



Program Listing for File errors.hpp

[Return to documentation for file \(include/holoscan/core/errors.hpp\)](#)

```
/* * SPDX-FileCopyrightText: Copyright (c) 2023 NVIDIA CORPORATION & AFFILIATES.
All rights reserved. * SPDX-License-Identifier: Apache-2.0 * * Licensed under the
Apache License, Version 2.0 (the "License"); * you may not use this file except in
compliance with the License. * You may obtain a copy of the License at * *
http://www.apache.org/licenses/LICENSE-2.0 * * Unless required by applicable law
or agreed to in writing, software * distributed under the License is distributed on an
"AS IS" BASIS, * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express or implied. * See the License for the specific language governing
permissions and * limitations under the License. */ #ifndef
HOLOSCAN_CORE_ERRORS_HPP #define HOLOSCAN_CORE_ERRORS_HPP #include
<stdexcept> #include <string> namespace holoscan { enum class ErrorCode {
kSuccess = 0, // No error kFailure = 1, // Generic Holoscan SDK error kReceiveError = 2,
// InputContext's receive() method errors kCodecError = 3, // codec's serialize(),
deserialize() method errors kInvalidArgument = 4, // Invalid argument kNotFound = 5,
// Not found kDuplicateName = 6, // Duplicate name kErrorCodeCount // Number of
error codes }; class RuntimeError : public std::runtime_error { public: // Inherit
constructors from std::runtime_error using std::runtime_error::runtime_error; explicit
RuntimeError(holoscan::ErrorCode error_code); RuntimeError(holoscan::ErrorCode
error_code, const std::string& what_arg); RuntimeError(holoscan::ErrorCode
error_code, const char* what_arg); static const char* error_string(const
holoscan::ErrorCode error_code); private: // Helper function to construct the error
message. static std::string construct_error_message(const holoscan::ErrorCode
error_code, const char* what_arg); static const char* ErrorStrings[]; }; } // namespace
holoscan #endif/* HOLOSCAN_CORE_ERRORS_HPP */
```

© Copyright 2022-2024, NVIDIA.. PDF Generated on 06/06/2024