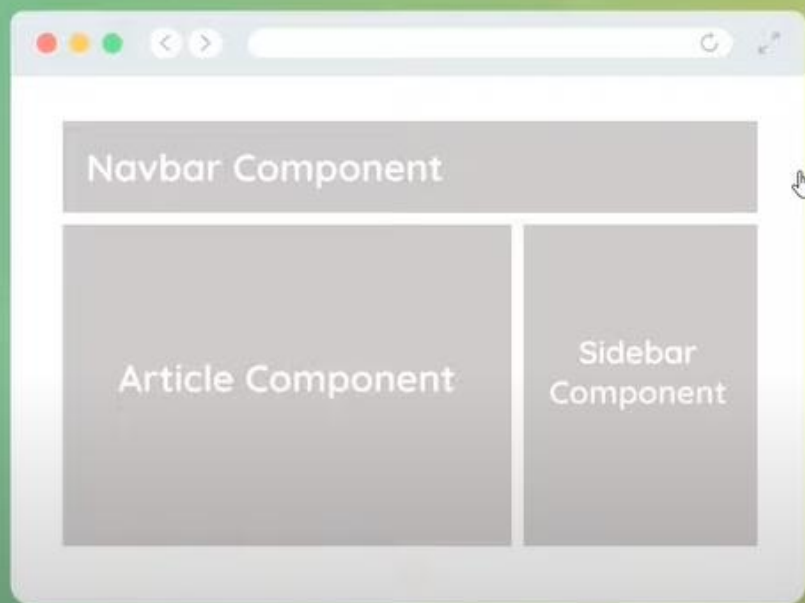


The Vue Router

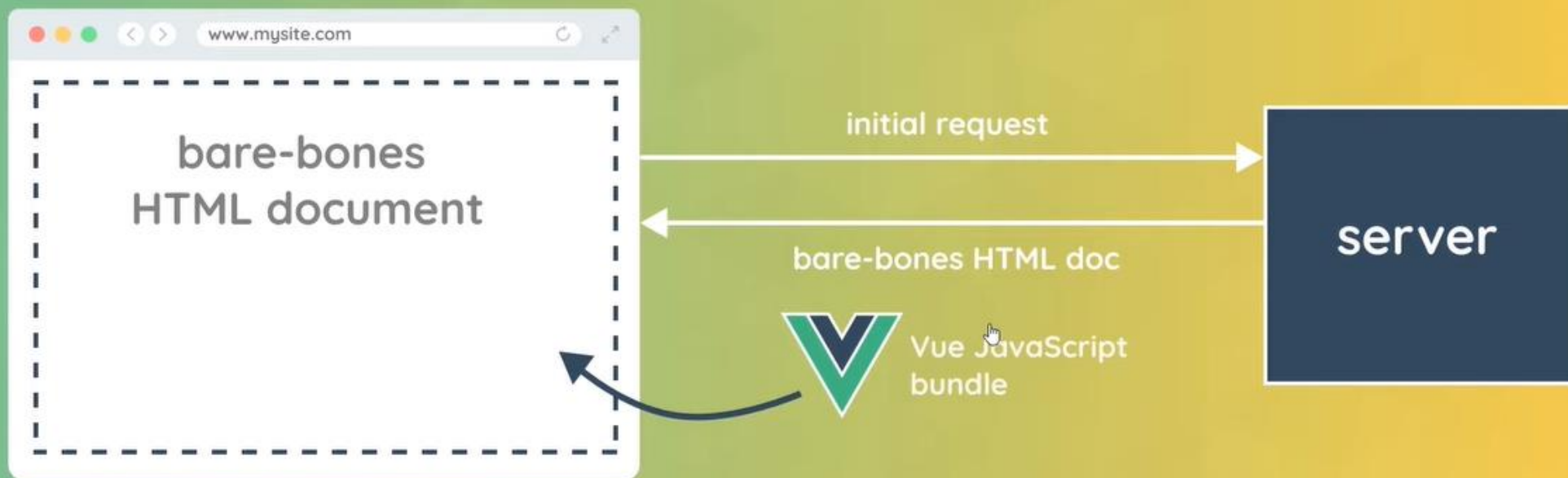
In this Vue 3 tutorial we'll take a look at the Vue Router in order to make a "multi-page" Vue website. We'll see how to set up the Vue Router, how to use Router Links, pass route parameters & more.



