**Express.js** is most popular Node.js web framework. It provides mechanisms to:

* Write handlers for request with different HTTP verbs at different URL paths (routes).
* Integrate with "view" rendering engines in order to generate responses by inserting data into templates.
* Set common web application settings like the port to use for connecting, and the location of templates that are used for rendering the response.
* Add additional request processing "middleware" at any point within the request handling pipeline.

**What does Express code look like?**

In a traditional data-driven website, a web application waits for HTTP requests from the web browser (or other client). When a request is received the application works out what action is needed based on the URL pattern and possibly associated information contained in POST data or GET data. Depending on what is required it may then read or write information from a database or perform other tasks required to satisfy the request. The application will then return a response to the web browser, often dynamically creating an HTML page for the browser to display by inserting the retrieved data into placeholders in an HTML template.

Express provides methods to specify what function is called for a particular HTTP verb (GET, POST, SET, etc.) and URL pattern ("Route"), and methods to specify what template ("view") engine is used, where template files are located, and what template to use to render a response. You can use Express middleware to add support for cookies, sessions, and users, getting POST/GET parameters, etc. You can use any database mechanism supported by Node (Express does not define any database-related behaviour).