

# Creating a payment module

## Creating a payment module

### Principles

A payment module is a regular PrestaShop module, except that it extends the `PaymentModule` class instead of the `Module` class:

```
class MyOwnPaymentMethod extends PaymentModule
```

It can be attached to the following hooks:

- `payment`: to display the payment method.
- `paymentReturn`: to display the payment confirmation.

The handling of the payment itself is done through a controller or a remote server.

If you are using a remote server, a return URL makes it possible to execute a script from your module's folder in order to validate and register the payment.

### Validating the payment

In order to register the payment validation, you must use the `validateOrder()` method from the `PaymentModule` class, using the following parameters:

- (`integer`) `id_cart`: the ID of the cart to validate.
- (`integer`) `id_order_state`: the ID of the order status (Awaiting payment, Payment accepted, Payment error, etc.).
- (`float`) `amount_paid`: the amount that the client actually paid.
- (`string`) `payment_method`: the name of the payment method.

### Securing the module

Here are a few basic rules:

- ALWAYS use HTTPS when communicating with a server.
- Check that the module is activated:

```
if (!$this->active)
    return;
```

- Check that the module is valid for the customer's delivery address.
- NEVER trust foreign data.