

Put

August 18, 2026

Put

Kind: Constant

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

Part of: [Common](#)

Route handler (method) Decorator. Routes HTTP PUT requests to the specified path.

`Put` is a route handler decorator that maps an HTTP `PUT` request to a controller method. Use it to define endpoints that update or replace a resource at a specific path, integrating the method with the framework's request-routing metadata.

Definition

```
createMappingDecorator(RequestMethod.PUT)
```

Value

```
createMappingDecorator(RequestMethod.PUT)
```

Diagram

```
graph LR
  Client[HTTP Client] -->|PUT /resources/:id| Router[HTTP Router]
  Router -->|Matches route metadata| Controller[Controller Method]
  Put["@Put('resources/:id')"] --> Controller
  Controller --> Response[HTTP Response]
```

Usage

```
import { Controller, Put, Body, Param } from '@nestjsjs/common';

@Controller('users')
export class UsersController {
  @Put(':id')
  updateUser(
    @Param('id') id: string,
    @Body() updateUserDto: UpdateUserDto,
  ) {
    return {
      id,
      ...updateUserDto,
    };
  }
}
```

AI Coding Instructions

- Apply `@Put()` only to controller methods that should handle HTTP `PUT` requests.
- Provide a path argument such as `@Put(':id')` when the route targets a specific resource; omit it only when using the controller-level path.
- Combine `@Put()` with parameter decorators such as `@Param()`, `@Body()`, and `@Query()` to access request data.
- Prefer `PUT` for full resource replacement or idempotent updates; use `PATCH` when supporting partial updates.
- Ensure the controller's base path and the `@Put()` path do not create unintended or duplicate route URLs.