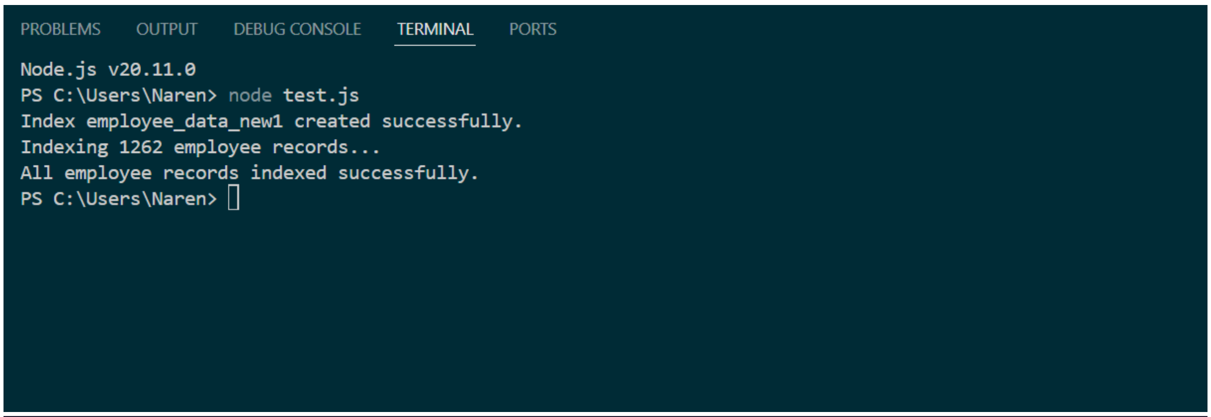
Solution

Code (JavaScript)

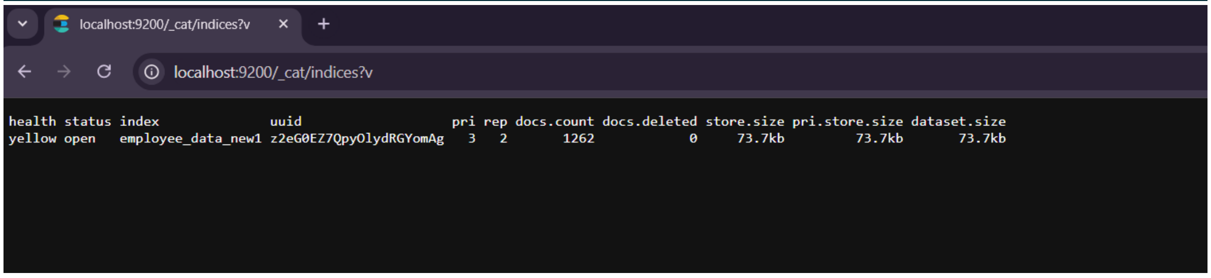
```
const { Client } = require('@elastic/elasticsearch');
const fs = require('fs');
const csv = require('csv-parser');
const client = new Client({
  node: 'http://localhost:9200',
  auth: {
    username: 'elastic',
    password: 'RIOOO'
  }
 });
const indexName = 'employee_data_new1';
  async function createIndex() {
    // Delete the index if it already exists
    const exists = await client.indices.exists({ index: indexName });
    if (exists.body) {
      await client.indices.delete({ index: indexName });
      console.log(`Index ${indexName} deleted.`);
    }
    await client.indices.create({
      index: indexName,
      body: {
        settings: {
          number_of_shards: 3,
          number_of_replicas: 2
        },
        mappings: {
          properties: {
            EmployeeID: { type: 'integer' },
            FirstName: { type: 'text' },
            LastName: { type: 'text' },
            Gender: { type: 'keyword' },
            Age: { type: 'integer' },
            Salary: { type: 'float' },
            Department: { type: 'text' },
            Position: { type: 'text' },
            DateOfHire: { type: 'date', format: 'yyyy-MM-dd' }
          }
        }
      }
    });
    console.log(`Index ${indexName} created successfully.`);
  }
async function loadDataToElasticsearch() {
  const bulkOperations = [];
  fs.createReadStream('C:\\Users\\Naren\\Downloads\\employee_data_new.csv')
  .pipe(csv())
  .on('data', (row) => {
    bulkOperations.push({
      index: { _index: indexName }
    });
    bulkOperations.push({
      EmployeeID: parseInt(row.EmployeeID),
      FirstName: row.FirstName,
      LastName: row.LastName,
      Gender: row.Gender,
      Age: parseInt(row.Age),
      Salary: parseFloat(row.Salary),
      Department: row.Department,
      Position: row.Position,
      DateOfHire: row.DateOfHire
    });
  })
  .on('end', async () => {
    console.log(`Indexing ${bulkOperations.length / 2} employee records...`);

    try {
      const bulkResponse = await client.bulk({ refresh: true, body: bulkOperations });
      if (bulkResponse && bulkResponse.errors) {
        console.error('Errors occurred during bulk indexing:', bulkResponse.errors);
      } else {
        console.log('All employee records indexed successfully.');
      }
    } catch (err) {
      console.error('Error during bulk indexing:', err);
    }
  });
}
(async () => {
  await createIndex();
  await loadDataToElasticsearch();
})();
```

Answer Screenshot :



```
Node.js v20.11.0
PS C:\Users\Naren> node test.js
Index employee_data_new1 created successfully.
Indexing 1262 employee records...
All employee records indexed successfully.
PS C:\Users\Naren>
```



```
localhost:9200/_cat/indices?v
```

```
health status index              uuid                   pri rep docs.count docs.deleted store.size pri.store.size dataset.size
yellow open   employee_data_new1 z2eG0EZ7QpyOlydRGYomAg   3   2       1262            0     73.7kb         73.7kb       73.7kb
```