

Unlock

August 18, 2026

Unlock

Kind: Constant

Source: `packages/common/decorators/http/request-mapping.decorator.ts`

Part of: [Common](#)

Route handler (method) Decorator. Routes Webdav UNLOCK requests to the specified path.

`Unlock` is a route handler decorator for WebDAV `UNLOCK` requests. Apply it to a controller method to map an `UNLOCK` request for a specific path to that handler, allowing the application to release an existing WebDAV lock.

Definition

```
createMappingDecorator(RequestMethod.UNLOCK)
```

Value

```
createMappingDecorator(RequestMethod.UNLOCK)
```

Diagram

```
graph LR
  Client[WebDAV Client] -->|UNLOCK /resource| Router[HTTP Request Router]
  Router -->|Matches path and UNLOCK method| Decorator["@Unlock('/resource')"]
  Decorator --> Handler[Controller Method]
  Handler --> Response[HTTP Response]
```

Usage

```
import { Unlock } from '@your-package/common/decorators/http/request-mapping.decorator';

class WebDavController {
  @Unlock('/documents/:id')
  async unlockDocument() {
    // Release the lock for the requested document.
    return {
      statusCode: 204,
    };
  }
}
```

AI Coding Instructions

- Use `@Unlock()` only on controller methods responsible for handling WebDAV `UNLOCK` requests.
- Provide a path that matches the resource route used by the corresponding WebDAV lock handling logic.
- Keep lock validation and token handling inside the decorated handler or delegated service layer.
- Do not use `Unlock` for generic HTTP `DELETE` or update operations; it is specifically for the WebDAV `UNLOCK` method.
- Ensure the application's HTTP router and WebDAV middleware support the `UNLOCK` HTTP verb.