



LACSPACE

Developer Documentation

Backend Kit — server-side essentials

@lacspace/idempotency

Exactly-once operations

v1.0.0 · zero dependencies · isomorphic · generated 2026-08-25

Overview

The 'don't double-charge, don't send twice' primitive. Run any operation at-most-once per idempotency key and replay the stored result on retries — concurrency-safe (in-flight de-dupe + atomic create), with request fingerprinting and a pluggable store. Framework-agnostic (every other lib is Hono/AWS-locked).

NOTE

When to use — Making an operation safe to call more than once — payment capture, order creation, webhook handling — under retries or duplicate requests.

Package: @lacspace/idempotency · v1.0.0

Kit: Backend Kit — server-side essentials

Runtime: zero dependencies · isomorphic

Install

```
npm i @lacspace/idempotency
```

Quick example

```
await idempotent(key, () => charge(order))
```

Returns: { **value**, **replayed** }

Tips & warnings

TIP

Register the in-flight promise synchronously (before any await) so concurrent same-key calls share one execution — the package does this for you.

TIP

Return the same stored result for a repeated key within the TTL window.

TIP

Zero-dependency and isomorphic — safe to import on the server, in the browser, on the edge and in React Native. It tree-shakes, so you only ship what you import.

WARNING

The default store is in-memory (per process). For multi-instance deployments, back it with a shared store (e.g. Redis).

Links & full reference

- npm: <https://www.npmjs.com/package/@lacspace/idempotency>
- Source & full README: <https://github.com/lacspace/npm-packages/tree/main/idempotency>
- Docs: lacspace.com/docs/idempotency

The npm page and GitHub README carry the complete API reference and more examples.