addons presta

Quantity Limits

USER MANUAL · Version 1.0.0 · PrestaShop 1.7.8 – 9.x

1. What this module does

Set minimum, maximum and multiple quantities per product, category and customer group. PrestaShop knows one quantity rule, the minimal quantity of a product; this module adds **rules**: a minimum quantity, a maximum quantity and a step ("sold in multiples of 6"), for chosen products, a category with its subcategories, a brand or all products, and for the customer groups you choose. The maximum counts **per cart** or **per customer over the last N days**.

A cart with a quantity that a rule does not allow cannot go to the checkout: the cart shows your message (for example "Water bottle: the maximum quantity is 3 — your cart has 5.") and the "Proceed to checkout" button is disabled, by PrestaShop's own mechanism. Under the product list a button **Set the quantity to 3** corrects the line with one click. The product page shows the limits and gives them to the quantity field. **The module never changes a cart by itself.** An order that is created all the same from a cart that breaks a rule (section 3, "The limit of the block") is flagged for you on its order page in the Back Office.

Works with **PrestaShop 1.7.8, 8.x and 9.x** (PHP 7.2.5 or newer), with the Classic and Hummingbird themes and with multistore.

2. Installation

1. Download the module from **My account** → **Order history** at addonspresta.com.
2. In your Back Office, open **Modules** → **Module Manager** and click **Upload a module**.
3. Drop the `ap_qtylimits_v1.0.0.zip` file and wait for the confirmation.
4. Click **Configure** on the module card.

If you have shell access, you can also unpack the zip into `modules/` and run `php bin/console prestashop:module install ap_qtylimits` from the shop folder.

Nothing changes in your shop until you add a rule. After the installation the module is switched on, but its list of rules is empty ("No rule yet: the module changes nothing. Add a rule."). To set up your first limit:

1. Open **Modules** → **Module Manager** → **Quantity Limits** → **Configure** (in a multistore, first select one shop in the Back Office header).
2. Open the **Rules** tab and click **Add a rule**.

3. Choose what **The rule is for** (for example **Chosen products** and the product IDs), tick the customer groups it is for (or none, for everybody), fill in at least one of **Minimum quantity**, **Sold in multiples of** and **Maximum quantity**, and click **Save**.
4. In the **Test a product** tab, type a product ID and a quantity and click **Check**: you see which rule applies, whether that quantity may go to the checkout and the message the customer would read.
5. Open your shop, look at the product page, put the product in the cart and look at the cart page.

3. How it works

What a rule is for

Every rule has a target, a way of counting, a list of customer groups and up to three limits. **The target** is one of four:

- **Chosen products** — the product IDs you type, up to 200 per rule.
- **A category with its subcategories** — every product of the category and of all its subcategories. **A product counts when any of its categories is inside**, not only its main category.
- **A brand** — every product of that brand.
- **All products** — every product of the shop.

Count decides what one rule counts for a product with combinations:

- **The product as a whole (all its combinations together)** — 2 pieces in size S and 4 in size M are 6 pieces of the product.
- **Each combination on its own** — size S and size M are limited separately, and the messages name the combination ("T-shirt (S)").

Whatever the target, **a rule limits each product on its own**. A rule for the category Drinks with a maximum of 6 allows 6 pieces of every drink, not 6 drinks together. Pieces that a cart rule gives away as a gift are not counted: the customer cannot remove them.

Packs. A PrestaShop pack has its own product ID in the cart and in the orders: a rule for a product does not limit a pack that contains it, and the pieces inside a pack are not counted for that product. A pack is a product of its own: add the pack's ID to the rule (or to a rule of its own) if it must be limited too.

For which customers: groups

Tick the customer groups a rule is for. **None ticked means every group.** The group that counts is the customer's **default group** (the "Default customer group" of the customer's account), the same group PrestaShop uses for prices; other groups of the customer do not count.

- A visitor who is not logged in is in the **Visitor** group.
- **A guest checkout counts as the Guest group**: a customer who orders without an account is in the Visitor group until the checkout creates the guest account, and in the Guest group from then on. A rule ticked only for "Customer" does not apply to a guest checkout.

