

[StoreKit](#) / Product

## Structure



# Product

Information about a product that you configure in App Store Connect.

iOS 15.0+ | iPadOS 15.0+ | macOS 12.0+ | tvOS 15.0+ | visionOS 1.0+ | watchOS 8.0+

```
struct Product
```

## Mentioned in

-  Supporting Family Sharing in your app
-  Supporting promoted In-App Purchases in your app

## Overview

The Product type represents the in-app purchases that you configure in App Store Connect and make available for purchase within your app. Use Product to perform all product-related tasks in your app, from displaying in-app purchases and offers to making a purchase and getting transaction and subscription status information.

To get a Product instance, call `products(for:)` and provide one or more in-app purchase product identifiers. Use a Product instance to display in-app purchases and subscription offers in your store, as follows:

- Show the localized name, description, and pricing information using `displayName`, `description`, and `displayPrice`, respectively.
- Determine whether a user is eligible for an introductory offer for the product using `isEligibleForIntroOffer`.
- Display your subscription offers using the subscription information in `subscription`.

When users initiate a purchase, call `purchase(options:)` or `purchase(confirmIn: options:)` on the product instance. If your app uses SwiftUI, you can also use `PurchaseAction`. Set purchase options (`Product.PurchaseOption`) to define an optional app account token, apply a promotional offer, or set a product quantity. Purchase options can also simulate an Ask to Buy scenario when you're testing your app in the sandbox environment.

Use a `Product` instance to learn whether a user is entitled to a product by checking `currentEntitlement`, which holds the transaction that entitles the user to the product. This transaction information, as well as the transaction in `latestTransaction`, are cryptographically signed by the App Store in JSON Web Signature (JWS) format.

If the product is an auto-renewable subscription, use the `status` and `renewalInfo` in the `subscription` information to help manage subscriptions and inform business decisions, such as presenting subscription offers.

For information about configuring In-App Purchases in App Store Connect, see [Overview for configuring In-App Purchases](#).

---

## Topics

### Requesting products from the App Store

```
static func products<Identifiers>(for: Identifiers) async throws -> [Product]
```

Requests product data from the App Store.

### Displaying a product description and price

```
let displayName: String
```

The localized display name of the product, if it exists.

```
let description: String
```

The localized description of the product.

```
let displayPrice: String
```

The localized string representation of the product price, suitable for display.

```
let price: Decimal
```

The decimal representation of the cost of the product, in local currency.

```
var priceFormatStyle: Decimal.FormatStyle.Currency
```

The format style for the numbers in the price of the product.

```
var subscriptionPeriodFormatStyle: Date.ComponentsFormatStyle
```

The format style for the date components related to a subscription's duration.

```
var subscriptionPeriodUnitFormatStyle: Product.SubscriptionPeriod.Unit.  
FormatStyle
```

The format style for subscription period units, such as week, month, or year.

## Purchasing a product

```
func purchase(options: Set<Product.PurchaseOption>) async throws ->  
Product.PurchaseResult
```

Initiates a purchase for the product with the App Store and displays the confirmation sheet.

```
func purchase(confirmIn: some UIScene, options: Set<Product.Purchase  
Option>) async throws -> Product.PurchaseResult
```

Initiates a purchase for the product with the App Store and displays the confirmation sheet.

```
func purchase(confirmIn: UIViewController, options: Set<Product.  
PurchaseOption>) async throws -> Product.PurchaseResult
```

Processes a purchase for the product.

```
func purchase(confirmIn: NSWindow, options: Set<Product.PurchaseOption  
>) async throws -> Product.PurchaseResult
```

Processes a purchase for the product.

```
struct PurchaseOption
```

Optional settings for a product purchase that add account information, purchase details, and offers, or that specify behaviors.

```
enum PurchaseResult
```

The result of a purchase.

```
enum PurchaseError
```

Error information for product purchase errors.

## Receiving current entitlement information

```
var currentEntitlement: VerificationResult<Transaction>?
```

The transaction that entitles the user to the product.

Deprecated

## Getting the latest transaction

```
var latestTransaction: VerificationResult<Transaction>?
```

The most recent transaction for the product.

## Getting subscription information

```
let subscription: Product.SubscriptionInfo?
```

The subscription information for an auto-renewable subscription.

```
struct SubscriptionInfo
```

Information about an auto-renewable subscription, such as its status, period, subscription group, and subscription offer details.

```
struct SubscriptionPeriod
```

Values that represent the duration of time between subscription renewals.

```
struct SubscriptionOffer
```

Information about a subscription offer that you configure in App Store Connect.

```
struct Status
```

The renewal status information for an auto-renewable subscription.

## Getting product identifiers and type

```
let id: String
```

The unique product identifier.

```
let type: Product.ProductType
```

The in-app purchase product type.

```
struct ProductType
```

The types of in-app purchases.

## Getting Family Sharing status

```
let isFamilyShareable: Bool
```

A Boolean value that indicates whether the product is available for Family Sharing in App Store Connect.

## Managing promoted in-app purchases

`struct PromotionInfo`

Information about a promoted In-App Purchase that customizes its order and visibility on the device.

## Loading products

`enum CollectionTaskState`

The state of a task that loads a collection of products in the background.

`enum TaskState`

The state of a task that loads a product in the background.

## Getting product info in JSON format

`var jsonRepresentation: Data`

The JSON representation of the product information.

## Getting subscription relationship

`struct SubscriptionRelationship`

## Instance Properties

`var currentEntitlements: Transaction.Transactions`

---

# Relationships

## Conforms To

Copyable

CustomDebugStringConvertible

Equatable

Hashable

Identifiable

Sendable  
SendableMetatype

---

## See Also

### Product and subscription information

`{}` Implementing a store in your app using the StoreKit API  
Offer In-App Purchases and manage entitlements using signed transactions and status information.

`struct SubscriptionInfo`  
Information about an auto-renewable subscription, such as its status, period, subscription group, and subscription offer details.

`typealias SubscriptionInfo`  
Information about an auto-renewable subscription.

`typealias SubscriptionStatus`  
Represents the renewal status information for an auto-renewable subscription.