WEEK-6

JADE TEMPLATE EXPRESS

1) Explain Jade Template with express.

You will discover how to use the Jade template engine in an Express.js-based Node.js application in this section.

Jade is a Node.js template engine. It's simple to learn Jade syntax. Its grammar makes use of whitespace and indentation.

Using NPM, add jade to your project as seen below.

npm install jade

The.jade file must contain the Jade template. Additionally, all.jade files must be placed in the views folder of the Node.js application's root folder.

Any template engine can be used with Express.js. Here, we'll employ a variety of Jade templates to generate dynamic HTML pages. Create a sample.jade file in the views folder and add the following Jade template inside to utilise Jade with Express.js.

Three fundamental traits of Jade

- Simple tags
- Giving the tags attributes
- Blocks of text

You may use CodePen and choose Jade as your HTML preprocessor to test this out as we go along, or you can use the online compiler on the official Jade page to convert your Jade to HTML.

You can use static template files in your application thanks to a template engine. The template engine converts a template file into an HTML file that is provided to the client at runtime by replacing variables with real values. An HTML page can be designed more easily using this method.

Pug, Mustache, and EJS are a few well-known template engines that support Express. The Express application generator supports a number of different languages in addition to Jade as its default.

For a list of template engines you can use with Express, go to Template Engines (Express wiki). See also Jade, Mustache, Dust, and Other JavaScript Templating Engines: Comparison.

The following application configuration properties need be set in app.js of the generator's default app in order to render template files:

views, the directory containing the template files. For instance, app.set('views', './views'). The application root directory's views directory is the default choice here.

The template engine to utilise is the view engine. For instance, app.set('view engine', 'pug') can be used to use the Pug template engine.

2) What is Template Engine. Give Example.

When you wish to quickly create web applications that are divided into separate components, template engines are employed. Additionally, templates make it possible for the server-side data that must be sent to the application to be rendered quickly.

You might, for instance, want to include elements like the body, navigation, footer, dashboard, etc.

The majority of server-side apps that are run on a single server and are not created as APIs employ template engines. Ejs, Jade, Pug, Mustache, HandlebarsJS, Jinja2, and Blade are some of the well-known ones.

When you use a template engine to create a server-side application, the engine replaces the variables in a template file with real values and shows the client this value. This makes it simpler to develop our application more quickly.

You can utilise a template engine for a server-side NodeJS runtime application.

The ejs template engine and expressJs are used to show how template engines function in the phases that follow. The sample below displays a user's information on a web page.

```
<html>
  <head>
    <title>This is the title</title>
  </head>

  <body>
    <h1>Welcome to User Details</h1>
    <p><b>Name:</b> <%= user.name %></p>
```

```
    <p><b>Email:</b> <%= user.email %></p>
    <p><b>Stack:</b> <%= user.stack %></p>
    <u><b>Hubbies</b></u>
    <% user.hubby.forEach(hubby =>{ %>
       <li><%= hubby %></li>
    <% })%>
  </body>
</html>
```

3) Define Express template.

You can use static template files in your application thanks to a template engine. The template engine converts a template file into an HTML file that is provided to the client at runtime by replacing variables with real values. An HTML page can be designed more easily using this method.

Pug, Mustache, and EJS are a few well-known template engines that support Express. The Express application generator supports a number of different languages in addition to Jade as its default.

For a list of template engines you can use with Express, go to Template Engines (Express wiki). See also Jade, Mustache, Dust, and Other JavaScript Templating Engines: Comparison.

You can use static template files in your apps thanks to a template engine. It changes variables in a template file into real values during runtime and converts the template into an HTML file that is provided to the client. Therefore, this method is ideal for creating HTML pages quickly.

The list of popular template engines compatible with Express.js is as follows:

- Pug (formerly known as jade)
- mustache
- dust
- ATPL
- ECO
- ECT

- EJS
- haml
- haml-coffee
- handlebars
- hogan
- jazz
- JQTPL
- JUST
- liquor
- QEJS
- swig
- templayed
- toffee
- underscore
- walrus
- whiskers

The pug (formerly known as jade) and moustache appear to be the most popular choices in the aforementioned template engines. Similar to Haml, which makes use of whitespace, is Pug. The template-benchmark shows that Pug is 2 times slower than Handlebars, EJS, and Underscore. The fastest seems to be ECT. One of the reasons why moustache template engine is popular among programmers is that it is one of the simplest and most adaptable template engines.

You can use static template files in your application thanks to the template engine. You must configure the following application configuration parameters in order to render template files:

Views: It designates the location of the template files' directory.

As an illustration, try app.set('views', './views').

the template engine you use is specified by the view engine. For instance, app.set('view engine', 'pug') can be used to use the Pug template engine.

Take a look at a template engine pug (formerly known as jade).

4) **Develop template engine. Using Jade templates to create HTML pages dynamically.**

While rendering static webpages from a server is technically conceivable, there are numerous drawbacks to this strategy, including code duplication and a lack of flexibility, particularly when receiving data from a database. Fortunately, Express.js gives us a means to use a template engine to generate dynamic HTML pages from our server-side applications.

A template engine operates in a surprisingly straightforward manner: you write a template and input variables into it using the proper syntax. You assign values to the variables declared in your template file at the proper route to render the template. These are continuously compiled when the template is being rendered.

The ability to build reusable parts known as partials that may be utilised in other files is one of the template engines' key features. This lessens the need for duplicating code and facilitates the implementation of updates.

There are numerous template engines on the market right now, but some of the more well-known ones are Pug (also known as Jade), Handlebars, EJS, Mustache, Swig, and others. The following template engines for Express are covered in this article:

- Pug
- EJS
- Handlebars