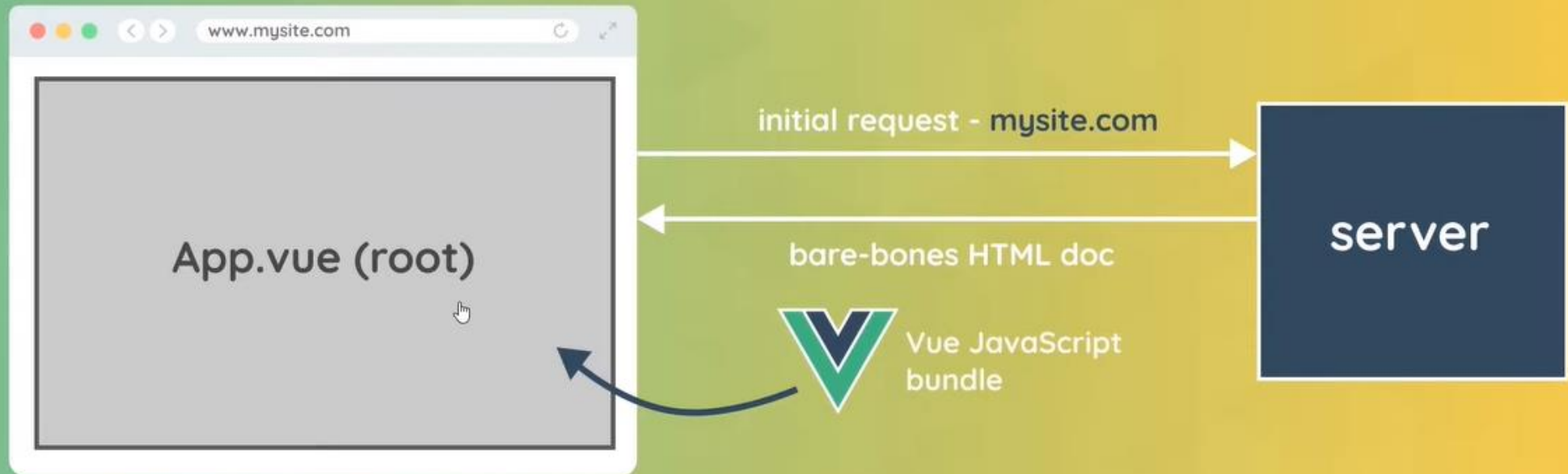
The Vue Router



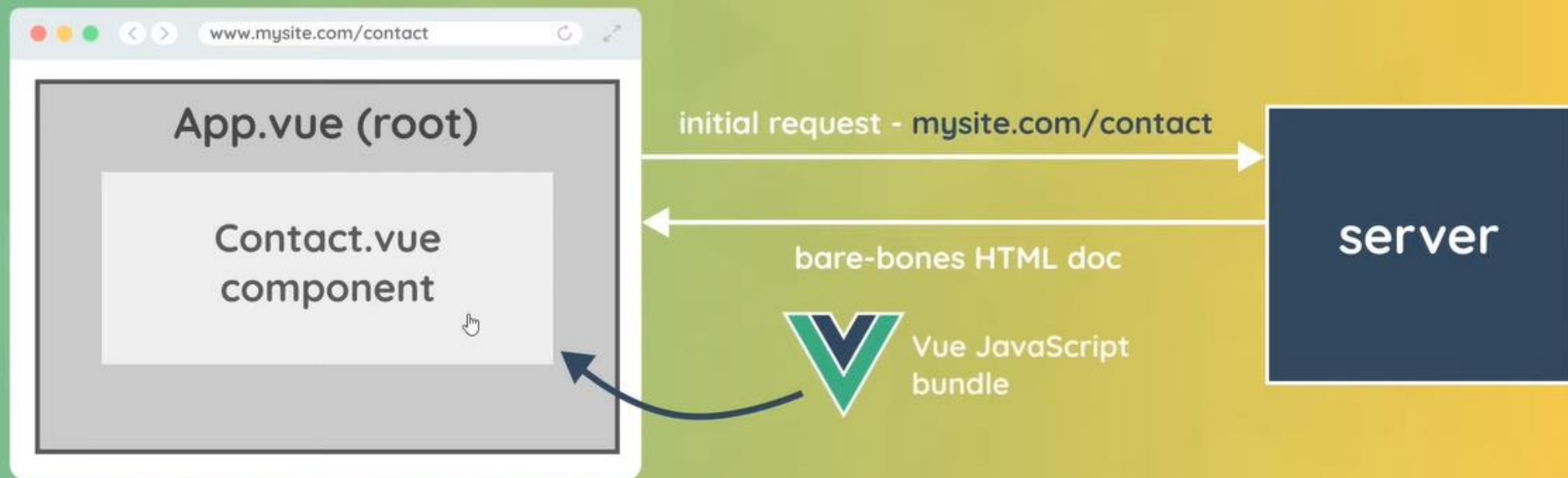
The Vue Router



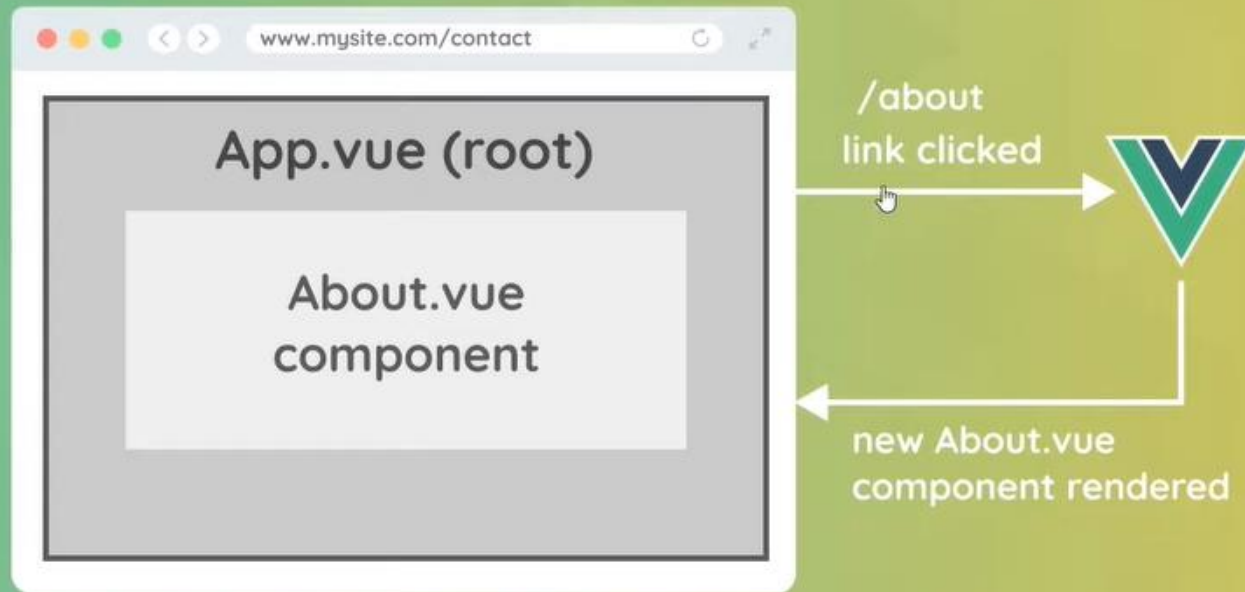
The Vue Router



The Vue Router



The Vue Router



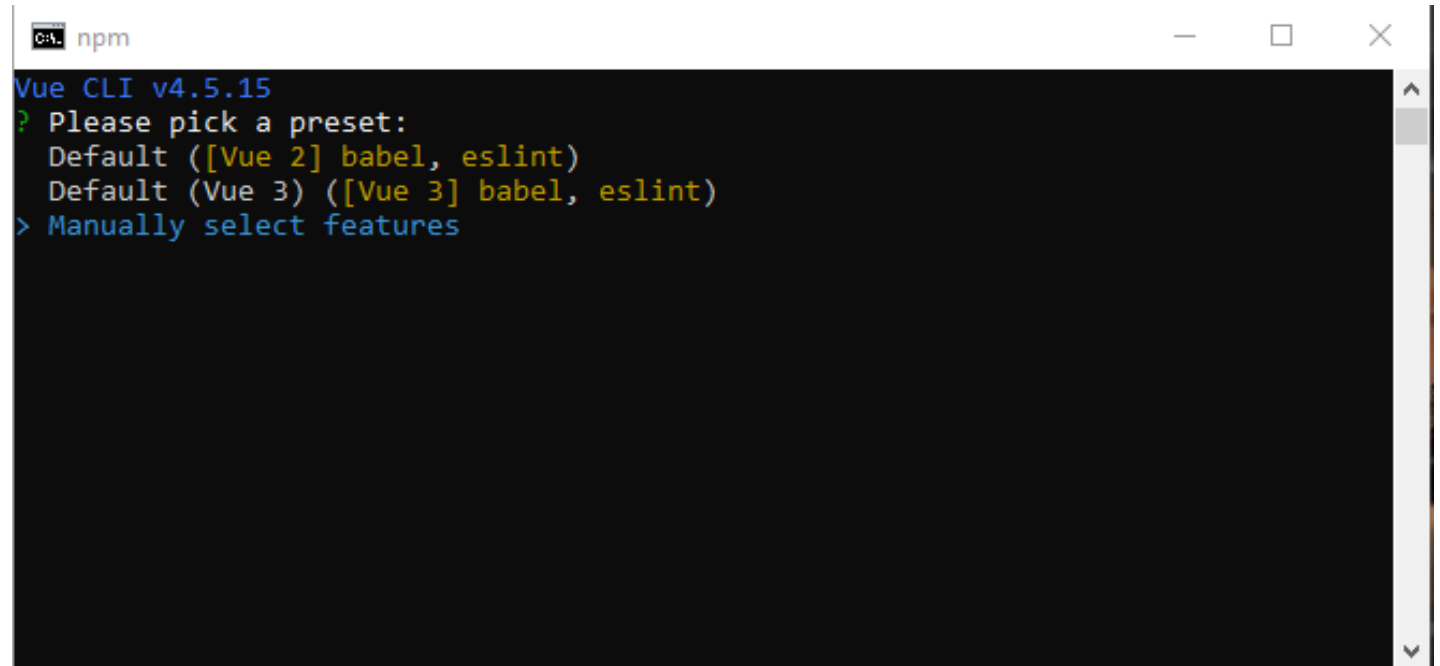


Vue Router Setup

Open cmd and write next commands:

- vue create

Choose row
'Manually select
features'
and click 'Enter'



```
C:\> npm  
Vue CLI v4.5.15  
? Please pick a preset:  
  Default ([Vue 2] babel, eslint)  
  Default (Vue 3) ([Vue 3] babel, eslint)  
> Manually select features
```

Select
Choose Vue
version
Babel
Router

Click 'Enter'

C:\> npm

Vue CLI v4.5.15

? Please pick a preset: Manually select features

? Check the features needed for your project:

- ☒ Choose Vue version
- ☒ Babel
- ☐ TypeScript
- ☐ Progressive Web App (PWA) Support
- ☒ Router
- ☐ Vuex
- ☐ CSS Pre-processors
- > ☐ Linter / Formatter
- ☐ Unit Testing
- ☐ E2E Testing

Select 3.x

Click 'Enter'

C:\> npm

Vue CLI v4.5.15

? Please pick a preset: Manually select features

? Check the features needed for your project: Choose Vue version, Babel, Router

? Choose a version of Vue.js that you want to start the project with

2.x

> 3.x

Select 3.x

Click 'Enter'

C:\> npm

Vue CLI v4.5.15

? Please pick a preset: Manually select features

? Check the features needed for your project: Choose Vue version, Babel, Router

? Choose a version of Vue.js that you want to start the project with

2.x

> 3.x

Write 'y'

Click 'Enter'

 npm

Vue CLI v4.5.15

? Please pick a preset: Manually select features

? Check the features needed for your project: Choose Vue version, Babel, Router

? Choose a version of Vue.js that you want to start the project with 3.x

? Use history mode for router? (Requires proper server setup for index fallback in production) (Y/n) y

Write 'y'

Click 'Enter'

 npm

Vue CLI v4.5.15

? Please pick a preset: Manually select features

? Check the features needed for your project: Choose Vue version, Babel, Router

? Choose a version of Vue.js that you want to start the project with 3.x

? Use history mode for router? (Requires proper server setup for index fallback in production) (Y/n) y