- For every shopper, leave the list empty or tick **Visitor**, **Guest** and **Customer**.

When a customer logs in, the rules of their group apply right away, on the next page.

The three limits

A rule has up to three limits; at least one is required. All are whole numbers from 1 to 99,999 (a step from 2).

Minimum quantity — the cart must hold at least this many pieces of the product. Example: a minimum of 10 for the brand "Office Pro", for the group Wholesale. A wholesale customer with 6 boxes of pens in the cart reads "Box of pens: the minimum quantity is 10 — your cart has 6." and cannot proceed to the checkout; the button **Set the quantity to 10** raises the line to 10.

Maximum quantity — the cart must not hold more than this many pieces. Two ways of counting:

- **Per cart** — only what is in the cart counts. Example: a maximum of 3 for product 15 (a water bottle on promotion). A cart with 5 shows "Water bottle: the maximum quantity is 3 — your cart has 5." and the button **Set the quantity to 3**.
- **Per customer over a period** — what the customer already ordered in the last N days (2 to 3,650 days) counts together with the cart. Example: a maximum of 2 per customer in 30 days for a scarce product. A customer who ordered 1 piece twelve days ago and puts 2 in the cart reads "Face mask: the maximum is 2 per customer in 30 days — you have already ordered 1 and your cart has 2." and the button **Set the quantity to 1**. Which orders count is explained in "The maximum per customer" below — read it before you use this kind of maximum.

Sold in multiples of (the step) — the quantity must be a multiple of this number. **Only multiples are allowed, and the minimum is rounded up to a multiple.** Example: sold in multiples of 6 with a minimum of 10 and a maximum of 24. The allowed quantities are 12, 18 and 24: the minimum of 10 becomes 12. A cart with 14 reads "Juice pack: sold in multiples of 6 — your cart has 14." and the button **Set the quantity to 12** (a quantity that is not a multiple is always rounded **down**, never above what the customer chose); a cart with 8 reads the minimum message with the minimum 12 and the button sets 12.

When a cart quantity breaks more than one limit, the customer reads one message for the product, in this order: the maximum first, then the minimum, then the multiple. The form refuses limits that would allow nothing ("With these numbers nothing could be bought: the minimum, rounded up to a multiple, is above the maximum.").

Which rule applies when several match

The list of rules is in priority order. **For every product, the first rule from the top that holds the product and is for the customer's group supplies all three limits; the rules below it are not looked at for that product.** Limits of several rules are never combined. A rule that is switched off is ignored. **Put the specific rules above the general ones.**

Example: rule 1 is "Chosen products: 15 — maximum 1 per customer in 30 days", rule 2 is "A category with its subcategories: Drinks — sold in multiples of 6". Product 15 is a drink. For product 15, rule 1 decides: the customer may order 1 piece in 30 days, and the multiples of rule 2 do not

apply to it. Every other drink is sold in multiples of 6. If product 15 must also be sold in multiples of 6, give rule 1 that step too. With the two rules the other way round, rule 2 would hold product 15 first and its maximum would never count.

Test a product shows which rule applies to a product for a group, with its number in the list.

PrestaShop's own minimal quantity

PrestaShop's own **Minimum quantity for sale** of a product (or of a combination) stays in force: PrestaShop checks it itself, with its own message. The module takes the larger of the two minimums, then rounds it up to the step.

Example: a product with PrestaShop's minimal quantity 3. A rule with a minimum of 2 → the minimum is 3. A rule with a minimum of 5 → 5. A rule sold in multiples of 4 without a minimum → 4 (3 rounded up to a multiple of 4). The product page shows that one number.

When a rule counts the product as a whole and the cart holds several combinations of it, PrestaShop checks its own minimal quantity of each line itself, and the module checks the rule's minimum for the sum.

The maximum per customer

Read this before you rely on a maximum per customer. The pieces "already ordered" are read from PrestaShop's orders whenever they are needed — in the cart, on the product page, for the button, at the creation of an order and in **Test a product** — always the same way.

A maximum per customer counts the pieces of every order the customer placed in this shop during the period — paid or still waiting for payment — except orders whose status is Canceled, Payment error or Refunded. Refunded and returned pieces do not count. Orders in your other shops do not count.

