

Documentation

[Apple Pay on the Web](#) / [Apple Pay JS API](#) / Checking for Apple Pay availability

Article

Checking for Apple Pay availability

Use the Apple Pay JS API to check whether Apple Pay is available, to check whether a device has a payment credential provisioned in Wallet, and to determine when to display an Apple Pay button.

Overview

Before displaying an Apple Pay button or creating an Apple Pay session, ensure that the Apple Pay JS API is available and enabled on the current device. To check whether the Apple Pay JS API is available in the browser, verify that the `window.ApplePaySession` class exists:

```
if (window.ApplePaySession) {  
    // The Apple Pay JS API is available.  
}
```

Next, check whether payments are possible. Call either the [`canMakePayments`](#) or [`applePayCapabilities`](#) methods, as follows:

[canMakePayments](#)

This method verifies that the device is capable of making Apple Pay payments; it doesn't verify that the person has a provisioned card for use with Apple Pay on the device. You can call this method at any time.

[applePayCapabilities](#)

This method verifies both that the device is capable of making Apple Pay payments, and that the person has at least one provisioned card. This method asynchronously contacts the Apple Pay servers as part of the verification process. You can only call this method if you want to default to Apple Pay during your checkout flow, or if you want to add an Apple Pay button to your product detail page.

Verify whether the device supports Apple Pay

According to the [Apple Pay on the Web Acceptable Use Guidelines](#), if you invoke the `applePayCapabilities` API and determine that a person has an active card provisioned into Wallet, then Apple Pay should be the primary, but not necessarily sole payment option on any webpage that accepts payments, like the checkout page or product detail page. In all other situations, use `canMakePayments` instead.

The `applePayCapabilities` method asynchronously contacts Apple Pay servers as part of the verification process and returns the `paymentCredentialStatus` of the device. This verifies on Safari and third-party browsers that the device is capable of making Apple Pay payments. It also verifies on Safari browsers that the device has at least one payment credential provisioned in Wallet. Depending on the response, you can determine if the device supports Apple Pay and whether to display an Apple Pay button.

The following are possible return values:

`paymentCredentialsAvailable`

Confirms that the device supports Apple Pay and there's at least one active payment credential in Wallet that qualifies for payments on the web. Show an Apple Pay button and offer Apple Pay as the primary, but not necessarily sole, payment option.

`paymentCredentialStatusUnknown`

Confirms that the device supports Apple Pay, but the Wallet information is unknown. Show an Apple Pay button and offer Apple Pay as a possible payment option.

`paymentCredentialsUnavailable`

Confirms that the device supports Apple Pay and that the device doesn't have any active payment credentials in Wallet. Offer Apple Pay as a payment option, but don't make it the sole or default payment option. This gives people the option to set up Apple Pay as part of their purchase.

`applePayUnsupported`

Indicates that the device doesn't support Apple Pay. Don't display an Apple Pay button or offer Apple Pay as a payment option.

The following code shows how to check that a payment method is available before displaying an Apple Pay button:

```
// Check if the Apple Pay JS API is available.  
if (window.ApplePaySession) {  
    var merchantIdentifier = 'YOUR MERCHANT IDENTIFIER';  
    var promise = ApplePaySession.applePayCapabilities(merchantIdentifier);  
    promise.then(function(capabilities) {  
        // Check if the person has an active payment credential provisioned in Wallet.  
        switch (capabilities.paymentCredentialStatus) {
```

```
        case "paymentCredentialsAvailable":  
            // Display an Apple Pay button and offer Apple Pay as the primary payment  
        case "paymentCredentialStatusUnknown":  
            // Display an Apple Pay button and offer Apple Pay as a payment option.  
        case "paymentCredentialsUnavailable":  
            // Consider displaying an Apple Pay button.  
        case "applePayUnsupported":  
            // Don't show an Apple Pay button or offer Apple Pay.  
    }  
}  
})  
}
```

Verify Apple Pay availability in China

To provide Apple Pay on the web for customers in China, you must first verify compatibility by checking the browser's userAgent string. This string provides information about the customer's device and operating system.

For the Chinese market, you need to check the user agent string to confirm that:

- The person's device is an "iPhone" or an "iPad".
- The device is running iOS 11.2 or later.

For example, the following shows a user agent string received from Safari on an iPhone running iOS 11.2:

```
Mozilla/5.0 (iPhone; CPU iPhone OS 11_2 like Mac OS X) AppleWebKit/604.1.34 (KHTML,
```

After verifying the person's device type and iOS version, call [canMakePayments](#) or [applePayCapabilities](#) as shown earlier, and display the Apple Pay button if the call returns `paymentCredentialsAvailable`.

Note

Don't display the Apple Pay button if the device is a Mac, if the iOS version is earlier than 11.2, or if the call to [canMakePayments](#) returns false.

See Also

[Apple Pay availability](#)

canMakePayments

Indicates whether the device supports Apple Pay.

applePayCapabilities

Indicates whether the device supports Apple Pay and whether the person has an active card in Wallet that qualifies for web payments.

~~canMakePaymentsWithActiveCard~~

Indicates whether the device supports Apple Pay and whether the user has an active card in Wallet.

Deprecated

PaymentCredentialStatus

Information about whether the device supports Apple Pay and the possible payment credentials the person has provisioned in Wallet.

PaymentCredentialStatusResponse

The response for information about the device's support for Apple Pay and the payment credential status.