Select row
'In dedicated
config files'

Click 'Enter'

npm

Vue CLI v4.5.15

? Please pick a preset: Manually select features

? Check the features needed for your project: Choose Vue version, Babel, Router

? Choose a version of Vue.js that you want to start the project with 3.x

? Use history mode for router? (Requires proper server setup for index fallback in production) Yes

? Where do you prefer placing config for Babel, ESLint, etc.? (Use arrow keys)

> In dedicated config files

In package.json

Write 'n'

Click 'Enter'

```
cat npm
Vue CLI v4.5.15
? Please pick a preset: Manually select features
? Check the features needed for your project: Choose Vue version, Babel, Router
? Choose a version of Vue.js that you want to start the project with 3.x
? Use history mode for router? (Requires proper server setup for index fallback in production) Yes
? Where do you prefer placing config for Babel, ESLint, etc.? In dedicated config files
? Save this as a preset for future projects? (y/N) n_
```


write next commands in cmd:

- `cd my-jobs`
- `code .`

```
File Edit Selection View Go Run Terminal Help

EXPLORER
OPEN EDITORS
  package.json
MY-JOBS
  node_modules
  public
  src
  .browserslistrc
  .gitignore
  babel.config.js
  package-lock.json
  package.json
  README.md

package.json
{
  "name": "my-jobs",
  "version": "0.1.0",
  "private": true,
  "scripts": {
    "serve": "vue-cli-service serve",
    "build": "vue-cli-service build"
  },
  "dependencies": {
    "core-js": "^3.6.5",
    "vue": "^3.0.0",
    "vue-router": "^4.0.0-0"
  },
  "devDependencies": {
    "@vue/cli-plugin-babel": "~4.5.0",
    "@vue/cli-plugin-router": "~4.5.0",
    "@vue/cli-service": "~4.5.0",
    "@vue/compiler-sfc": "^3.0.0"
  }
}
```

Modify file router/index.js

```
import { createRouter, createWebHistory } from 'vue-router'
import Home from '../views/Home.vue'
```

```
import About from '../views/About.vue'
```

Add
code

```
const routes = [
  {
    path: '/',
    name: 'Home',
    component: Home
  },
  {
    path: '/about',
    name: 'About',
```

```
    component: About
```

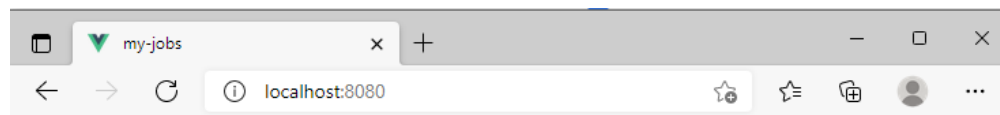
Modify
code

```
  }
]

const router = createRouter({
  history: createWebHistory(process.env.BASE_URL),
  routes
})

export default router
```

write commands in cmd: npm run serve and go to http://localhost:8080/



[Home](#) | [About](#)



Welcome to Your Vue.js App

For a guide and recipes on how to configure / customize this project,
check out the [vue-cli documentation](#).

Installed CLI Plugins

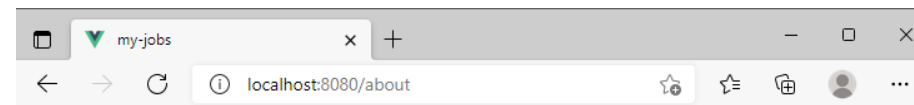
[babel](#) [router](#)

Essential Links

[Core Docs](#) [Forum](#) [Community Chat](#) [Twitter](#) [News](#)

Ecosystem

[vue-router](#) [vuex](#) [vue-devtools](#) [vue-loader](#) [awesome-vue](#)



[Home](#) | [About](#)

This is an about page



Router Links

`<router-link>` is the component for enabling user navigation in a router-enabled app. The target location is specified with the `to` prop. In addition, the link automatically gets an active CSS class when the target route is active.

`<router-link>` is preferred over hard-coded `` for the following reasons:

- It works the same way in both HTML5 history mode and hash mode, so if you ever decide to switch mode, or when the router falls back to hash mode in IE9, nothing needs to be changed.
- In HTML5 history mode, router-link will intercept the click event so that the browser doesn't try to reload the page.
- When you are using the base option in HTML5 history mode, you don't need to include it in to prop's URLs

[More information](#)

Modify file App.vue

```
<template>
...
</template>

<style>
#app {
  font-family: Avenir, Helvetica, Arial, sans-serif;
  -webkit-font-smoothing: antialiased;
  -moz-osx-font-smoothing: grayscale;
  text-align: center;
  color: #2c3e50;
}

#nav {
  padding: 30px;
}

#nav a {
  font-weight: bold;
  color: #2c3e50;

  text-decoration: none;
  padding: 10px;
  border-radius: 4px;
}

#nav a.router-link-exact-active {
  color: white;
  background: #42b983;
}
</style>
```

Add
code

Modify
code

Home | About



Welcome to Your Vue.js App

For a guide and recipes on how to configure / customize this project,
check out the [vue-cli documentation](#).

Installed CLI Plugins

[babel](#) [router](#)

Home | About

This is an about page

Modify file views/Home.vue

```
<template>
  <div class="home">
    <h1>Homepage</h1>
    <p>Welcome to my world!</p>
  </div>
</template>

<script>
export default {
  name: 'Home',
  components: {}
}
</script>
```

Modify file views/About.vue

```
<template>
  <div class="about">
    <h1>This is an about
page</h1>
    <p>My story... </p>
  </div>
</template>
```


Home

About

Homepage

Welcome to my world!

Home

About

This is an about page

My story...



Folder Structure

In the folder **views** create new file Jobs.vue

```
<template>
  <h1>Jobs</h1>
  <div v-for="job in jobs" :key="job.id">
    <h2>{{ job.title }}</h2>
  </div>
</template>

<script>
export default {
  data() {
    return {
      jobs: [
        { title: 'Ninja UX Designer', id: 1, details: 'lorem' },
        { title: 'Ninja Web Developer', id: 2, details: 'lorem' },
        { title: 'Ninja Vue Developer', id: 3, details: 'lorem' },
      ],
    }
  }
}
</script>

<style>
</style>
```

Modify file router/index.js

```
import { createRouter, createWebHistory } from 'vue-router'
import Home from '../views/Home.vue'
import About from '../views/About.vue'
```

```
import Jobs from '../views/Jobs.vue'
```

```
const routes = [
  {
    path: '/',
    name: 'Home',
    component: Home
  },
  {
    path: '/about',
    name: 'About',
    component: About
  },
```

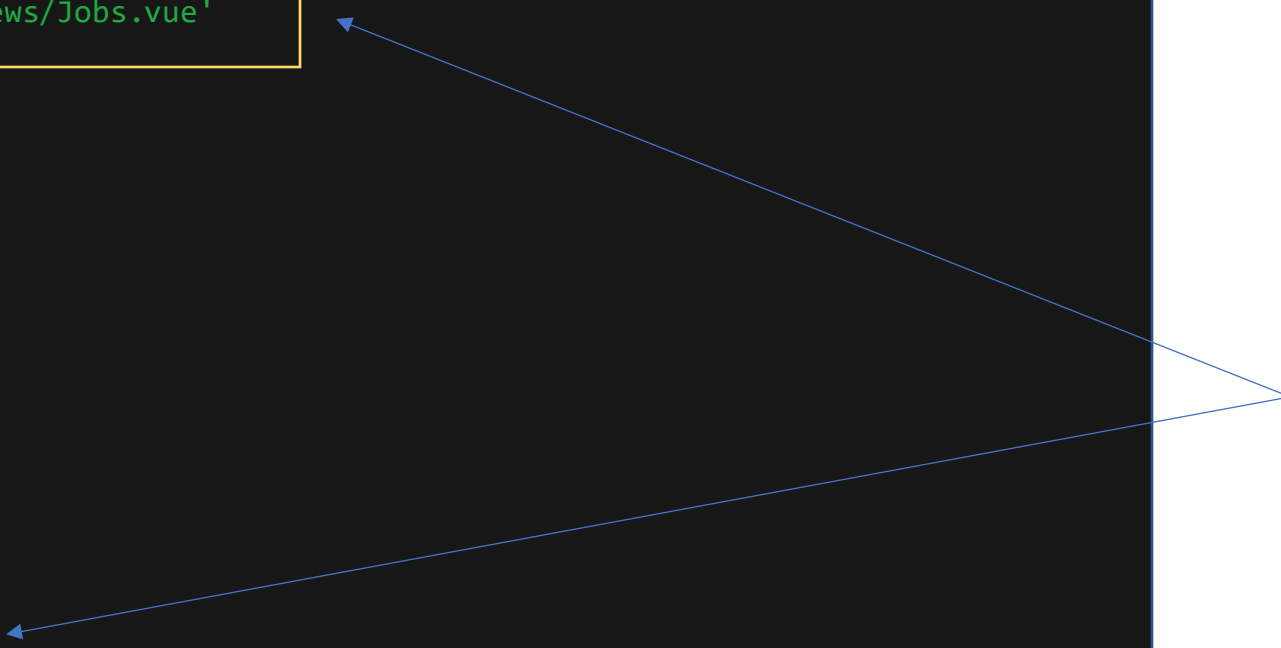
```
{
  path: '/jobs',
  name: 'Jobs',
  component: Jobs
}
```

```
]
```

```
const router = createRouter({
  history: createWebHistory(process.env.BASE_URL),
  routes
})
```

```
export default router
```