In detail: every order of the customer **in this shop**, placed in the last N days (the date of the order, not of its payment), **whatever its payment state** — paid, in preparation, shipped, delivered, and also an order that waits for a bank wire, a cheque, a cash-on-delivery confirmation or a card payment, an order on backorder, and an order with a status of your own or of a payment module. **An order waiting for a bank wire therefore reserves the limit from the moment it is placed.** From each counted order, the pieces that were refunded or returned are taken off.

What frees the limit again:

- the order gets PrestaShop's status **Canceled**, **Payment error** or **Refunded**, from that moment;
- its pieces are refunded or returned on the order page (a partial refund frees the refunded pieces);
- the period ends: N days after the order was placed;
- the order is deleted.

An order that is never paid keeps the customer's pieces reserved until you give it the status Canceled, refund its pieces, or the period ends. If you use a maximum per customer, cancel orders that will not be paid. A second order while the first one still waits for payment is refused at the cart, and when it is created past the block, it is flagged (see "The limit of the block").

Statuses you created yourself, or that a payment module added (for example for a failed, expired or voided payment), are not treated as cancelled: their orders keep counting, and the customer may be unable to order the product again. Give such an order PrestaShop's own Canceled status to free the limit. A status of your own named "cancelled" is no exception: the module knows only PrestaShop's own three statuses above, so an unknown status never lets a customer go above the maximum.

If you move an order back from Canceled, Payment error or Refunded to another status, it counts again, but the orders the customer placed in the meantime are not checked again: look at the customer's other orders of the period first.

Guests and new accounts cannot be recognised: for them a maximum per customer is a maximum per cart. Switch guest checkout off if the limit must hold. Only a logged-in customer has a purchase history. A visitor and a customer who orders as a guest (guest checkout) can buy the maximum again in every new cart, and **a customer who opens a new account starts from zero.** The module does not recognise people by their e-mail address or delivery address. Guest checkout is switched off in **Shop Parameters → Order Settings** ("Enable guest checkout") — then only logged-in customers can order; it still cannot stop a person from opening a second account.

More details:

- In a multistore, only the orders of the shop of the rule count, also when shops share their customers and orders.
- Gift pieces of past orders count: in an order they cannot be told apart from bought ones.
- Nothing is stored: the sum is read from PrestaShop's orders each time it is needed and never copied.

What the customer sees

On the product page. For a product that a rule limits:

- The quantity field gets the limits: its minimum, its step and — for a maximum per cart — its maximum, and it starts at the smallest allowed quantity. The theme's own "+" and "−" buttons move by the step and stop at the limits, and a typed value is corrected to the nearest allowed one (7 becomes 6 with multiples of 6, 99 becomes the maximum). When the cart already holds the product, the field offers what may still be added.
- The theme prints its own sentence "The minimum purchase order quantity for the product is 12." with the module's minimum.
- Under the **Add to cart** button, a short line gives the other limits, for example "Sold in multiples of 6 · Maximum quantity: 24" or "Maximum quantity: 2 per customer in 30 days". **A logged-in customer who already ordered pieces also reads "— you have already ordered 1"**, and "You have reached the maximum quantity for this product." when nothing more may be added. You can switch the line off in the settings; the field keeps its limits.
- A product with **volume discounts** whose quantity had to be raised refreshes once after the page has loaded, so the price shown is the price of the quantity in the field.
- Nothing is shown in quick view and in product lists, and nothing for a product that is not available for order.

The script only helps. Without JavaScript, or with a theme that has its own quantity control, the field may not follow the step or the maximum — **the limits are still enforced in the cart**, which judges every quantity that reaches it.

After "Add to cart". The module does not refuse an "Add to cart": the product goes into the cart. When its quantity is not allowed, the theme's pop-up shows the message of the product, the same text as in the cart. On phones (screens narrower than 992 px) the Hummingbird theme hides the part of its pop-up that holds the message; there the customer reads it on the cart page.

In the cart. A cart with a quantity that is not allowed shows one message per product in the warning box of your theme, above the disabled **Proceed to checkout** button (at most five messages, then "More products with a quantity that is not allowed: 2."). This is PrestaShop's own mechanism, the one it uses for its own minimum: the messages and the button follow every quantity change in the cart without reloading the page.

