ProximityReader / PaymentCardReader

Class

PaymentCardReader

An object you use to configure Tap to Pay on iPhone on the current device.

iOS 15.4+ | iPadOS 15.4+ | Mac Catalyst 17.0+

```
class PaymentCardReader
```

Mentioned in

Adding support for Tap to Pay on iPhone to your app

Overview

A `PaymentCardReader` object coordinates the reading of payment and loyalty cards using existing iPhone hardware. Use it to read cards that might otherwise require specialized hardware. For example, use it to collect contactless payment information from another iPhone or from NFC-enabled hardware, such as NFC-enabled plastic cards.

After you create the `PaymentCardReader` object, call `prepare(using:)` to validate the payment pipeline and receive a new `PaymentCardReaderSession`. Use the session object to read card information related to payments and refunds, or to verify card information.

Topics

Creating a payment reader

```
init(options: PaymentCardReader.Options)
```

Creates a payment card reader with the specified options.

```
struct Options
```

Additional information you use to configure a payment card reader.

Getting the feature availability

```
static let isSupported: Bool
```

A Boolean value that indicates whether this device model supports Tap to Pay on iPhone.

Configuring Tap to Pay on iPhone

```
func prepare(using: PaymentCardReader.Token) async throws -> PaymentCardReaderSession
```

Configures the pipeline for reading payment or loyalty cards.

Displaying the Tap to Pay on iPhone’s terms and conditions

```
func isAccountLinked(using: PaymentCardReader.Token) async throws -> Bool
```

A Boolean value that indicates whether the account is already linked.

```
func linkAccount(using: PaymentCardReader.Token) async throws
```

Presents a sheet for the merchant to accept Tap to Pay on iPhone’s Terms and Conditions on a device.

```
func relinkAccount(using: PaymentCardReader.Token) async throws
```

Presents a sheet for the merchant to re-accept Tap to Pay on iPhone’s Terms and Conditions on a device using a different Apple Account.

```
struct Token
```

A secure token that you receive from your participating payment service provider.

Observing reader events

```
let events: AsyncStream<PaymentCardReader.Event>
```

A stream of events you receive indicating the activities of the payment card reader.

```
enum Event
```

An event you receive indicating the state or activity of the payment card reader.

Getting the configuration details

`var readerIdentifier: String`

The unique identifier for this card reader.

`let options: PaymentCardReader.Options`

The defined configuration settings when the reader was created.

Deprecated

~~`let id: String`~~

A unique identifier for this object.

Deprecated

~~`func prepare(using: PaymentCardReader.Token, updateHandler: ((PaymentCardReader.UpdateEvent) -> Void)?) async throws -> PaymentCardReaderSession`~~

Configures the pipeline for reading payment or loyalty cards.

Deprecated

~~`enum UpdateEvent`~~

An event you receive during the configuration of the payment system.

Deprecated

Instance Methods

`func fetchPaymentCardReaderStore() throws -> PaymentCardReaderStore`

Returns a store containing the read results the framework obtained using a Store and Forward session.

`func prepareStoreAndForward() async throws -> StoreAndForwardPaymentCardReaderSession`

Configures the pipeline for reading payment or loyalty cards in Store and Forward mode.

Relationships

Conforms To

`Sendable`, `SendableMetatype`

See Also

Payment card reader

Setting up Tap to Pay on iPhone

Request and configure the required entitlement to support Tap to Pay on iPhone.

Adding support for Tap to Pay on iPhone to your app

Configure your app to use Tap to Pay on iPhone to read contactless payment cards.

`class PaymentCardReaderSession`

The object you use to start reading a contactless payment or loyalty card.