Compressing and Decompressing Data using Zlib

Akash Pundir

System Programming –I

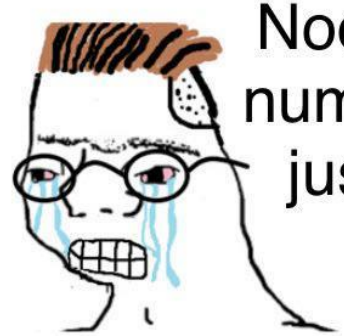
School of Computer Science and Engineering

zlib is **a built-in Node.js module that provides compression and decompression functionalities using the zlib library**, which is a general-purpose data compression library. It supports various compression algorithms.

- Here are some key features and functionalities of the zlib module:

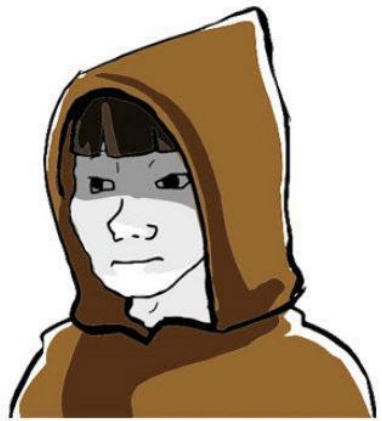
- **Compression**: zlib provides methods for compressing data using algorithms like Gzip and DEFLATE. These compression methods reduce the size of the data, making it suitable for transmission over networks or storage on disk.
- **Decompression**: It also supports decompressing compressed data using the same algorithms. This allows you to restore the original data from compressed representations.
- **Stream Interface**: zlib provides stream-based interfaces for both compression and decompression. This means you can process data incrementally, which is useful for large datasets or data streams.
- **Error Handling**: The module provides error handling mechanisms, allowing you to handle errors that may occur during compression or decompression operations.

Nooo compression isn't real, numbers are numbers, you're just re-structuring the data



Compression makes my files smaller

Compression makes my files smaller



Compression and Decompression using zlib

```
const zlib = require('zlib');

// Example data
const input = 'Hello, world!';

// Compress the data
zlib.gzip(input, (err, compressedData) => {
  if (err) {
    console.error('Error compressing data:', err);
    return;
  }

  // Decompress the data
  zlib.gunzip(compressedData, (err, decompressedData) => {
    if (err) {
      console.error('Error decompressing data:', err);
      return;
    }
    console.log('Decompressed data:', decompressedData.toString());
  });
});
```

Test your Knowledge

How can you create a Node.js server that serves a specific text file, compresses it with gzip encoding, and dynamically responds to HTTP requests? Provide a detailed code solution.

Import Necessary Files

```
const http = require('http');
const fs = require('fs');
const zlib = require('zlib');
```

Our Main Code

```
const server = http.createServer((req, res) => {
    const filePath = 'example.txt';
    const readStream = fs.createReadStream(filePath);

    res.writeHead(200, {
        'Content-Type': 'text/plain',
        'Content-Encoding': 'gzip' // Setting the content encoding to gzip
    });

    // Compressing the file and piping it to the response stream
    readStream.pipe(zlib.Gzip()).pipe(res);

    readStream.on('error', (err) => {
        console.error('Error reading file:', err);
        res.statusCode = 500;
        res.end('Internal Server Error');
    });
});
```

Finishing Touches….

```
const PORT = 3000;
server.listen(PORT, () => {
    console.log(`Server is running on port
${PORT}`);
});
```