

API Collection

Encoding, Decoding, and Serialization

Serialize and deserialize instances of your types with implicit or customized encoding.

Topics

Custom Encoding and Decoding

Encoding and Decoding Custom Types

Make your data types encodable and decodable for compatibility with external representations such as JSON.

`typealias Codable`

A type that can convert itself into and out of an external representation.

`protocol Encodable`

A type that can encode itself to an external representation.

`protocol Decodable`

A type that can decode itself from an external representation.

`protocol CodingKey`

A type that can be used as a key for encoding and decoding.

`protocol CodingKeyRepresentable`

A type that can be converted to and from a coding key.

`struct CodingUserInfoKey`

A user-defined key for providing context during encoding and decoding.

Encoders and Decoders

`protocol Encoder`

A type that can encode values into a native format for external representation.

`protocol Decoder`

A type that can decode values from a native format into in-memory representations.

`enum EncodingError`

An error that occurs during the encoding of a value.

`enum DecodingError`

An error that occurs during the decoding of a value.

Encoding Containers

`protocol SingleValueEncodingContainer`

A container that can support the storage and direct encoding of a single non-keyed value.

`struct KeyedEncodingContainer`

A concrete container that provides a view into an encoder's storage, making the encoded properties of an encodable type accessible by keys.

`protocol KeyedEncodingContainerProtocol`

A type that provides a view into an encoder's storage and is used to hold the encoded properties of an encodable type in a keyed manner.

`protocol UnkeyedEncodingContainer`

A type that provides a view into an encoder's storage and is used to hold the encoded properties of an encodable type sequentially, without keys.

Decoding Containers

`struct KeyedDecodingContainer`

A concrete container that provides a view into a decoder's storage, making the encoded properties of a decodable type accessible by keys.

`protocol SingleValueDecodingContainer`

A container that can support the storage and direct decoding of a single nonkeyed value.

`protocol KeyedDecodingContainerProtocol`

A type that provides a view into a decoder's storage and is used to hold the encoded properties of a decodable type in a keyed manner.

`protocol UnkeyedDecodingContainer`

A type that provides a view into a decoder's storage and is used to hold the encoded properties of a decodable type sequentially, without keys.

See Also

Tools for Your Types



Basic Behaviors

Use your custom types in operations that depend on testing for equality or order and as members of sets and dictionaries.



Initialization with Literals

Allow values of your type to be expressed using different kinds of literals.