Under the product list the block "Set the allowed quantity with one click:" lists each product that one click can correct, with a button **Set the quantity to N**. The shop computes N itself — the nearest allowed quantity — and the cart page then says "The quantity of Water bottle was set to 3.". When PrestaShop refuses the change (not enough stock, for example), the customer reads "The quantity could not be changed. Change it in the cart." There is no button:

- for a product counted as a whole whose pieces are spread over several lines (several combinations) — the message says what is wrong and the theme's own quantity buttons change the cart;
- when nothing can be bought any more (a maximum per customer already reached) — the customer has to remove the product;
- for a line split over several delivery addresses. When PrestaShop shows such a line as one row — it does so while product customization is switched on, PrestaShop's default — a button can appear: a click on it changes nothing, and the customer reads "The quantity could not be changed. Change it in the cart."

The cart is never changed behind the customer's back: only the customer's click on a button changes a quantity.

At the checkout. Every address of the checkout sends a blocked cart back to the cart, where the messages are. On PrestaShop 1.7.8 a customer who is logged in and opens the checkout address with a blocked cart lands on the home page instead of the cart — exactly as with PrestaShop's own minimum on that version.

The messages. Each kind of message has a pattern per language. A field left empty uses the default text, translated into the customer's language:

- Below the minimum: "{product}: the minimum quantity is {limit} — your cart has {quantity}."
- Above the maximum per cart: "{product}: the maximum quantity is {limit} — your cart has {quantity}."
- Above the maximum per customer: "{product}: the maximum is {limit} per customer in {days} days — you have already ordered {bought} and your cart has {quantity}."
- Not a multiple: "{product}: sold in multiples of {limit} — your cart has {quantity}."

The placeholders are replaced in the cart: {product} is the name of the product (with the combination when the rule counts each combination on its own), {limit} the limit that is broken (the minimum rounded up to the step, the maximum or the step), {quantity} the quantity in the cart, {days} the period of the rule, {bought} the pieces the customer already ordered in the period. Numbers are written in the customer's number format.

Messages are **plain text**: no HTML, the characters `<` and `>` are refused, up to 255 characters per field, and a placeholder other than the five above is refused when you save. Whatever is stored, **a message that would come out empty falls back to the default text**: a cart that breaks a rule always has a message.

Other modules. The text of the cart's warning box is shared: PrestaShop's own minimum order amount and other modules that block the checkout the same way (for example Order Amount Limits) write into it too. The module **adds its messages after the text that is already there and never removes it**, and when your rules accept the cart it does not touch the box at all — the other module's message and block stay. One box can therefore show the message of another module followed by the messages of this one.

The limit of the block

Read this before you rely on a minimum, a maximum or a step. The block is **PrestaShop's own checkout block**: it works where PrestaShop shows the cart and the checkout, not at the moment the order is created.

The module closes the checkout for a cart that breaks a rule, but it cannot refuse an order at the moment the order is created — PrestaShop's own minimum purchase amount has the same limit. An order can still go through, for example when the customer changes the cart in a second browser tab after opening the payment step, or when a payment module's confirmation address is opened directly. The same goes for a one-page-checkout module or an express-payment button that creates orders outside PrestaShop's checkout: such a module never shows PrestaShop the cart before the order. The second-tab case needs no technical skill. No module can refuse an order at that moment without changing PrestaShop's core, which this module never does.

What the module does about it: every order created in the storefront or through the webservice is checked again at the moment it is created, with the same rules as the cart page, for the customer of the order (the customer's default group, and for a maximum per customer the orders that customer placed before, as described above — an earlier order that still waits for payment counts).

Such an order is created normally and may already be paid. The module does not cancel it, does not change its status and tells the customer nothing. It shows a warning at the top of the order page in the Back Office — each product with the limit and the quantity of the cart — and writes one warning in Advanced Parameters → Logs.

- The warning on the order page reads "Quantity Limits: this order breaks a rule", then one line per product, for example "Water bottle: maximum quantity 3, the cart had 5." or "Face mask: maximum 2 per customer in 30 days, the customer had already ordered 2 and the cart had 2.".

