

# GatewayTimeoutException

August 18, 2026

## GatewayTimeoutException

**Kind:** Class

**Source:** `packages/common/exceptions/gateway-timeout.exception.ts`

**Part of:** [Common](#)

Defines an HTTP exception for *Gateway Timeout* type errors.

`GatewayTimeoutException` represents an HTTP 504 Gateway Timeout error. Use it when an upstream service, proxy, or gateway does not respond within the expected time, allowing the application's exception handling layer to return a consistent HTTP error response.

**Extends:** `HttpException`

## Diagram

```
graph LR
  A[Application Handler] --> B[Calls Upstream Service]
  B -->|Request times out| C[GatewayTimeoutException]
  C --> D[Global Exception Filter]
  D --> E[HTTP 504 Gateway Timeout Response]
```

## Usage

```
import { GatewayTimeoutException } from '@nestjs/common';

async function fetchInventory(productId: string) {
  try {
    return await inventoryClient.get(`/products/${productId}`, {
      timeout: 5_000,
    });
  } catch (error) {
    if (error.code === 'ECONNABORTED') {
      throw new GatewayTimeoutException(
```

```
        'The inventory service did not respond in time.',  
    );  
}  
  
    throw error;  
}  
}
```

## AI Coding Instructions

- Throw `GatewayTimeoutException` when an upstream dependency exceeds its configured response deadline.
- Use this exception for HTTP 504 scenarios; do not use it for client request timeouts or malformed requests.
- Include a safe, actionable message without exposing upstream credentials, internal URLs, or sensitive error details.
- Ensure HTTP clients and service integrations define explicit timeout values so timeout failures can be handled consistently.
- Allow the framework's global exception filter to serialize the exception into the standard HTTP error response.