



PaymentGatewayAsyncAdapter Interface

Implement the interface to allow customers to process payments asynchronously.

Namespace

[CommercePayments](#)

Usage

Implementing an asynchronous adapter also requires the `processNotification` method from [GatewayNotificationResponse](#) class.

Example

```
global with sharing class SampleAsyncAdapter
    implements commercepayments.PaymentGatewayAsyncAdapter,
               commercepayments.PaymentGatewayAdapter {

    global SampleAsyncAdapter() {
    }

    global commercepayments.GatewayResponse processRequest(
        commercepayments.PaymentGatewayContext gatewayContext) {
    }

    global commercepayments.GatewayNotificationResponse processNotification(
        commercepayments.PaymentGatewayNotificationContext gatewayNotificationContext) {
    }
}
```

- [PaymentGatewayAsyncAdapter Methods](#)
- [PaymentGatewayAsyncAdapter Example Implementation](#)

PaymentGatewayAsyncAdapter Methods

The following are methods for `PaymentGatewayAsyncAdapter`.

- [processNotification\(paymentGatewayNotificationContext\)](#)
Entry point for processing notifications from payment gateways.

`processNotification(paymentGatewayNotificationContext)`

Entry point for processing notifications from payment gateways.

Signature

```
global commercepayments.GatewayNotificationResponse
processNotification(commercepayments.PaymentGatewayNotificationContext var1)
```



The `PaymentGatewayNotificationContext` object wraps all the information related to a gateway notification.

Return Value

Type: `GatewayNotificationResponse`

When the payment gateway sends a notification to the payments platform, the platform responds with a `GatewayNotificationResponse` indicating whether the platform succeeded or failed at receiving the notification.

PaymentGatewayAsyncAdapter Example Implementation

This is a sample implementation of the `commercepayments.PaymentGatewayAsyncAdapter` interface.

```
global with sharing class AdyenAdapter implements commercepayments.PaymentGatewayAsyncAdapter {
    global AdyenAdapter() {}

    global commercepayments.GatewayResponse processRequest(commercepayments.PaymentGatewayAsyncContext ctx) {
        // Implementation for processRequest
    }

    global commercepayments.GatewayNotificationResponse processNotification(commercepayments.PaymentGatewayNotificationContext ctx) {
        // Implementation for processNotification
    }
}

commercepayments.RequestType requestType = gatewayContext.getPaymentRequestType();
if (requestType == commercepayments.RequestType.Capture) {
    req.setEndpoint('/pal/servlet/Payment/v52/capture');
    body = buildCaptureRequest((commercepayments.CaptureRequest)gatewayContext.getPaymentRequest());
} else if (requestType == commercepayments.RequestType.ReferencedRefund) {
    req.setEndpoint('/pal/servlet/Payment/v52/refund');
    body = buildRefundRequest((commercepayments.ReferencedRefundRequest)gatewayContext.getPaymentRequest());
}

req.setBody(body);
req.setMethod('POST');
commercepayments.PaymentsHttp http = new commercepayments.PaymentsHttp();
HttpResponse res = null;
try {
    res = http.send(req);
} catch (CalloutException ce) {
    commercepayments.GatewayErrorResponse error = new commercepayments.GatewayErrorResponse(ce.getMessage());
    return error;
}

if (requestType == commercepayments.RequestType.Capture) {
    response = createCaptureResponse(res);
} else if (requestType == commercepayments.RequestType.ReferencedRefund) {
    response = createRefundResponse(res);
}
return response;

commercepayments.PaymentGatewayNotificationRequest notificationRequest = gatewayContext.getNotificationRequest();
Blob request = notificationRequest.getRequestBody();
Map<String, Object> jsonReq = (Map<String, Object>)JSON.deserializeUntyped(request.getBytes());
List<Object> notificationItems = (List<Object>)jsonReq.get('notificationItems');
Map<String, Object> notificationRequestItem = (Map<String, Object>)((List<Object>)notificationItems[0]).get('notificationRequestItem');
Boolean success = Boolean.valueOf(notificationRequestItem.get('success'));
String pspReference = (String)notificationRequestItem.get('pspReference');
String eventCode = (String)notificationRequestItem.get('eventCode');
Double amount = (Double)((Map<String, Object>)notificationRequestItem.get('amount')).get('amount');

commercepayments.NotificationStatus notificationStatus = null;
if (success) {
```



```
notification = new commercepayments.CaptureNotification();
} else if ('REFUND'.equals(eventCode)) {
    notification = new commercepayments.ReferencedRefundNotification();
}
notification.setStatus(notificationStatus);
notification.setGatewayReferenceNumber(merchantReference);
notification.setAmount(amount);

commercepayments.NotificationSaveResult saveResult = commercepayments.Notification

commercepayments.GatewayNotificationResponse gnr = new commercepayments.GatewayNot
if (saveResult.isSuccess()) {
    system.debug('Notification accepted by platform');
} else {
    system.debug('Errors in the result ' + Blob.valueOf(saveResult.getErrorMessage));
}
gnr.setStatusCode(200);
gnr.setResponseBody(Blob.valueOf('[' + accepted + ']'));
return gnr;
```

DID THIS ARTICLE SOLVE YOUR ISSUE?
Let us know so we can improve!

Share your feedback



- DEVELOPER CENTERS
- Heroku
 - MuleSoft
 - Tableau
 - Commerce Cloud
 - Lightning Design System
 - Einstein
 - Quip

- POPULAR RESOURCES
- Documentation
 - Component Library
 - APIs
 - Trailhead
 - Sample Apps
 - Podcasts
 - AppExchange

- COMMUNITY
- Trailblazer Connect
 - Events and Webinars
 - Partner Community
 - Blog
 - Salesforce Answers
 - Salesforce AppExchange

© Copyright 2025 Salesforce, Inc. All rights reserved. Various trademarks held by their respective owners. Salesforce, Inc. Salesforce Tower, 415 Mission Street, 3rd Floor, San Francisco, CA 94105, United States

[Privacy Information](#) [Terms of Service](#) [Legal](#) [Use of Cookies](#) [Trust](#) [Cookie Preferences](#)
 [Your Privacy Choices](#) [Responsible Disclosure](#) [Contact](#)