

## # Vuejs-Tutorial

This will help you learn Vuejs 2.0 with Example

Vuejs Tutorial With Example is today's main topic. Vue.js is quite famous front end library in nowadays. So It is worth to take a look at it. It is simple, minimal core with an incrementally adoptable stack that can handle apps of any scale. Vue (pronounced /vju:/, of view) is a progressive framework for building user interfaces. The core library is focused on the view layer only and is very easy to pick up and integrate with other libraries or existing projects. On the other hand, Vue is also perfectly capable of powering sophisticated Single-Page Applications.

Open up your command prompt in an Administrator mode

decide where you want to create the folder, select that directory path and change to that directory on your command prompt,

after that go to step 1

Step 1: Create one project folder.

Create one project folder, in my case VuejsTutorial.

**mkdir VuejsTutorial** - Go into that folder.

**cd VuejsTutorial** - Type following command to set up the package.json file.

**npm init** - Now, give the answers, and after that in your root folder, there is a created file called package.json

*the image named "VuejsTutorial" will show you a preview*

Now, give the answers, and after that in your root folder, there is a created file called **package.json**

*the image named "tutorial 1" will show you a preview*

Then create one **index.html** file in a root, which will be served by webpack the module bundle.

```
<!-- index.html -->

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>Vuejs Tutorial With Example</title>
  </head>
  <body>
  </body>
</html>
```

the image named "index.html" will show you a preview

Step 2: Make webpack.config.js file.

Now, we need to you create one first webpack build. So follow these steps.

Create a file called **webpack.config.js** in your root folder.

```
// webpack.config.js

module.exports = {
  // This is the "main" file which should include all other modules
  entry: './src/main.js',
  // Where should the compiled file go?
  output: {
    filename: 'bundle.js'
  },
}
```

```
resolve: {
  alias: {
    vue: 'vue/dist/vue.js'
  }
},
module: {
  // Special compilation rules
  loaders: [
    {
      // Ask webpack to check: If this file ends with .js, then apply some transforms
      test: /\.js$/,
      // Transform it with babel
      loader: 'babel-loader',
      // don't transform node_modules folder (which don't need to be compiled)
      exclude: /node_modules/
    }
  ]
}
}
```

the image named "webpack.config.js" will show you a preview

Create one file called **.babelrc** in your root folder.

Babel is the tool used to convert ES6 syntax into present day JavaScript which is ES5.

the image named ".babelrc" will show you a preview

Now install following development dependencies. (the image named "install development dependencies" will show you a preview)

```
npm install --save-dev babel-preset-es2015 babel-preset-stage-3 babel-core babel-loader babel-plugin-transform-runtime babel-runtime webpack webpack-dev-server
```

Next, we need to install webpack globally in our PC. (the image named "install webpack" will show you a preview)

```
npm install -g webpack webpack-dev-server
```

So now we need to install Vue library via NPM (the image named "install vue library" will show you a preview)

```
npm install --save vue
```

Create one folder in root called **src**. In that create one JavaScript file called **main.js**

In **index.html** file put the following code to include our bundle.js file. re edit the **index.html** file

```
<script src="bundle.js"></script>
```

So your index.html file will look like this.

```
<!-- index.html -->
```

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
<meta charset="utf-8">
```

```
<title>Vuejs Tutorial With Example</title>
```

```
</head>
```

```
<body>
  <div id="app">
    <h3>{{ message }}</h3>
  </div>
  <script src="bundle.js"></script>
</body>
</html>
```

Your **package.json** file will look like this. re edit your **package.json** file

```
{
  "name": "vuejstutorial",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "start": "webpack-dev-server"
  },
  "author": "",
  "license": "ISC",
  "devDependencies": {
    "babel-core": "^6.25.0",
    "babel-loader": "^7.1.1",
    "babel-plugin-transform-runtime": "^6.23.0",
    "babel-preset-es2015": "^6.24.1",
    "babel-preset-stage-3": "^6.24.1",
    "babel-runtime": "^6.25.0",
```

```
"webpack": "^3.4.1",  
"webpack-dev-server": "^2.6.1"  
},  
"dependencies": {  
  "vue": "^2.4.2"  
}  
}
```

Here, we have installed **webpack-dev-server** as a development dependency, so I include webpack-dev-server command in the **package.json** file.

So that when you type npm start then, It will start development server. When you change any .js file, then the server will automatically

restart. No need to manually do it.

before you start with npm start

edit your **package.json** file by removing the second scripts and leaving only

```
"scripts": {  
  "start": "webpack-dev-server"  
},
```

else you'll get a Start script missing error when running npm start

after that install your javascript babelrc plugins with

**npm install --save-dev babel-preset-es2015**

and then type

```
{  
  "presets": ["es2015"]  
}
```

into your **.babelrc** file and save

then you can Run the following command.

**npm start**

It will start at this URL: **http://localhost:8080/**

You can see on the screen: your message typed in your index.html file