

# normalizeLanguage

August 17, 2026

## normalizeLanguage

**Kind:** Function**Source:** `src/middleware/language/language.ts`**Part of:** [Middleware](#)

Validate and normalize language codes

`normalizeLanguage` validates an incoming language code and converts it to the format expected by the language middleware. It acts as a boundary between external request values and the normalized language value used by the application.

## Signature

```
function normalizeLanguage(lang: string | null | undefined, options: DetectorOptions): string |
```

## Parameters

| Name                 | Type                         |
|----------------------|------------------------------|
| <code>lang</code>    | <code>`string</code>         |
| <code>options</code> | <code>DetectorOptions</code> |

**Returns:** `string` | `undefined`

## Diagram

```
graph LR
  A[Incoming language code] --> B[normalizeLanguage]
  B --> C[{Valid code?}]
  C -->|Yes| D[Normalized language code]
  C -->|No| E[Fallback or validation result]
```

## Usage

```
import { normalizeLanguage } from './middleware/language/language';

const requestedLanguage = 'EN-us';
const language = normalizeLanguage(requestedLanguage);

console.log(language);
```

## AI Coding Instructions

- Pass request-derived language values through `normalizeLanguage` before using them for locale selection or translations.
- Keep language-code formatting rules in this function rather than duplicating case conversion or validation in middleware callers.
- Preserve the function's handling of invalid or missing values when changing language middleware behavior.
- Update callers and tests together if supported language codes or fallback behavior changes.