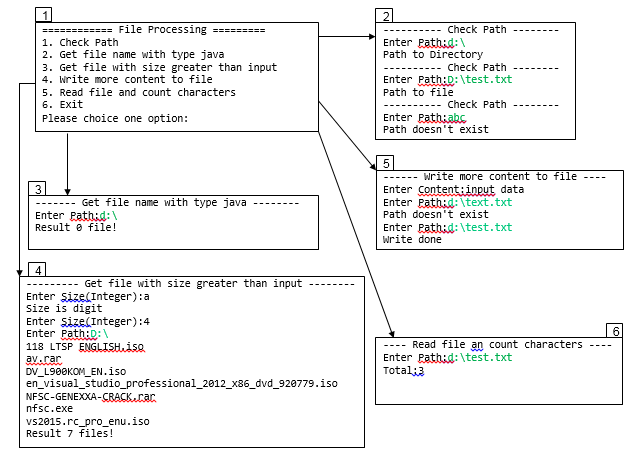
**Function 1:** Display a menu and ask users to select an option.

* Users run the program. The program prompts users to select an option.
* Users select an option, perform **Function** **2**.

**Function 2:** Perform function based on the selected option..

* Option 1: Check the path
* Enter a path and check the path is exits or not ?
* If not, display a notice : “Path doesn’t exist”
* . If path exist, display a notice that is file path or directory path?
* Option 2: Listed file .java
* Enter a path and check the path is exits
* If not, display a notice : “Path doesn’t exist”
* Display quantity of .java file
* Option 3: List all files with file size larger than input size
* Enter size n(KB) , Check n must be numeric , if it is the other type then display notice: “Value of size is digit”
* Enter a path
* If not exist display a notice : “Path doesn’t exist”, Please Try again.
* Display quantities of file and print file name with size > n in the folder.
* Option 4: Enter file path then add more content from the keyboard.
* Input content from keyboard that the user want to add to file
* Enter file path
* If not exist display a notice: “Path doesn’t exist” Please Try again.
* Notice that successfully add content inputted from keyboard into file
* Option 5: Count word in .text file
* Enter a path to.txt file
* Print to screen total word of file
* Option 6: Exit program

### *Expectation of User interface:*



**Guidelines**

**Student must implement methods**

checkInputPath

getAllFileNameJavaInDirectory

getFileWithSizeGreaterThanInput

appendContentToFile

countCharacter

**in startup code.**

Use the classes:

java.io.BufferedReade,java.io.BufferedWriter,java.io.File, java.io.FileFilter, java.io.FileReader, java.io.FileWriter, java.io.FilenameFilter, java.io.IOException, java.util.ArrayList, java.util.List;

to file manipulation.

**Function 1:** Check the path

* Implement function: public void checkInputPath(String path) throws Exception
  + Input:
* path: File path or Directory path
  + Return value:
* Exception("Path doesn't exist").
* Exception("Path to file").
* Exception("Path to Directory").

**Function 2:** List all the java file .

* Implement function: List<String> getAllFileNameJavaInDirectory(String path) throws Exception
  + Input:
* path: File path
  + Return value:
* List of file name.
* Exception("Path doesn't exist").

**function 3:** Search file with file size lager than n

* Implement function: public static File[] getFileWithSizeGreaterThanInput(String path, int size) throws Exception.
  + Input:
* path: File path
* size: file size
  + Return value:
* List of file.
* Exception("Path doesn't exist").

**function 4:** Add content from keyboard.

* Implement function: public boolean appendContentToFile(String path, String contentInput) throws Exception.
  + Input:
* path:File path
* contentInput: inputted content from keyboard
  + Return value:
* Recording status file.
* Exception("Path doesn't exist").

**function 5:** Count the number of character which are separated by a whitespace in file.

* Implement function: public int countCharacter(String path) throws Exception.
  + Input:
* path: file path
  + Return value:
* Number of character.
* Exception("Path doesn't exist").