

ParseBoolPipeOptions

August 18, 2026

ParseBoolPipeOptions

Kind: Interface

Source: `packages/common/pipes/parse-bool.pipe.ts`

Part of: [Common](#)

`ParseBoolPipeOptions` configures the behavior of NestJS's `ParseBoolPipe`, which converts incoming request values into boolean values. It lets applications customize the HTTP status code, exception creation, and whether missing values should be accepted without parsing.

Properties

Property	Type
<code>errorHttpStatusCode</code>	<code>ErrorHttpStatusCode</code>
<code>exceptionFactory</code>	<code>(error: string) => any</code>
<code>optional</code>	<code>boolean</code>

Diagram

```
graph LR
  A[Incoming request value] --> B[ParseBoolPipe]
  B --> C{Value is boolean-compatible?}
  C -->|Yes| D[Return boolean]
  C -->|No| E[exceptionFactory]
  E --> F[Throw configured exception]
  G[ParseBoolPipeOptions] --> B
  G --> H[errorHttpStatusCode]
  G --> I[exceptionFactory]
  G --> J[optional]
```

Usage

```
import {
  BadRequestException,
  ParseBoolPipe,
  type ParseBoolPipeOptions,
} from '@nestjs/common';

const options: ParseBoolPipeOptions = {
  errorHttpStatusCode: 422,
  optional: true,
  exceptionFactory: (error) =>
    new BadRequestException({
      message: 'Expected a boolean query parameter.',
      details: error,
    }),
};

const parseBoolean = new ParseBoolPipe(options);

// Useful for values such as "true", "false", true, or false.
const enabled = await parseBoolean.transform('true', {
  type: 'query',
  data: 'enabled',
  metatype: Boolean,
});
// enabled === true
```

API Coding Instructions

- Use `optional: true` when the value may be `null` or `undefined`; otherwise, invalid or absent values should trigger the configured exception.
- Provide an `exceptionFactory` when the application needs a consistent custom error response format.
- Use `errorHttpStatusCode` for the default exception behavior; it may be unnecessary when `exceptionFactory` creates its own HTTP exception.
- Apply these options when constructing `ParseBoolPipe` for route parameters, query parameters, or request body fields that must be parsed as booleans.