

StreamableFileOptions

August 18, 2026

StreamableFileOptions

Kind: Interface

Source: `packages/common/file-stream/interfaces/streamable-options.interface.ts`

Part of: [Common](#)

Options for `StreamableFile`

`StreamableFileOptions` configures HTTP response metadata for a `StreamableFile`. It defines the file MIME type, optional content disposition header value(s), and content length so clients can correctly handle streamed file responses.

Properties

Property	Type
<code>type</code>	<code>string</code>
<code>disposition</code>	<code>`string</code>
<code>length</code>	<code>number</code>

Diagram

```
graph LR
  A[File stream or buffer] --> B[StreamableFile]
  C[StreamableFileOptions] --> B
  C --> D[type: Content-Type]
  C --> E[disposition: Content-Disposition]
  C --> F[length: Content-Length]
  B --> G[HTTP response]
```

Usage

```
import { Controller, Get, StreamableFile } from '@nestjs/common';
import { createReadStream } from 'node:fs';
import { join } from 'node:path';
import type { StreamableFileOptions } from '@nestjs/common';

@Controller('reports')
export class ReportsController {
  @Get('download')
  download(): StreamableFile {
    const filePath = join(process.cwd(), 'reports', 'monthly-report.pdf');

    const options: StreamableFileOptions = {
      type: 'application/pdf',
      disposition: 'attachment; filename="monthly-report.pdf"',
      length: 1024 * 256,
    };

    return new StreamableFile(createReadStream(filePath), options);
  }
}
```

AI Coding Instructions

- Set `type` to a valid MIME type so browsers and API clients interpret the streamed content correctly.
- Use `disposition` with `attachment` to force downloads, or `inline` when the file should render in the browser.
- Provide `length` when known to enable an accurate `Content-Length` response header.
- Ensure the declared `length` matches the actual stream or buffer size; incorrect values can cause incomplete downloads or hanging clients.
- Pass these options directly to `StreamableFile` when returning files from NestJS controllers.