Add
code



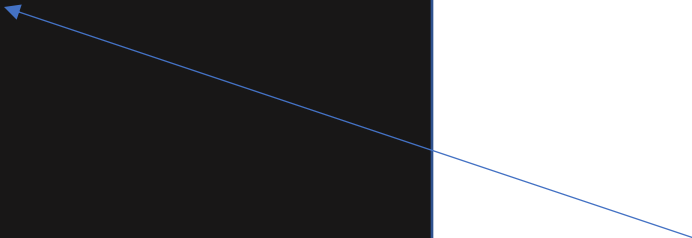
Modify file App.vue

```
<template>
  <div id="nav">
    <router-link to="/">Home</router-link> |
    <router-link to="/about">About</router-link>

    <router-link to="/jobs">Jobs</router-link>
  </div>
  <router-view/>
</template>

<style>
  ...
</style>
```

Add
code



[Home](#) | [About](#)

[Jobs](#)

Jobs

Ninja UX Designer

Ninja Web Developer

Ninja Vue Developer

In the folder **views** create new file **JobDetails.vue**

In the folder **views** create new folder **jobs**

Move files **JobDetails.vue** and **Jobs.vue** to folder **jobs**

Failed to compile.

Home | About **About**

./src/views/Jobs.vue

Module build failed (from ./node_modules/vue-loader-v16/dist/index.js):

Error: ENOENT: no such file or directory, open 'C:\Users\MarizzaMil\Documents\TutsVue\my-jobs\src\views\Jobs.vue'

Ninja UX Designer

Ninja Web Developer

Ninja Vue Developer

Modify file router/index.js

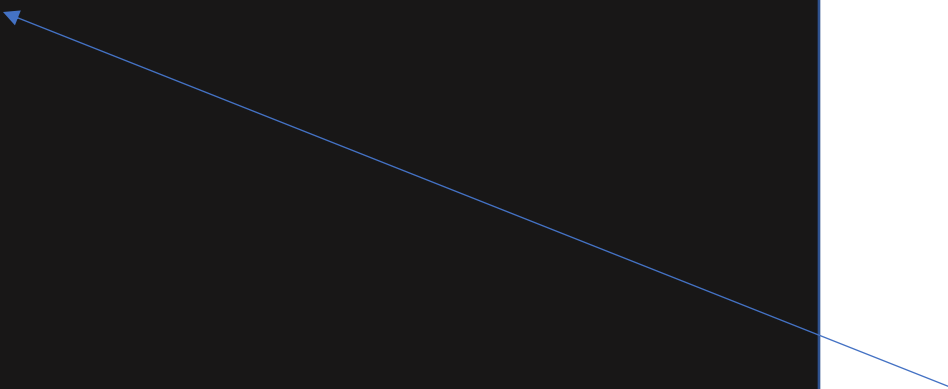
```
import { createRouter, createWebHistory } from 'vue-router'
import Home from '../views/Home.vue'
import About from '../views/About.vue'
import Jobs from '../views/jobs/Jobs.vue'

const routes = [
  {
    path: '/',
    name: 'Home',
    component: Home
  },
  {
    path: '/about',
    name: 'About',
    component: About
  },
  {
    path: '/jobs',
    name: 'Jobs',
    component: Jobs
  }
]

const router = createRouter({
  history: createWebHistory(process.env.BASE_URL),
  routes
})

export default router
```

Modify
code



Route Parameters

[More Information](#)

Route Parameters

/jobs/123



/jobs/567

/jobs/web-dev-job-231

Route Parameters

/jobs/123

/jobs/567

/jobs/web-dev-job-231



Details.vue



Route Parameters

/jobs/123

/jobs/567

/jobs/web-dev-job-231



Details.vue

/jobs/:id

Modify file router/index.js

```
import { createRouter, createWebHistory } from 'vue-router'
import Home from '../views/Home.vue'
import About from '../views/About.vue'
import Jobs from '../views/jobs/Jobs.vue'
```

```
import JobDetails from '../views/jobs/JobDetails.vue'
```

```
const routes = [
  {
    path: '/',
    name: 'Home',
    component: Home
  },
  {
    path: '/about',
    name: 'About',
    component: About
  },
  {
    path: '/jobs',
    name: 'Jobs',
    component: Jobs
  },

```

```
{
  path: '/jobs/:id',
  name: 'JobDetails',
  component: JobDetails
},

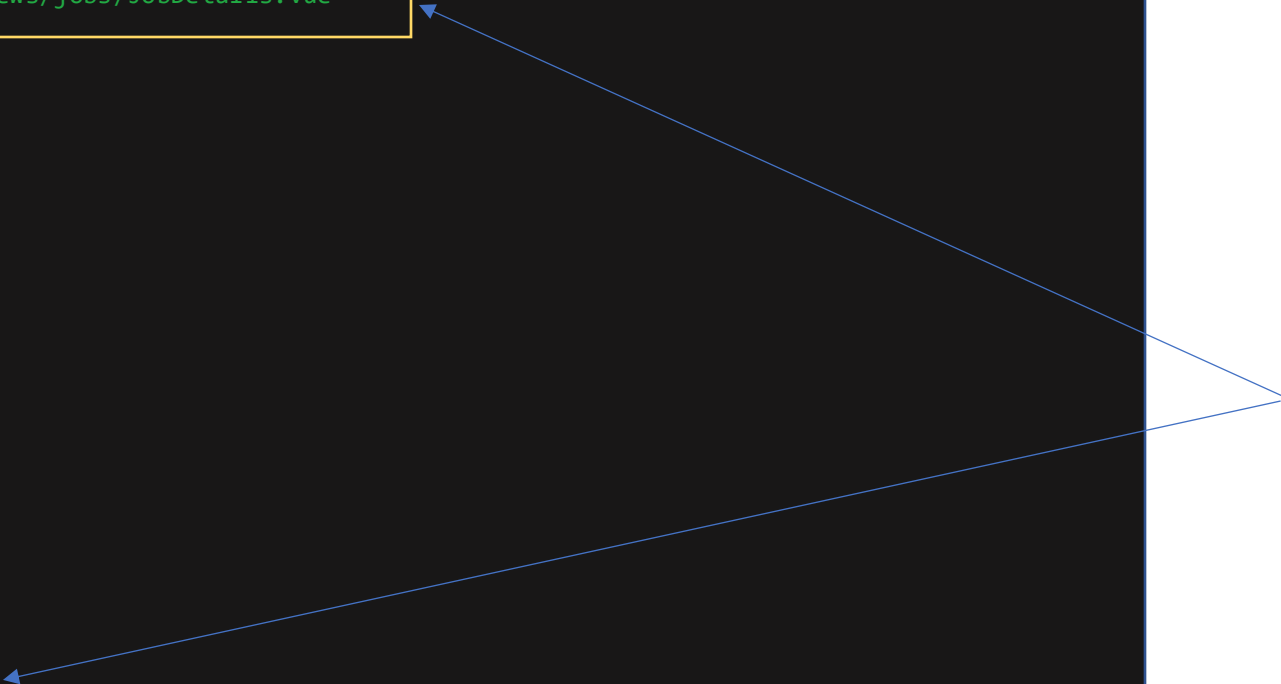
```

```
]
```

```
const router = createRouter({
  history: createWebHistory(process.env.BASE_URL),
  routes
})
```

```
export default router
```

