

ClassTransformOptions

August 17, 2026

ClassTransformOptions

Kind: Interface

Source: `packages/common/interfaces/external/class-transform-options.interface.ts`

Part of: [Common](#)

Options to be passed during transformation.

`ClassTransformOptions` defines configuration passed to class transformation operations, such as converting plain objects into class instances or serializing instances into plain data. It controls property exposure rules, groups, versioning, decorator behavior, circular-reference handling, and implicit type conversion.

Properties

Property	Type
<code>strategy</code>	<code>'excludeAll'</code>
<code>groups</code>	<code>string[]</code>
<code>version</code>	<code>number</code>
<code>excludePrefixes</code>	<code>string[]</code>
<code>ignoreDecorators</code>	<code>boolean</code>
<code>targetMaps</code>	<code>any[]</code>
<code>enableCircularCheck</code>	<code>boolean</code>
<code>enableImplicitConversion</code>	<code>boolean</code>
<code>excludeExtraneousValues</code>	<code>boolean</code>
<code>exposeDefaultValues</code>	<code>boolean</code>
<code>exposeUnsetFields</code>	<code>boolean</code>

Diagram

```
graph LR
  A[Transformation Operation] --> B[ClassTransformOptions]
  B --> C[strategy<br/>excludeAll | exposeAll]
  B --> D[groups and version]
  B --> E[excludePrefixes]
  B --> F[Decorator Controls]
  B --> G[Type Conversion]
  B --> H[Circular Reference Check]
  B --> I[Target Maps]

  F --> F1[ignoreDecorators]
  F --> F2[excludeExtraneousValues]
  F --> F3[exposeDefaultValues]
  G --> G1[enableImplicitConversion]
```

Usage

```
import { plainToInstance } from 'class-transformer';
import type { ClassTransformOptions } from '@nestjsjs/common';

class CreateUserDto {
  name: string;
  age: number;
}

const options: ClassTransformOptions = {
  strategy: 'excludeAll',
  enableImplicitConversion: true,
  excludeExtraneousValues: true,
  groups: ['create'],
  version: 1,
};

const payload = {
  name: 'Ada Lovelace',
  age: '36',
  internalToken: 'do-not-expose',
};

const user = plainToInstance(CreateUserDto, payload, options);

// With implicit conversion enabled, age can be converted to a number.
console.log(user.age);
```

AI Coding Instructions

- Pass `ClassTransformOptions` to transformation utilities consistently when converting request payloads, DTOs, or response objects.
- Use `excludeExtraneousValues` with explicit exposure decorators to prevent undeclared input properties from being included.
- Enable `enableImplicitConversion` only when automatic primitive conversion is intended; validate transformed values separately.
- Use `groups` and `version` to support context-specific or versioned serialization without duplicating DTO classes.
- Enable `enableCircularCheck` for object graphs that may contain circular references, noting that it can add processing overhead.