# Node.js

Node.js is a runtime environment that allows JavaScript to be run on the server-side.

It is built on Chrome's V8 JavaScript engine and allows building scalable network applications.

Key features include event-driven architecture, non-blocking I/O, and asynchronous programming.

Example:

```javascript  
const http = require('http');  
const server = http.createServer((req, res) => {  
 res.statusCode = 200;  
 res.setHeader('Content-Type', 'text/plain');  
 res.end('Hello, World!');  
});  
server.listen(3000);  
```

# Express.js

Express.js is a minimal and flexible Node.js web application framework that provides robust features for building web and mobile applications.

It simplifies routing, middleware integration, and request handling.

Example:

```javascript  
const express = require('express');  
const app = express();  
app.get('/', (req, res) => {  
 res.send('Hello World');  
});  
app.listen(3000);  
```

# MongoDB

MongoDB is a NoSQL, document-oriented database that stores data in JSON-like documents (BSON).

It allows for flexible schema and horizontal scaling.

Key operations: CRUD, indexing, aggregation, population, etc.

Example CRUD operations:

```javascript  
// Connect using Mongoose  
mongoose.connect('mongodb://localhost:27017/mydb');  
  
// Schema definition  
const User = mongoose.model('User', { name: String, age: Number });  
  
// Create  
const user = new User({ name: 'Alice', age: 30 });  
user.save();  
  
// Read  
User.find({}, (err, users) => console.log(users));  
  
// Update  
User.updateOne({ name: 'Alice' }, { $set: { age: 31 } });  
  
// Delete  
User.deleteOne({ name: 'Alice' });  
```