



AuthorizationRequest Class

Sends information about an authorization request to a gateway adapter during a service call. This class extends the `BaseRequest` class and inherits all its methods.

Namespace

[CommercePayments](#)

Usage

This class contains information about a transaction authorization request. The gateway adapter reads fields from this class while constructing an authorization JSON request to send to the payment gateway. An object of this class is available by calling `getPaymentRequest()` in the [PaymentGatewayContext Class](#).

Example

Creating a `buildAuthRequest` class to store information about the authorization request.

```
private String buildAuthRequest(commercepayments.AuthorizationRequest authRequest) {
    // Multiply amount by 100.0 to convert to cents
    String requestBody = createRequestBody(String.valueOf((authRequest.amount*100.0)));
    return requestBody;

    private String createRequestBody(String amount, commercepayments.AuthorizationRequest authRequest) {
        JSONGenerator jsonGeneratorInstance = JSON.createGenerator(true);
        String currencyIso = authRequest.currencyIsoCode;
        commercepayments.AuthApiPaymentMethodRequest paymentMethod = authRequest.paymentMethod;
        commercepayments.GatewayErrorResponse error;
        // Write data to the JSON string.
        jsonGeneratorInstance.writeStartObject();
        jsonGeneratorInstance.writeStringField('merchantAccount', '{!$Credential.Username}');
        jsonGeneratorInstance.writeStringField('reference', authRequest.comments == null ? '' : authRequest.comments);

        if(currencyIso == null) {
            currencyIso = UserInfo.getDefaultCurrency();
        }

        jsonGeneratorInstance.writeFieldName('amount');
        jsonGeneratorInstance.writeStartObject();
        jsonGeneratorInstance.writeStringField('value', amount);
        jsonGeneratorInstance.writeStringField('currency', currencyIso);
        jsonGeneratorInstance.writeEndObject();

        commercepayments.CardPaymentMethodRequest cardPaymentMethod;
        if(paymentMethod != null) {
            cardPaymentMethod = paymentMethod.cardPaymentMethod;
            if (cardPaymentMethod != null) {
                if (cardPaymentMethod.CardCategory != null) {
                    if (commercepayments.CardCategory.CreditCard == cardPaymentMethod.CardCategory) {
                        jsonGeneratorInstance.writeFieldName('card');
                        jsonGeneratorInstance.writeStartObject();
                        if (cardPaymentMethod.cvv != null) {
                            jsonGeneratorInstance.writeStringField('cvc', String.valueOf(cardPaymentMethod.cvv));
                        }
                        if (cardPaymentMethod.cardholdername != null) {
                            jsonGeneratorInstance.writeStringField('cardholdername', cardPaymentMethod.cardholdername);
                        }
                        jsonGeneratorInstance.writeEndObject();
                    }
                }
            }
        }
    }
}
```



```

        jsonGeneratorInstance.writeStringField('expiryYear', String.valueOf(
    }
    jsonGeneratorInstance.writeEndObject();
    } else {
        //Support for other card type
    }
    } else {
        throw new SampleValidationException('Required Field Missing : CardCategory');
    }
    } else {
        throw new SampleValidationException('Required Field Missing : CardPaymentMethod');
    }
    } else {
        throw new SampleValidationException('Required Field Missing : PaymentMethod');
    }
    jsonGeneratorInstance.writeEndObject();
    return jsonGeneratorInstance.getAsString();
}

```

- [AuthorizationRequest Constructors](#)
- [AuthorizationRequest Properties](#)
- [AuthorizationRequest Methods](#)

AuthorizationRequest Constructors

The following are constructors for `AuthorizationRequest`.

- [AuthorizationRequest\(amount\)](#)
Constructor for building the amount in an authorization request. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

AuthorizationRequest(amount)

Constructor for building the amount in an authorization request. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

Signature

```
global AuthorizationRequest(Double amount)
```

Parameters

amount

Type: [Double](#)

The amount of the authorization.

AuthorizationRequest Properties

The following are properties for `AuthorizationRequest`.

- [accountId](#)
The customer account where the authorization is performed.
- [amount](#)
The total amount of the authorization. Can be positive or negative.
- [comments](#)
Comments about the authorization. Users can enter comments to provide additional information.
- [currencyIsoCode](#)
The ISO currency code for the authorization request.



The customer account where the authorization is performed.

Signature

```
global String accountId {get; set;}
```

Property Value

Type: [String](#)

amount

The total amount of the authorization. Can be positive or negative.

Signature

```
global Double amount {get; set;}
```

Property Value

Type: [Double](#)

comments

Comments about the authorization. Users can enter comments to provide additional information.

Signature

```
global String comments {get; set;}
```

Property Value

Type: [String](#)

currencyIsoCode

The ISO currency code for the authorization request.

Signature

```
global String currencyIsoCode {get; set;}
```

Property Value

Type: [String](#)

paymentMethod

The payment method used to process the authorization in the authorization request.

Signature

```
global AuthApiPaymentMethodRequest paymentMethod {get; set;}
```

Property Value

Type: [AuthApiPaymentMethodRequest](#)

AuthorizationRequest Methods

The following are methods for `AuthorizationRequest`.



- **hashCode()**
Maintains the integrity of lists of type `AuthorizationRequest` by determining the uniqueness of the external object in a list.
- **toString()**
Converts a date to a string.

equals(obj)

Maintains the integrity of lists of type `AuthorizationRequest` by determining the equality of external objects in a list. This method is dynamic and based on the `equals` method in Java.

Signature

```
global Boolean equals(Object obj)
```

Parameters

obj

Type: `Object`

External object whose key is to be validated.

Return Value

Type: `Boolean`

hashCode()

Maintains the integrity of lists of type `AuthorizationRequest` by determining the uniqueness of the external object in a list.

Signature

```
global Integer hashCode()
```

Return Value

Type: `Integer`

toString()

Converts a date to a string.

Signature

```
global String toString()
```

Return Value

Type: `String`

DID THIS ARTICLE SOLVE YOUR ISSUE?

Let us know so we can improve!

[Share your feedback](#)



MuleSoft
Tableau
Commerce Cloud
Lightning Design System
Einstein
Quip

Component Library
APIs
Trailhead
Sample Apps
Podcasts
AppExchange

Events and Calendar
Partner Community
Blog
Salesforce Admins
Salesforce Architects

© 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](#)