- **The warning on the order page is shown only to employees whose profile has the View permission for Quantity Limits (Advanced Parameters > Team > Permissions > Modules). Grant it to every profile that handles or ships orders.** PrestaShop gives that permission at the installation only to the profiles that may manage modules: an employee of a packing or customer-service profile who may open orders but not the module sees **no warning** on a flagged order. In **Permissions**, select the profile and tick **View** in the row of Quantity Limits.
- The line in the log is a warning attached to the order, written whoever looks at the order: "Quantity Limits: order 123 was created with a cart that breaks the rules of the module (products concerned: 1); see the warning on the order page."
- No new status, no history line, no message, no e-mail, nothing on the customer's confirmation page. **You decide**: contact the customer, cancel the order, or accept it.

Check that warning before you ship. It is shown only on the order page: not in the orders list, not by e-mail and not to systems that read orders through the webservice or an export. If orders are fulfilled automatically when they are paid (downloadable products, an ERP, dropshipping or shipping-label exports), such an order can be delivered before anyone sees the warning.

Orders created in the Back Office are neither blocked nor flagged; orders created through the webservice are not blocked, but they are flagged. This includes an order a payment module creates from its Back Office page while an employee is logged in: it is not flagged either. The employee chose what is in a Back Office order; the rules are for customers.

Not detected: a guest or a customer with a second account who orders the maximum again (a maximum per customer counts only the orders of the logged-in account); a pack that contains a limited product; two orders of the same customer created at the very same instant.

An order placed while the module was disabled is never flagged later; editing the order or the rules does not change or remove a warning; uninstalling the module deletes the warnings. The warning is a fact of the moment the order was created, with the rules and the cart as they were then.

- When one cart is split into several orders (several carriers or warehouses), the cart is judged once, as a whole, and every order of it gets the warning. A warning lists at most 10 products.
- If the check of a new order fails, the order is not flagged and one line at most per hour is written to the log (see "Failures" below).

Full-page caches

If you use a full-page cache module, do not let it store product pages of logged-in customers: the page shows the customer's own number of pieces already ordered. For a logged-in customer the product page carries numbers of that customer only: "you have already ordered N" in the line, and the limits of the quantity field. PrestaShop's own caches never keep that part of the page. Switch caching off for logged-in customers, or leave the hook `displayProductAdditionalInfo` out of what the cache module stores; otherwise one customer could read another customer's number (a number only — no name and no order).

Performance

The module judges the cart once per page and reuses that answer for everything else on the page. Measured with 50 active rules, a cart of 30 lines and a customer with 200 orders, the decision took about 6 ms with three small queries, and almost nothing when it was asked again on the same page; whole pages were as fast with the module switched on as switched off, within the noise of the measurement. A shop without an active rule runs no query at all. A new order costs the same decision once more.

Personal data, connections, e-mails

The module stores your rules and the flags of orders in two tables of its own; a flag holds the order's ID, the product's ID, the limit and the quantities — **no personal data beyond the order ID of a flagged order**. **The purchase history of a customer is read from PrestaShop's orders when it is needed and never copied.** The module sets no cookie and keeps nothing in the visitor's browser. It makes no outgoing connection — no licence check, no statistics, nothing to addonspresta.com or any other server. It has no cron job and sends no e-mails.

Failures. If a part of the module fails on a page, the page is shown normally and the cart is **not** blocked by the module (PrestaShop's own minimal quantity stays at work), and one line at most per hour is written to **Advanced Parameters → Logs**, starting with "Quantity Limits:".

4. Features and settings

Open **Modules → Module Manager → Quantity Limits → Configure**. The **Help** box at the top links this manual and the module page at addonspresta.com in the language of your Back Office, gives our support address and shows the installed version. Below it are three tabs: **Settings**, **Rules** and **Test a product**.

Settings

Saved for the shop or shop group selected in the Back Office header with **Save**; "Settings saved." confirms it. When a value is wrong, nothing is saved and the form shows the message above your values. Above the form the page warns you when the shop is in catalogue mode ("The shop is in catalogue mode: nothing can be ordered, so the module checks nothing.").

