

Introduction

In Vue.js, there are two different types of router history modes available: `createWebHistory` and `createWebHashHistory`. This report will outline the differences between these two history modes.

Web History Mode

`createWebHistory` uses the HTML5 History API to modify the URL without requiring a page reload. This means that URLs look cleaner and more natural (e.g., `/about` instead of `/#/about`). However, this mode requires server-side configuration for proper handling of fallback routes.

Here's an example usage:

```
1 import { createRouter, createWebHistory } from 'vue-router'
2
3 const router = createRouter({
4   history: createWebHistory(),
5   // Other options go here...
6 })
```

Web Hash History Mode

On the other hand, `createWebHashHistory` uses the hash portion (`#`) of the URL to simulate client-side routing without actually changing the page's location. This mode works out-of-the-box with any server configuration since all requests will be sent to your root path regardless of what comes after the hash sign.

Here's an example usage:

```
1 import { createRouter, createWebHashHistory } from 'vue-router'
2
3 const router = createRouter({
4   history: createWebHashHistory(),
5   // Other options go here...
6 })
```

It's worth noting that some older browsers may not support HTML5 History API which means using Web Hash History might be a better choice in such cases as it provides consistent behavior across all modern and legacy browsers.

Conclusion

Both `createWebHistory()` and `createWebHashHistory()` are useful depending on specific use cases but one should choose wisely based on the requirements of their application.