Node.JS

Practice – 3

Overview

In this hands-on session you will continue learning ***Node.js*** *common core modules. Upon completion of this hands-on session, you should be able to:*

- *Use the Node core module – fs, os, path, event, stream, buffer etc.*
- *Perform file operations like:*
  - *Read from a file, Write or Append to a file*
  - *Read and traverse a directory*
- *Use EventEmitter to work with user-defined events*
  - *Define, register and trigger events*
- *Use Streams for processing*
  - *Chunk of data*
  - *Piping stream process*
- *Use 'crypto' for encryption and 'zlib' for compressing*

Hands-on

1. Write programs to read the content of a file.

   [a] Synchronously
   [b] Asynchronously

2. Write programs to write some content to a file.

   [a] Synchronously
   [b] Asynchronously

   Enhance the above program(s) where you had written some content to a file by making it append-able.

3. For a given absolute path for a file, get the following:

   - The directory names
   - The base name of the file
   - The extension of the file.

4. Write a program which display the following:

   [a] The platform on which Node.js is compiled

[b] The hostname
[c] The Operating System release number.

5. Write a program to find the following:

   [a] The total memory and the amount of free memory available in the system.
   [b] The number of CPU’s, the model name and speed.

6. Google out - **How to dive a car step-by-step?**

   Your challenge is to create an **EventEmitter** instance using the 'event' module

   Later, create event handlers for different steps
   Register event handlers to the events and trigger the events.

   On triggering the events the output should look like the steps you got from Google - How to drive a car step-by-step?

7. Create a readable stream which will enable the user to read the contents of a file.

8. When we can read chunks of data from a readable stream. If the chunk happens to be a single character, we can process character by character.

   Keeping this in mind, your challenge is to write functions for the following:

   [a] To count the number of Vowels and Consonants from the file
   NOTE: Consonants are only letters other than vowels

   [b] Percentage of vowels against the total number of letters
   Percentage of consonants against the total number of letters,

   Let your text file have a simple text so that you should be able to predict the output.

   Record your observation!

9. With the help of **piping** with streams, duplicate a file into two different copies.

10. Read the contents of the file and encrypt it using the **'crypto'** module.

    Use the '**SHA-512**' hashing algorithm to encrypt the data,

    Ensure that the encrypted data is shown on the standard output stream.

11. Implement a simple **'echo'** utility which as the name suggest will echo what is being typed on the standard input stream to the standard output stream

Processing should stop when the user presses the **'Ctrl+C'** keys.

12. It is known to us that the **'pipeline()'** method should be used instead of 'pipe()'

    Using the **'pipeline()'** method of the 'stream' module, compress the file.

    The **'zlib'** module can be used for compressing