Enabled

On after the installation. "Switched off, nothing is checked, shown or blocked. The rules stay." New orders are not flagged while it is off.

Show the limits on the product page

On after the installation. "A short line under the Add to cart button, for example "Sold in multiples of 6 · Maximum quantity: 24". The quantity field follows the limits also when the line is off."

Messages in the cart

"A cart with a quantity that is not allowed shows one message per product above the disabled "Proceed to checkout" button, and the pop-up after "Add to cart" shows the message of the product just added. Leave a field empty to use the default text, which follows the customer's language." One field per language for each kind:

- **Below the minimum**
- **Above the maximum per cart**
- **Above the maximum per customer**
- **Not a multiple**

Under each field, "Default: …" shows the default text in your Back Office language. The placeholders and the rules for the text are in section 3, "What the customer sees". The messages refused when you save:

- "A message can have up to 255 characters."
- "The messages are plain text: remove the characters < and >."
- "A message uses an unknown placeholder. The placeholders are: {product} {limit} {quantity} {days} {bought}"

Rules

Rules belong to one shop: the tab works when one shop is selected in the Back Office header ("Rules belong to one shop. Select one of your shops in the header to manage its rules." otherwise). A shop can have up to **100 rules**.

The list shows the rules in priority order with the columns **#**, **For** (the target), **Counted**, **Customer groups**, **Limits** (in words, for example "minimum 10 · multiples of 6 · maximum 24 per cart") and **Enabled**. "All groups" stands for an empty list; long lists end with "and 3 more". A product, category, brand or group the shop no longer has is marked "(1 not in this shop)" or "(not in this shop any more)". The buttons of each row:

- **Move up** and **Move down** (the arrows) — change the priority ("Order of the rules changed.");
- **Switch off** / **Switch on** — a rule that is switched off stays in the list, greyed, and is ignored ("Rule updated.");
- **Edit** — opens the form;
- **Delete** — after the question "Delete this rule?" ("Rule deleted.").

Add a rule opens an empty form; it is hidden when the shop has 100 rules.

The form ("Add a rule" or "Edit the rule"):

- **Enabled** — the rule counts only when ticked.
- **The rule is for** — **Chosen products**, **A category with its subcategories**, **A brand** or **All products**. "For every product the first rule from the top of the list that holds the product and is for the customer's group supplies all the limits; the rules below it are not looked at for that product. The limits apply to each product on its own, never to several products together."

- **Product IDs** — only for **Chosen products**: "The IDs of the products, separated by commas or spaces, for example 12, 15, 48. Up to 200 products of this shop." You find the ID in **Catalog → Products**.
- **Category** — only for **A category with its subcategories**: a list of the shop's categories, subcategories indented under their parent.
- **Brand** — only for **A brand**: the brands of the shop.
- **Count** — **The product as a whole (all its combinations together)** or **Each combination on its own**.
- **Customer groups** — tick boxes for the groups of the shop. "Nothing ticked: the rule is for every group. The customer's default group counts; a visitor who is not logged in is in the Visitor group, and a customer who orders without an account (guest checkout) is in the Guest group. For every shopper leave the list empty or tick Visitor, Guest and Customer."
- **Minimum quantity** — "Empty: no minimum. PrestaShop's own minimal quantity of the product stays in force; the larger of the two applies."
- **Sold in multiples of** — "Empty: any quantity. With 6 the customer can buy 6, 12, 18 ...; a minimum that is not a multiple is rounded up to one (minimum 10 becomes 12)."
- **Maximum quantity** — "Empty: no maximum."
- **The maximum counts** — **Per cart** or **Per customer over a period**. "Per customer: the pieces in the customer's orders of this shop placed in the period count together with the cart. An order counts from the moment it is placed, also while it waits for payment, until it gets the status Canceled, Refunded or Payment error; refunded and returned pieces of the other orders do not count. Only logged-in customers can be recognised: for visitors and guests the number is a maximum per cart."
- **Period in days** — only for **Per customer over a period**: from 2 to 3,650 days.

Fields that do not belong to the chosen target or maximum are ignored when you save.

