StoreKit / ··· / finishTransaction(_:) Deprecated

Instance Method

finishTransaction(_:) Deprecated

Notifies the App Store that the app finished processing the transaction.

iOS 3.0–18.0 Deprecated | iPadOS 3.0–18.0 Deprecated | Mac Catalyst 13.1–18.0 Deprecated | macOS 10.7–15.0 Deprecated | tvOS 9.0–18.0 Deprecated | visionOS 1.0–2.0 Deprecated | watchOS 6.2–11.0 Deprecated

```
func finishTransaction(_ transaction: SKPaymentTransaction)
```

Deprecated

Use Transaction.finish().

Parameters

`transaction`

The transaction to finish.

Mentioned in

- Testing In-App Purchases in Xcode
- Finishing a transaction
- Implementing offer codes in your app
- Testing an interrupted purchase
- Testing complete transactions

Discussion

Transactions on the payment queue are persistent until they are completed. StoreKit calls your observer's `paymentQueue(_:updatedTransactions:)` method every time your app launches or resumes from background to tell you about transactions in the queue. After you've finished processing a transaction in your app, always call the `finishTransaction(_:)` method to finish the transaction and remove it from the queue.

Call `finishTransaction(_:)` only after the app has finished all work it performs to complete the transaction. The transaction's state determines which steps you might take:

- For a failed transaction (`SKPaymentTransactionState.failed`), update your user interface, track information in analytics, and perform other similar tasks.

- For a successful transaction (`SKPaymentTransactionState.purchased` or `SKPaymentTransactionState.restored`), perform all necessary actions to unlock the functionality the user has purchased before finishing the transaction. For example, if you are downloading content, finish the transaction only after the downloads are complete.

If you validate receipts, validate them before completing the transaction, and take one of the paths described above.

In rare circumstances, this call might fail, and you'll receive updates for that transaction again. For this reason, you should record information in your app about the transactions it has processed and which steps the app has already completed. That way, you don't repeat steps that shouldn't be performed multiple times. For example, if you are processing a consumable transaction, you only want to add the consumable benefit once.

If you call `finishTransaction(_:)` on a transaction that is in the `SKPaymentTransactionState.purchasing` state, StoreKit raises an exception.

See Also

Related Documentation

~~`func paymentQueue(SKPaymentQueue, updatedTransactions: [SKPaymentTransaction])`~~

Tells an observer that one or more transactions have been updated.

Required

Deprecated

Managing Transactions

~~`var delegate: (any SKPaymentQueueDelegate)?`~~

A delegate that provides information needed to complete transactions.

Deprecated

~~`var transactions: [SKPaymentTransaction]`~~

Returns an array of pending transactions.

Deprecated

~~`func add(SKPayment)`~~

Adds a payment request to the queue.

Deprecated