

IntrinsicException

August 17, 2026

IntrinsicException

Kind: Class

Source: `packages/common/exceptions/intrinsic.exception.ts`

Part of: `Common`

Exception that represents an intrinsic error in the application.

When thrown, the default exception filter will not log the error message.

`IntrinsicException` represents an expected, application-level error that should not be treated as an unexpected failure. When this exception is thrown, the default exception filter handles the response without logging the exception message, making it suitable for controlled or non-actionable errors.

Extends: `Error`

Diagram

```
graph LR
  A[Application code] -->|throw new IntrinsicException| B[IntrinsicException]
  B --> C[Default Exception Filter]
  C --> D[Error response returned]
  C -. does not log message .-> E[Application logs]
```

Usage

```
import { IntrinsicException } from '@app/common/exceptions/intrinsic.exception';

export class AccountService {
  async getAccount(accountId: string) {
    const account = await this.accountRepository.findById(accountId);

    if (!account) {
      throw new IntrinsicException('Account is unavailable');
    }
  }
}
```

```
    return account;
  }
}
```

AI Coding Instructions

- Use `IntrinsicException` for expected application errors that should not produce default exception-filter log messages.
- Provide a clear, safe message suitable for the client or the application's error-handling flow.
- Do not use this exception for unexpected infrastructure failures, such as database connectivity or unhandled programming errors.
- Ensure the default exception filter remains registered so `IntrinsicException` receives its intended no-log handling behavior.

Used by

4 references from 4 files. Each is a place in this repository where the symbol is actually used — go read one rather than trusting an example.

Imported by (4)

- `BaseExceptionHandler` — `packages/core/exceptions/base-exception-filter.ts :17`
- `ExternalExceptionHandler` — `packages/core/exceptions/external-exception-filter.ts :3`
- `BaseRpcExceptionHandler` — `packages/microservices/exceptions/base-rpc-exception-filter.ts :15`
- `ErrorPayload` — `packages/websockets/exceptions/base-ws-exception-filter.ts :11`