Nothing is dropped behind your back. Typed product IDs that are not products of this shop are refused with their numbers ("These IDs are not products of this shop — remove or correct them: 7, 9999"), never removed silently. A group the rule has but the shop no longer has (it was deleted) stays in the list, ticked, as "#12 (not in this shop any more)"; you decide whether it goes: untick it and save. **A group posted in the form that is not a group of this shop refuses the save** ("The list of customer groups holds a group that is not a group of this shop. Open the rule again and choose the groups from the list.").

An empty list means everybody, so it is never emptied by accident. When a save would leave the groups of a rule empty although the rule had a list before, the save is refused ("No customer group would be left in the list, so the rule would apply to every group. Tick at least one group, or tick the confirmation under the list to apply the rule to every group.") and the form shows the box "Yes, the list left empty is meant: apply this rule to every customer group." Tick it and save again to make the rule apply to everybody. A new rule with nothing ticked needs no confirmation.

Click **Save** ("Rule saved.") or **Cancel**. Other messages of the form:

- "Choose what the rule is for."

- "Enter the product IDs as whole numbers separated by commas or spaces, for example 12, 15, 48."
- "Enter at least one product ID."
- "A rule can list up to 200 products. Use a category or a brand for more."
- "Choose a category of this shop for the rule." / "Choose a brand of this shop for the rule."
- "Enter the minimum and the maximum as whole numbers from 1 to 99999, or leave them empty."
- "Enter "Sold in multiples of" as a whole number from 2 to 99999, or leave it empty."
- "Fill in at least one limit: the minimum, the multiple or the maximum."
- "With these numbers nothing could be bought: the minimum, rounded up to a multiple, is above the maximum."
- "Enter the period as a whole number of days from 2 to 3650."
- "A shop can have up to 100 rules."

Test a product

"Check which rule a product gets and whether a quantity of it may go to the checkout. Nothing is changed." The test works in the context of one shop ("Select one of your shops in the header to test a product." otherwise). Fill in:

- **Product ID** and, for a rule that counts each combination on its own, **Combination ID (optional)**;
- **Quantity in the cart**;
- **Customer group**;
- **Customer ID (optional)** — "With a customer, the customer's default group is used instead of the group above, and a maximum per customer counts the pieces the customer ordered in the period. Only the number of pieces is shown."

Click **Check**. The test runs the shop's own decision and answers: the product and the group, which rule applies ("Rule 2 applies — ...") or "No rule of the module applies to this product for this group.", PrestaShop's own minimal quantity, the quantities allowed in the cart ("Allowed in the cart: from 12 to 24, in multiples of 6."), for a customer the pieces ordered in the period, then "A cart with 14 may go to the checkout." or "A cart with 14 is blocked. The customer reads:" followed by the message exactly as the customer would read it, and the quantity the cart button would set ("The button in the cart sets the quantity to 12." or "No button: the customer has to remove the product.").

The Customer ID field is shown only to employees whose profile may view Customers; the others read "A maximum per customer can be tested with a customer's purchase history only by employees who may view customers." The customer must be an account of this shop. The test shows a number of pieces only — never a name, an e-mail address or an order. A guest account is treated as a visitor ("This is a guest account: guests are not recognised, a maximum per customer is checked as a maximum per cart."), and without a customer a maximum per customer is checked as a maximum per cart.

The test also says when the module is switched off in the settings or the shop is in catalogue mode, because nothing is checked in the shop then.

5. Multistore

Rules belong to one shop: each shop has its own list, and you edit it with that shop selected in the Back Office header. **Settings** are saved for the shop or shop group selected in the header; values saved with **All shops** apply to every shop that has no value of its own. In the All shops or a group context only an employee who manages every shop of that context can save ("Select one of your shops in the header to change the settings: in the All shops or group context they also apply to shops you do not manage."). **Test a product** works in one shop at a time.

Each storefront uses the rules and settings of its own shop, and each new order is checked with the rules and settings of the shop it was placed in. A maximum per customer counts only the orders of that shop. One licence covers every shop of the installation.

