



SVELTEKIT • REFERENCE

# \$app/stores

ON THIS PAGE



```
import { getStores, navigating, page, updated } from '$app/stores';
```



## getStores

```
function getStores(): {  
  page: typeof page;  
  
  navigating: typeof navigating;  
  
  updated: typeof updated;  
};
```

## navigating

A readable store. When navigating starts, its value is a `Navigation` object with `from`, `to`, `type` and (if `type === 'popstate'`) `delta` properties. When navigating finishes, its value reverts to `null`.

On the server, this store can only be subscribed to during component initialization. In the browser, it can be subscribed to at any time.

```
const navigating: import('svelte/store').Readable<  
  import('@sveltejs/kit').Navigation | null  
>;
```



A readable store whose value contains page data.

On the server, this store can only be subscribed to during component initialization. In the browser, it can be subscribed to at any time.

```
const page: import('svelte/store').Readable<
  import('@sveltejs/kit').Page
>;
```

## updated

A readable store whose initial value is `false`. If `version.pollInterval` is a non-zero value, SvelteKit will poll for new versions of the app and update the store value to `true` when it detects one. `updated.check()` will force an immediate check, regardless of polling.

On the server, this store can only be subscribed to during component initialization. In the browser, it can be subscribed to at any time.

```
const updated: import('svelte/store').Readable<boolean> & {
  check(): Promise<boolean>;
};
```

[✎ Edit this page on GitHub](#)

PREVIOUS

[\\$app/server](#)

NEXT

[\\$env/dynamic/private](#)