

ServeStaticOptions

August 17, 2026

ServeStaticOptions

Kind: Interface

Source: `packages/platform-express/interfaces/serve-static-options.interface.ts`

Part of: Platform Express

Interface describing options for serving static assets.

`ServeStaticOptions` configures how static assets are served by the Express platform integration. It controls cache behavior, index-file resolution, redirects, dotfile access, extension handling, and custom response headers when using static file middleware.

Properties

Property	Type
<code>dotfiles</code>	<code>string</code>
<code>etag</code>	<code>boolean</code>
<code>extensions</code>	<code>string[]</code>
<code>fallthrough</code>	<code>boolean</code>
<code>immutable</code>	<code>boolean</code>
<code>index</code>	<code>`boolean</code>
<code>lastModified</code>	<code>boolean</code>
<code>maxAge</code>	<code>`number</code>
<code>redirect</code>	<code>boolean</code>
<code>setHeaders</code>	<code>(res: any, path: string, stat: any) => any</code>
<code>prefix</code>	<code>string</code>

Diagram

```
graph LR
  A[Static asset request] --> B[ServeStaticOptions]
  B --> C[File resolution]
  B --> D[Cache headers]
  B --> E[Redirect and fallback behavior]
  B --> F[Custom setHeaders callback]
  C --> G[HTTP response]
  D --> G
  E --> G
  F --> G
```

Usage

```
import { NestFactory } from '@nestjs/core';
import { NestExpressApplication } from '@nestjs/platform-express';
import { join } from 'path';
import { AppModule } from './app.module';

async function bootstrap() {
  const app = await NestFactory.create<NestExpressApplication>(AppModule);

  app.useStaticAssets(join(__dirname, '..', 'public'), {
    maxAge: '1d',
    etag: true,
    lastModified: true,
    index: ['index.html'],
    extensions: ['html'],
    fallback: true,
    redirect: true,
    immutable: false,
    dotfiles: 'ignore',
    setHeaders(res, path, stat) {
      if (path.endsWith('.css') || path.endsWith('.js')) {
        res.setHeader('Cache-Control', 'public, max-age=31536000, immutable');
      }
    },
  });

  await app.listen(3000);
}

bootstrap();
```

AI Coding Instructions

- Pass `ServeStaticOptions` to `NestExpressApplication.useStaticAssets()` when configuring Express-hosted static directories.

- Use `maxAge`, `etag`, `lastModified`, and `immutable` together to define a consistent browser and CDN caching strategy.
- Keep `dotfiles` restrictive unless hidden files must be publicly accessible; prefer `'ignore'` or `'deny'` over serving them.
- Enable `fallthrough` when unresolved static requests should continue to Nest controllers or other middleware.
- Use `setHeaders` for asset-specific headers, but avoid overriding cache headers inconsistently across related file types.