Modify
code



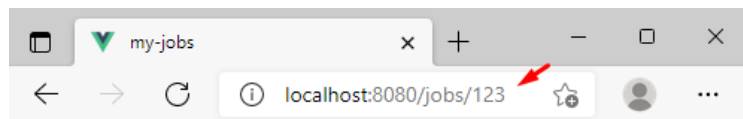
Modify file views/jobs/jobDetails.vue

```
<template>
  <h1>Job Details</h1>
  <p>The job id is {{ $route.params.id }}</p>
</template>

<script>
export default {

}
</script>

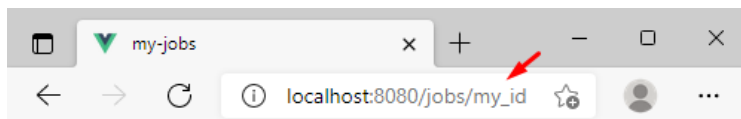
<style>
</style>
```



[Home](#) | [About](#) [Jobs](#)

Job Details

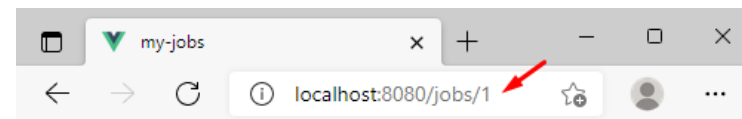
The job id is 123



[Home](#) | [About](#) [Jobs](#) :

Job Details

The job id is my_id



[Home](#) | [About](#) [Jobs](#)

Job Details

The job id is 1

Modify file views/jobs/jobDetails.vue

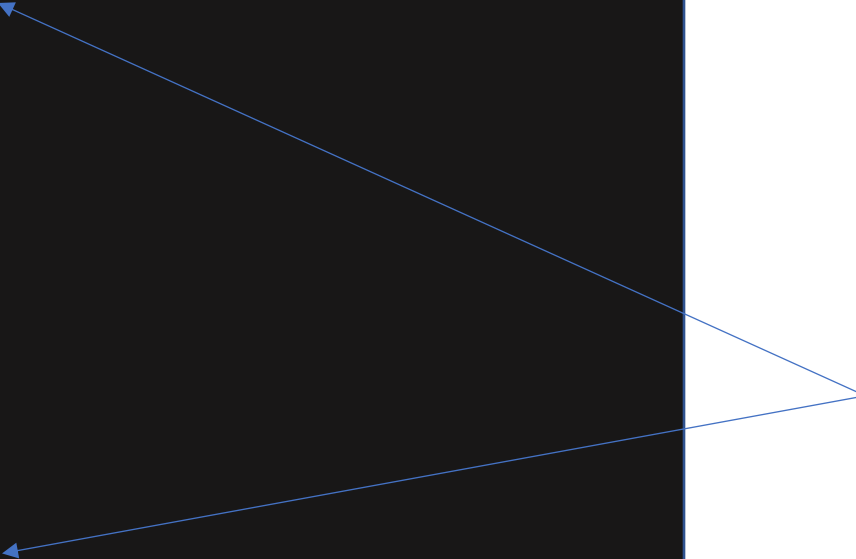
```
<template>
  <h1>Job Details</h1>
  <!-- <p>The job id is {{ $route.params.id }}</p> -->

  <p>The job id is {{ id }}</p>
</template>

<script>
export default {
  data() {
    return {
      id: this.$route.params.id
    }
  }
}
</script>

<style>
</style>
```

Add
code



Dynamic Links

[More Information](#)

Modify file views/jobs/jobs.vue

```
<template>
  <h1>Jobs</h1>
  <div v-for="job in jobs" :key="job.id">

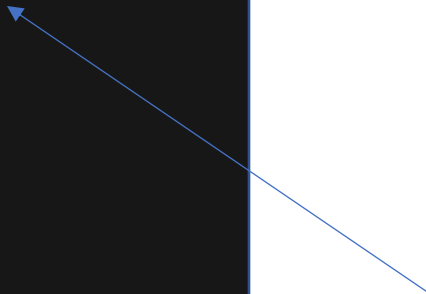
    <router-link :to="{name: 'JobDetails', params: { id: job.id }}">
      <h2>{{ job.title }}</h2>
    </router-link>

  </div>
</template>

<script>
export default {
  data() {
    return {
      jobs: [
        { title: 'Ninja UX Designer', id: 1, details: 'lorem' },
        { title: 'Ninja Web Developer', id: 2, details: 'lorem' },
        { title: 'Ninja Vue Developer', id: 3, details: 'lorem' },
      ],
    }
  }
}
</script>

<style>
</style>
```

Add
code



Home | About

Jobs

Jobs

[Ninja UX Designer](#)

[Ninja Web Developer](#)

[Ninja Vue Developer](#)

```
Elements Console Sources Network Performance Memory
<!DOCTYPE html>
<html lang>
  <head>...</head>
  ... <body> == $0
    <noscript>...</noscript>
    <div id="app" data-v-app>
      <div id="nav">...</div>
      <h1>Jobs</h1>
      <div>
        <a href="/jobs/1" class>...</a>
      </div>
      <div>
        <a href="/jobs/2" class>...</a>
      </div>
      <div>
        <a href="/jobs/3" class>...</a>
      </div>
    </div>
    <!-- built files will be auto injected -->
    <script type="text/javascript" src="/js/chunk-vendors.js"></script>
    <script type="text/javascript" src="/js/app.js"></script>
  </body>
</html>
```

Modify file views/jobs/jobDetails.vue

```
<template>
  <h1>Job Details</h1>
  <!-- <p>The job id is {{ $route.params.id }}</p> -->
  <p>The job id is {{ id }}</p>
</template>
```

```
<script>
export default {
```

```
  props: ['id'],
```

```
  // data() {
  //   return {
  //     id: this.$route.params.id
  //   }
  // }
```

```
}
</script>
```

```
<style>
</style>
```

Add
code

Comment
code

Modify file router/index.js

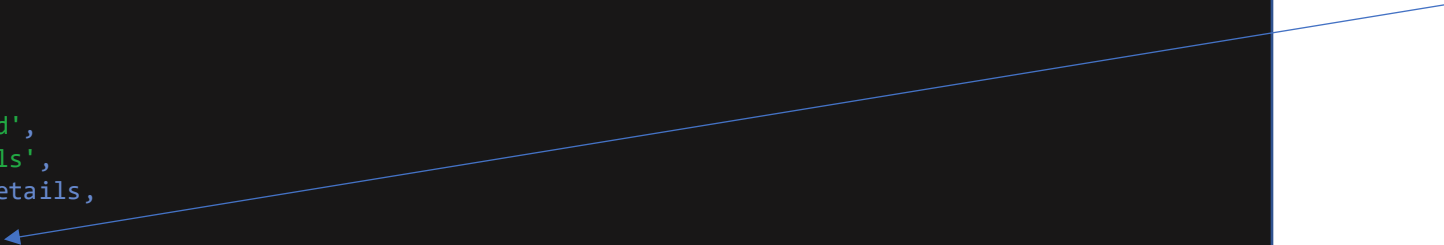
```
import { createRouter, createWebHistory } from 'vue-router'
import Home from '../views/Home.vue'
import About from '../views/About.vue'
import Jobs from '../views/jobs/Jobs.vue'
import JobDetails from '../views/jobs/JobDetails.vue'

const routes = [
  {
    path: '/',
    name: 'Home',
    component: Home
  },
  {
    path: '/about',
    name: 'About',
    component: About
  },
  {
    path: '/jobs',
    name: 'Jobs',
    component: Jobs
  },
  {
    path: '/jobs/:id',
    name: 'JobDetails',
    component: JobDetails,
    props: true
  },
]

const router = createRouter({
  history: createWebHistory(process.env.BASE_URL),
  routes
})

export default router
```