6. Uninstall

Module Manager → Quantity Limits → Uninstall deletes all the rules, all the settings and all the flags of orders, in every shop. The two tables of the module are dropped: the warnings disappear from the order pages, and the rules cannot be restored. **Reset** does the same and then installs the module again with the default settings — **also without any rule**. Write your rules down before you uninstall or reset. Your orders, carts and products are not touched.

Disable keeps the rules, the settings and the flags; while the module is disabled nothing is limited, nothing is blocked, new orders are not checked, and PrestaShop's own minimal quantity applies alone. Switching off **Enabled** in the settings does the same for one shop.

7. FAQ and troubleshooting

A limit is not applied. Type the product ID, the quantity and the group in **Test a product**: it says which rule applies and whether the quantity may go to the checkout. Then check:

1. The module is enabled in the Module Manager and **Enabled** is on in the settings of this shop.
2. The rule is switched on, and the shop is not in catalogue mode.
3. The customer's **default** group is in the rule's groups (a guest checkout is in the Guest group, a visitor who is not logged in in the Visitor group).
4. The product is in the rule's target: its ID in the list, any of its categories inside the rule's category, or the rule's brand.
5. No rule above it in the list holds the product first: the first matching rule supplies all the limits.

If it still does not work, look in **Advanced Parameters → Logs** for an entry starting with "Quantity Limits:".

The customer cannot buy the product although they have not bought it. An order that waits for payment (a bank wire, a cheque) or that was cancelled with a status of your own reserves the limit. Open the customer's orders: cancel an unpaid order with PrestaShop's own **Canceled** status, or

refund its pieces, and the limit is free again. **Test a product** with the Customer ID shows the pieces that count.

The quantity field ignores the step or the maximum. Your theme has its own quantity control that the module's script does not know. The cart still enforces every limit: a wrong quantity is refused at the cart, with the message and the button **Set the quantity to N**.

The price does not match the quantity in the field. When the module raises the starting quantity of a product with volume discounts, the page refreshes once to show the price of that quantity. Without JavaScript this refresh does not happen; the cart computes the right price.

An order above the maximum (or below the minimum) exists. Look at the top of the order page in the Back Office: a flagged order shows the warning "Quantity Limits: this order breaks a rule" with the products and the numbers, and **Advanced Parameters → Logs** has a line for it. The order was placed past the checkout block (section 3, "The limit of the block"), for example with a second browser tab, or a second order was created while the first one waited for payment. The module changed nothing in the order: contact the customer, cancel the order or accept it. If an employee does not see the warning, give their profile the module's **View** permission. An order created by an employee in the Back Office is never flagged.

The message appears next to the message of another module. The cart's warning box is shared: PrestaShop's own minimum order amount or another module (for example Order Amount Limits) and this module each add their own text (section 3, "Other modules"). The checkout opens when none of them blocks it.

A one-page checkout or an express-payment button ignores the limits. Such a module creates the order without showing PrestaShop the cart, so the block cannot stop it (section 3, "The limit of the block"). Orders placed through it are still checked when they are created and flagged when they break a rule.

A logged-in customer on PrestaShop 1.7.8 lands on the home page. With a blocked cart, PrestaShop 1.7.8 sends a logged-in customer who opens the checkout to the home page, also for its own minimum. The messages are in the cart.

There is no "Set the quantity" button for a product. The product is counted as a whole over several lines, nothing can be bought any more, or the line is split over several delivery addresses (section 3, "What the customer sees"). The message says what is wrong; the customer changes the cart with the theme's own buttons.

"The page has expired. Reload it and save again." The page was open for a long time, or you logged in again in another tab. Reload the page and save again. In the shop, "The page has expired. Try again." after a click on **Set the quantity to N** means the same: the customer clicks again.

A text of the module is still in English in my shop or my Back Office. Open **International → Translations → Installed modules translations**, select the module and your language, and check the translation. Your own message patterns are set per language in **Settings → Messages in the cart**.

8. Updates and support

Updates are free for life: download new versions from **My account → Order history** at addonspresta.com and upload them like a new module. Your rules and settings are kept.

Support is included for 12 months from your order: write to **mail@addonspresta.com** or use the contact form at addonspresta.com (Monday to Friday, 10:00–16:00; we usually answer within 2 business days). Please include your PrestaShop version, the module version and the steps that lead to the problem.