

UnprocessableEntityException

August 18, 2026

UnprocessableEntityException

Kind: Class

Source: `packages/common/exceptions/unprocessable-entity.exception.ts`

Part of: [Common](#)

Defines an HTTP exception for *Unprocessable Entity* type errors.

`UnprocessableEntityException` represents an HTTP 422 error, used when a request is syntactically valid but cannot be processed because its content fails validation or violates business rules. It extends the framework's HTTP exception infrastructure so controllers and services can return consistent client-facing error responses.

Extends: `HttpException`

Diagram

```
graph LR
    Client[Client Request] --> Controller[Controller / Service]
    Controller --> Validation{Valid request content?}
    Validation -->|No| Exception[UnprocessableEntityException]
    Exception --> Response[HTTP 422 Response]
    Validation -->|Yes| Handler[Continue processing]
```

Usage

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

function updateProfile(input: { email: string }) {
  if (!input.email.includes('@')) {
    throw new UnprocessableEntityException(
      'A valid email address is required.',
    );
  }
}
```

```
return { updated: true };  
}
```

AI Coding Instructions

- Throw `UnprocessableEntityException` when request data is structurally valid but fails semantic validation or business-rule checks.
- Prefer clear, actionable error messages that help API consumers correct the invalid input.
- Use HTTP 422 instead of `BadRequestException` when the request format is valid but a field value or state prevents processing.
- Allow the application's global exception filter to serialize the exception into the standard HTTP error response format.