Add
code



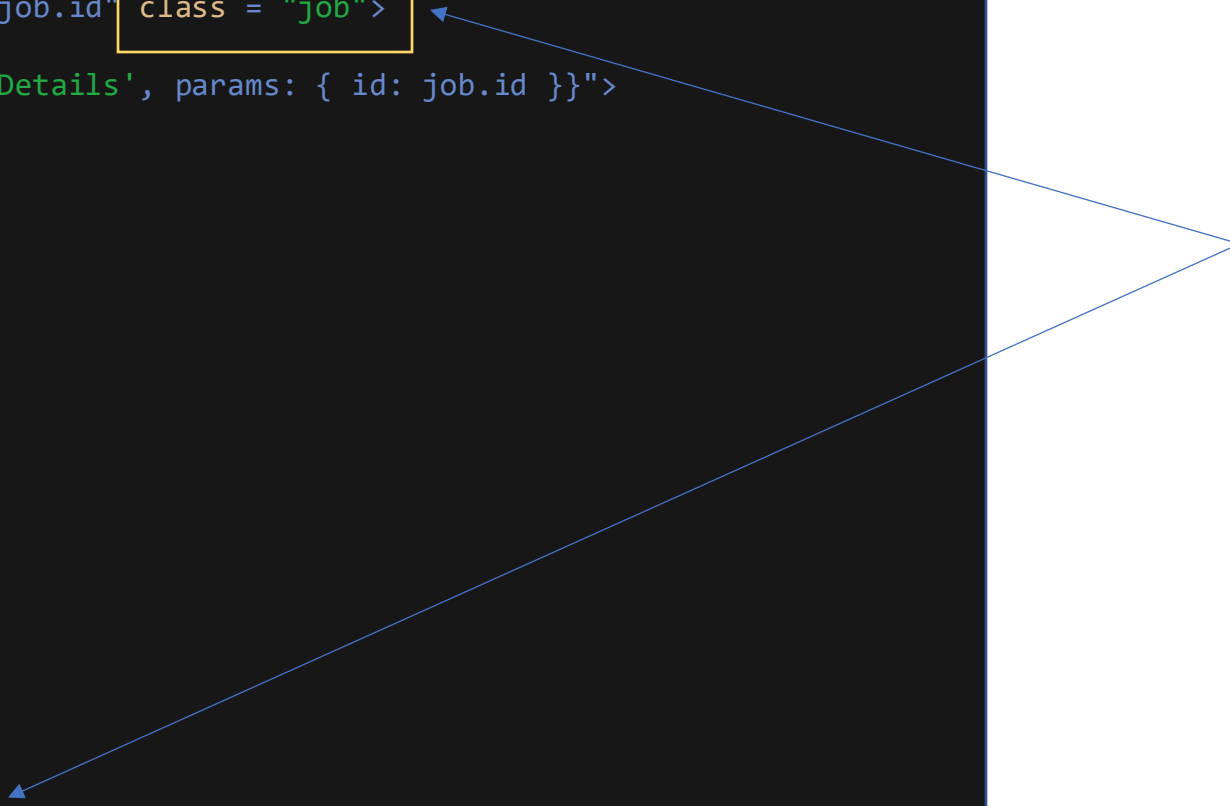
Modify file views/jobs/jobs.vue

```
<template>
  <h1>Jobs</h1>
  <div v-for="job in jobs" :key="job.id" class = "job">
    <router-link :to="{name: 'JobDetails', params: { id: job.id }}">
      <h2>{{ job.title }}</h2>
    </router-link>
  </div>
</template>
```

```
<script>
...
</script>
```

```
<style>
.job h2{
  background: #f4f4f4;
  padding: 20px;
  border-radius: 10px;
  margin: 10px auto;
  max-width: 600px;
  cursor: pointer;
  color: #444;
}
.job h2:hover {
  background: #ddd;
}
.job a{
  text-decoration: none;
}
</style>
```

Add
code



[Home](#) | [About](#)

[Jobs](#)

Jobs

Ninja UX Designer

Ninja Web Developer

Ninja Vue Developer



Welcome

Elements

Console

Sources

Network

Performance



```
<!DOCTYPE html>
<html lang>
  <head>...</head>
  <body> == $0
    <noscript>...</noscript>
    <div id="app" data-v-app>
      <div id="nav">...</div>
      <h1>Jobs</h1>
      <div class="job">
        <a href="/jobs/1" class>...</a>
      </div>
      <div class="job">
        <a href="/jobs/2" class>...</a>
      </div>
      <div class="job">
        <a href="/jobs/3" class>...</a>
      </div>
    </div>
    <!-- built files will be auto injected -->
    <script type="text/javascript" src="/js/chunk-vendors.js"></script>
    <script type="text/javascript" src="/js/app.js"></script>
  </body>
</html>
```

Redirects & 404's

[More Information](#)

Modify file router/index.js

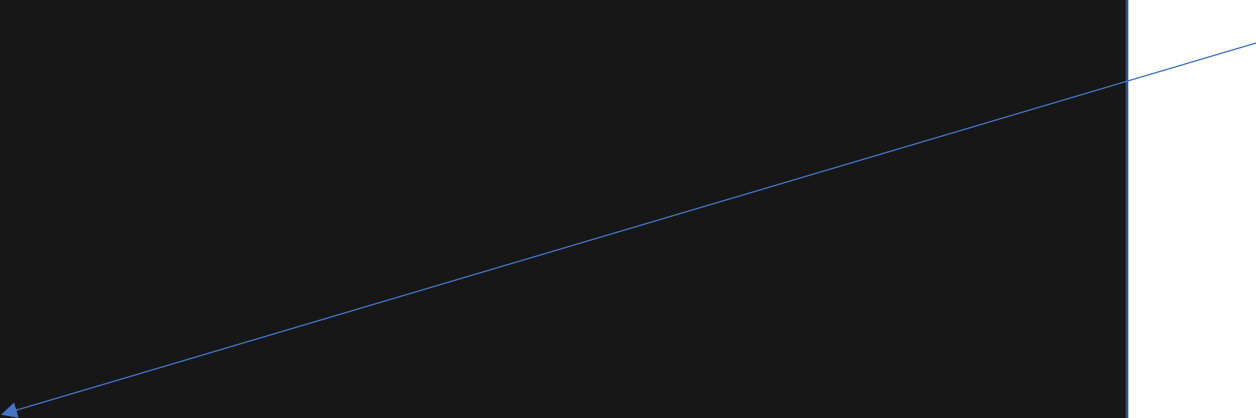
```
import { createRouter, createWebHistory } from 'vue-router'
import Home from '../views/Home.vue'
import About from '../views/About.vue'
import Jobs from '../views/jobs/Jobs.vue'
import JobDetails from '../views/jobs/JobDetails.vue'

const routes = [
  {
    path: '/',
    name: 'Home',
    component: Home
  },
  {
    path: '/about',
    name: 'About',
    component: About
  },
  {
    path: '/jobs',
    name: 'Jobs',
    component: Jobs
  },
  {
    path: '/jobs/:id',
    name: 'JobDetails',
    component: JobDetails,
    props: true
  },
  // redirect
  {
    path: '/all-jobs',
    redirect: '/jobs'
  }
]

const router = createRouter({
  history: createWebHistory(process.env.BASE_URL),
  routes
})

export default router
```

Add
code



Click Enter



Home | About **Jobs**

Jobs

Ninja UX Designer

Ninja Web Developer

Ninja Vue Developer



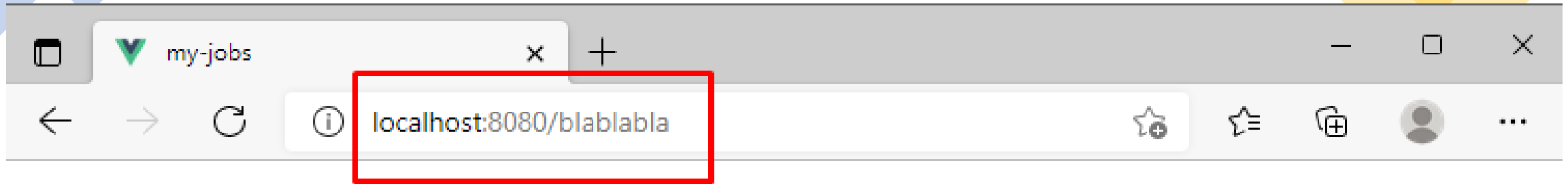
Home | About **Jobs**

Jobs

Ninja UX Designer

Ninja Web Developer

Ninja Vue Developer



[Home](#) | [About](#) [Jobs](#)

Create file views/NotFound.vue

```
<template>
  <h2>404</h2>
  <h3>That page does not exist</h3>
</template>

<script>
export default {
}
</script>

<style>
</style>
```

