V3.x

"vue-router": "^3.0.1",

<https://v3.router.vuejs.org/installation.html>

v4.x

<script src="https://unpkg.com/vue@3"></script>

<script src="https://unpkg.com/vue-router@4"></script>

<div id="app">

<h1>Hello App!</h1>

<p>

<!-- use the router-link component for navigation. -->

<!-- specify the link by passing the `to` prop. -->

<!-- `<router-link>` will render an `<a>` tag with the correct `href` attribute -->

<router-link to="/">Go to Home</router-link>

<router-link to="/about">Go to About</router-link>

</p>

<!-- route outlet -->

<!-- component matched by the route will render here -->

<router-view></router-view></div>

router-link**[#](https://router.vuejs.org/guide/" \l "router-link)**

Note how instead of using regular a tags, we use a custom component router-link to create links. This allows Vue Router to change the URL without reloading the page, handle URL generation as well as its encoding. We will see later how to benefit from these features.

router-view**[#](https://router.vuejs.org/guide/" \l "router-view)**

router-view will display the component that corresponds to the url. You can put it anywhere to adapt it to your layout.

<https://router.vuejs.org/>

<https://router.vuejs.org/installation.html>

**Installation**

## **Direct Download / CDN[#](https://router.vuejs.org/installation.html" \l "direct-download-cdn)**

[https://unpkg.com/vue-router@4](https://unpkg.com/vue-router@4" \t "https://router.vuejs.org/_blank)

[Unpkg.com](https://unpkg.com/" \t "https://router.vuejs.org/_blank) provides npm-based CDN links. The above link will always point to the latest release(版本) on npm. You can also use a specific version/tag via URLs like https://unpkg.com/vue-router@4.0.5/dist/vue-router.global.js.

## **npm**

**Essentials**

Getting Started

Creating a Single-page Application with Vue + Vue Router feels natural: with Vue.js, we are already composing our application with components. When adding Vue Router to the mix, all we need to do is map our components to the routes and let Vue Router know where to render them. Here's a basic example:

<script src="https://unpkg.com/vue@3"></script>

<script src="https://unpkg.com/vue-router@4"></script>

<div id="app">

<h1>Hello App!</h1>

<p>

<!-- use the router-link component for navigation. -->

<!-- specify the link by passing the `to` prop. -->

<!-- `<router-link>` will render an `<a>` tag with the correct `href` attribute -->

<router-link to="/">Go to Home</router-link>

<router-link to="/about">Go to About</router-link>

</p>

<!-- route outlet -->

<!-- component matched by the route will render here -->

<router-view></router-view></div>