Modify file router/index.js

```
import { createRouter, createWebHistory } from 'vue-router'
import Home from '../views/Home.vue'
import About from '../views/About.vue'
import Jobs from '../views/jobs/Jobs.vue'
import JobDetails from '../views/jobs/JobDetails.vue'
```

```
import NotFound from '../views/NotFound.vue'
```


```
const routes = [
  ...
  // redirect
  {
    path: '/all-jobs',
    redirect: '/jobs'
  },
```

```
  // 404 catchall
  {
    path: '/*:catchAll(.*)',
    name: 'NotFound',
    component: NotFound
  }
]
```

```
const router = createRouter({
  history: createWebHistory(process.env.BASE_URL),
  routes
})
```

```
export default router
```

Add
code





[Home](#) | [About](#) [Jobs](#)

404

That page does not exist

Programmatic Navigation

[More Information](#)

Modify file App.vue

```
<template>
  <div id="nav">
    <router-link to="/">Home</router-link> |
    <router-link to="/about">About</router-link>
    <router-link to="/jobs">Jobs</router-link>
  </div>

  <button @click="redirect">Redirect</button>
  <button @click="back">Go back</button>
  <button @click="forward">Go forward</button>

  <router-view/>
</template>
```

```
<script>
export default {
  methods: {
    redirect() {
      this.$router.push({ name: 'Home' })
    },
    back() {
      this.$router.go(-1)
    },
    forward() {
      this.$router.go(1)
    }
  }
}
</script>
```

```
<style>
...
```

```
button {
  margin: 0 10px;
  padding: 10px;
  border: none;
  border-radius: 4px;
}
</style>
```

Click to add text

Add
code



Home | **About** **Jobs**

Redirect

Go back

Go forward

Homepage

Welcome to my world!

Using JSON Server

JSON Server

db.json

```
{  
  "blogs": [  
    { "title": "new site design", id: 1 },  
    { "title": "dev job stats", id: 2 },  
    { "title": "tech party 2021", id: 3 }  
  ]  
}
```

JSON Server

db.json

```
{
  "blogs": [
    { "title": "new site design", id: 1 },
    { "title": "dev job stats", id: 2 },
    { "title": "tech party 2021", id: 3 }
  ]
}
```

JSON Server

localhost:3000/blogs (GET)

localhost:3000/blogs (POST)

localhost:3000/blogs/id (GET)

Create folder **data**

In the **folder** data create file **db.json**

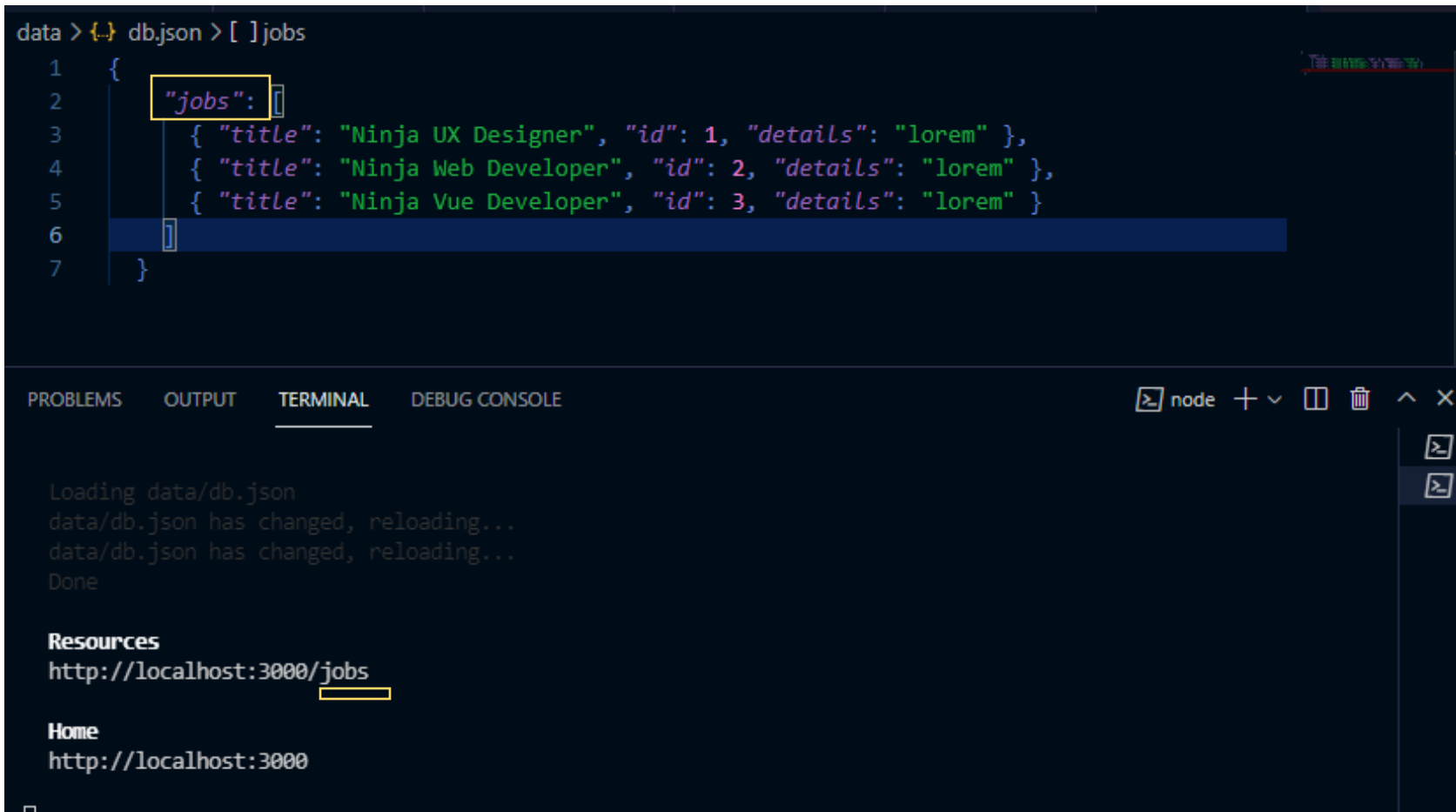
```
{
  "jobs": [
    { "title": "Ninja UX Designer", "id": 1, "details": "lorem" },
    { "title": "Ninja Web Developer", "id": 2, "details": "lorem" },
    { "title": "Ninja Vue Developer", "id": 3, "details": "lorem" }
  ]
}
```

Open new Terminal

In the Terminal write next command:

npm install json-server --save

npx json-server --watch data/db.json



The screenshot shows a VS Code editor with a file named `data/db.json` open. The file contains a JSON object with a `jobs` array. The `jobs` array has three elements, each representing a job with a title, id, and details. The `jobs` property is highlighted with a yellow box. Below the editor, the terminal window is open, showing the output of the `npx json-server` command. The terminal output indicates that the data is being loaded and reloaded as the file changes. The `Resources` section shows the URL `http://localhost:3000/jobs`, which is also highlighted with a yellow box. The `Home` section shows the URL `http://localhost:3000`.

```
data > { db.json > [ ] jobs
1  {
2    "jobs": [
3      { "title": "Ninja UX Designer", "id": 1, "details": "lorem" },
4      { "title": "Ninja Web Developer", "id": 2, "details": "lorem" },
5      { "title": "Ninja Vue Developer", "id": 3, "details": "lorem" }
6    ]
7  }
```

PROBLEMS OUTPUT **TERMINAL** DEBUG CONSOLE

node + - [] [] ^ X

Loading data/db.json
data/db.json has changed, reloading...
data/db.json has changed, reloading...
Done

Resources
<http://localhost:3000/jobs>

Home
<http://localhost:3000>

Modify *data/db.json*

```
data > {} db.json > ...
1  {}
2  "jobs": [
3    { "title": "Ninja UX Designer", "id": 1, "details": "lorem" },
4    { "title": "Ninja Web Developer", "id": 2, "details": "lorem" },
5    { "title": "Ninja Vue Developer", "id": 3, "details": "lorem" }
6  ],
7  "blogs": []
8  }
```

PROBLEMS OUTPUT **TERMINAL** DEBUG CONSOLE

node + v [] [] ^ x

Loading data/db.json
data/db.json has changed, reloading...
Done

Resources
<http://localhost:3000/jobs>
<http://localhost:3000/blogs>

Home
<http://localhost:3000>



```
[
  {
    "title": "Ninja UX Designer",
    "id": 1,
    "details": "lorem"
  },
  {
    "title": "Ninja Web Developer",
    "id": 2,
    "details": "lorem"
  },
  {
    "title": "Ninja Vue Developer",
    "id": 3,
    "details": "lorem"
  }
]
```


Fetching Data

[More Information](#)

Modify file views/jobs/jobs.vue

```
<template>
  <h1>Jobs</h1>
  <div v-for="job in jobs" :key="job.id" class = "job">

    <router-link :to="{name: 'JobDetails', params: { id: job.id }}">
      <h2>{{ job.title }}</h2>
    </router-link>

  </div>
</template>

<script>
export default {

  data() {
    return {
      jobs: [],
    },
  },

  mounted() {
    fetch('http://localhost:3000/jobs')
      .then(res => res.json())
      .then(data => this.jobs = data)
      .catch(err => console.log(err.message))
  }

}
</script>

<style>...</style>
```

Modify
code

Add
code



[Home](#) | [About](#) [Jobs](#)

Redirect

Go back

Go forward

Jobs

Ninja UX Designer

Ninja Web Developer

Ninja Vue Developer

Modify file views/jobs/jobDetails.vue

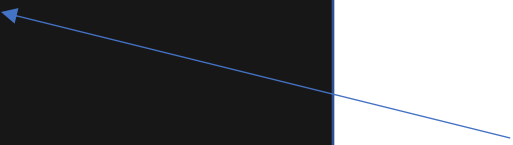
```
<template>
  <h1>Job Details</h1>
  <p>The job id is {{ id }}</p>
</template>

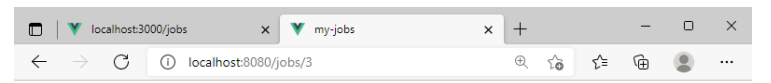
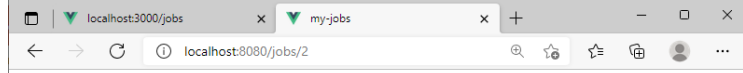
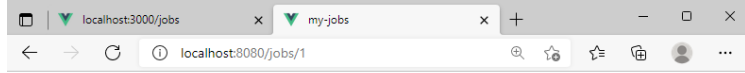
<script>
export default {
  props: ['id'],

  data() {
    return {
      job: null
    }
  },
  mounted() {
    fetch('http://localhost:3000/jobs/' + this.id)
      .then(res => res.json())
      .then(data => this.job = data)
      .catch(err => console.log(err.message))
  }
}
</script>

<style>
</style>
```

Add
code





Modify file views/jobs/jobDetails.vue

```
<template>
  <h1>{{ job.title }}</h1> <!-- gets an error -->
  <p>The job id is {{ id }}</p>
</template>

<script>
export default {
  props: ['id'],
  data() {
    return {
      job: null
    }
  },
  mounted() {
    fetch('http://localhost:3000/jobs/' + this.id)
      .then(res => res.json())
      .then(data => this.job = data)
      .catch(err => console.log(err.message))
  }
}
</script>

<style>
</style>
```

Add
code



Conditionally Showing Data

Modify file views/jobs/jobDetails.vue

```
<template>

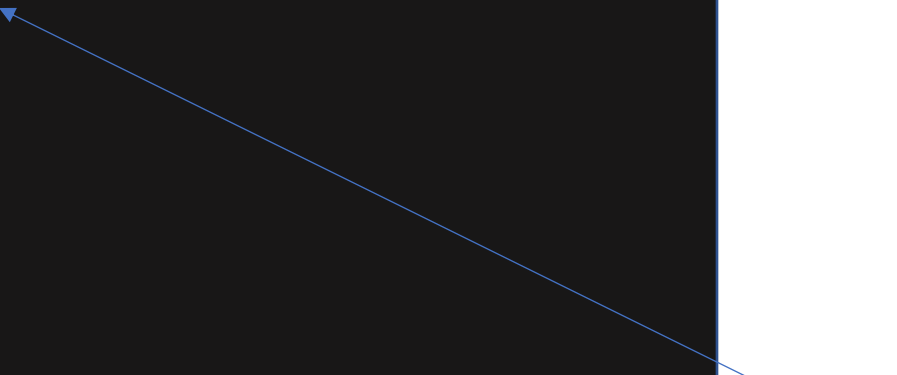
  <div v-if="job">
    <h1>{{ job.title }}</h1>
    <p>{{ job.details }}</p>
  </div>

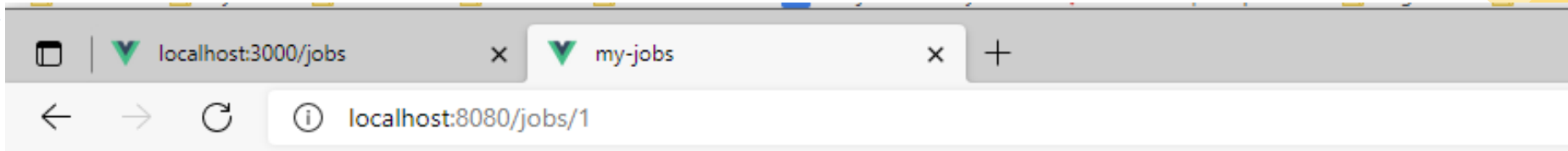
</template>

<script>
export default {
  props: ['id'],
  data() {
    return {
      job: null
    }
  },
  mounted() {
    fetch('http://localhost:3000/jobs/' + this.id)
      .then(res => res.json())
      .then(data => this.job = data)
      .catch(err => console.log(err.message))
  }
}
</script>

<style>
</style>
```

Modify
code





[Home](#) | [About](#) [Jobs](#)

Redirect

Go back

Go forward

Ninja UX Designer

lorem

Modify file views/jobs/jobDetails.vue

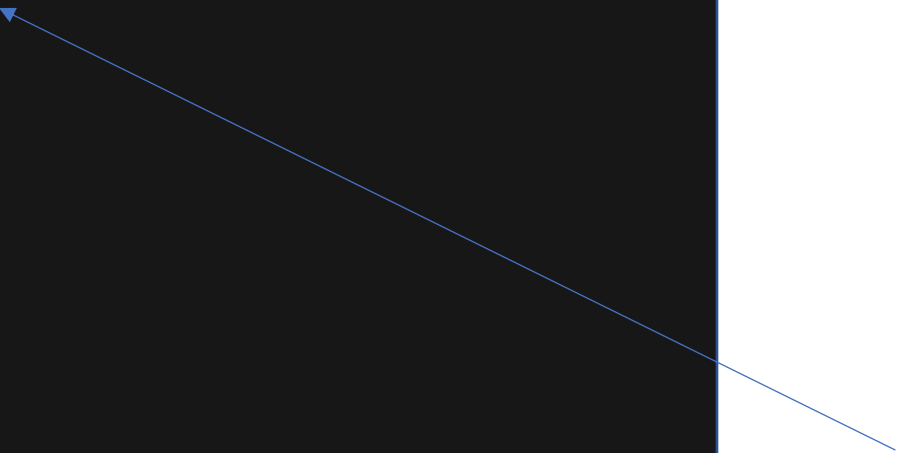
```
<template>
  <div v-if="job">
    <h1>{{ job.title }}</h1>
    <p>{{ job.details }}</p>
  </div>

  <div v-else>
    <p>Loading job details...</p>
  </div>
</template>

<script>
export default {
  props: ['id'],
  data() {
    return {
      job: null
    }
  },
  mounted() {
    fetch('http://localhost:3000/jobs/' + this.id)
      .then(res => res.json())
      .then(data => this.job = data)
      .catch(err => console.log(err.message))
  }
}
</script>

<style>
</style>
```

Add
code



Modify file views/jobs/jobs.vue

```
<template>
  <h1>Jobs</h1>

  <div v-if="jobs.length">
    <div v-for="job in jobs" :key="job.id" class="job">
      <router-link :to="{name: 'JobDetails', params: { id: job.id }}">
        <h2>{{ job.title }}</h2>
      </router-link>
    </div>
  </div>
  <div v-else>
    <p>Loading jobs...</p>
  </div>
</template>
```

```
</template>

<script>
export default {
  data() {
    return {
      jobs: [],
    }
  },
  mounted() {
    fetch('http://localhost:3000/jobs')
      .then(res => res.json())
      .then(data => this.jobs = data)
      .catch(err => console.log(err.message))
  }
}
</script>
<style>...</style>
```

Modify
code

