GURUCODERS · PRESTASHOP MODULE

Omnibus: Lowest Price in 30 Days

User manual · version 1.0.0

Contents	
1. What the module does	The rule, and what this adds to your shop
2. Requirements	Versions, themes, database, what it costs a page
3. Installation	Three steps, and where it appears
4. The first run, in order	Rebuild, recompute, read, then switch on
5. What your customers see	Three real cases, including the uncomfortable one
6. The back office, screen by screen	Dashboard, Price history, Jobs and evidence, Settings
7. Keeping it up to date	On its own, or on a schedule
8. Turning it off and uninstalling	The switch, and what uninstalling destroys
9. When something looks wrong	Seven symptoms and what to check
10. Help	Where to write, and what to send

1. What the module does

The module adds one line next to the price on your product pages: the lowest price your shop charged in the last 30 days. That line is what Directive (EU) 2019/2161 - the "Omnibus" directive, in Romania OUG 58/2022 and HG 947/2000 art. 4^1 - requires whenever you announce a price reduction, and the percentage you advertise has to be computed against that figure.

It records the history from the day you install it, **and rebuilds the 30 days that already passed** from data your shop already holds: your past orders, dated specific prices and catalog price rules. So the shop is compliant on the day you install, not thirty days later. Module technical name: `gcomnibus`.

In the back office it shows, product by product and customer group by customer group, which of your reductions the law supports and which it does not, why any single figure is what it is, and it exports the evidence as a CSV you can attach to an answer.

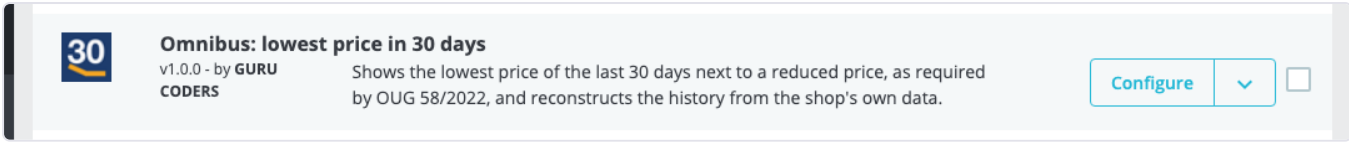
> **What the module is not.** It computes a reference price from evidence it can show you, and states that figure on the page. It does not change your prices, your badges or your struck-through prices, and it is not legal advice. The wording on your storefront and the decision to announce a reduction remain yours.

2. Requirements

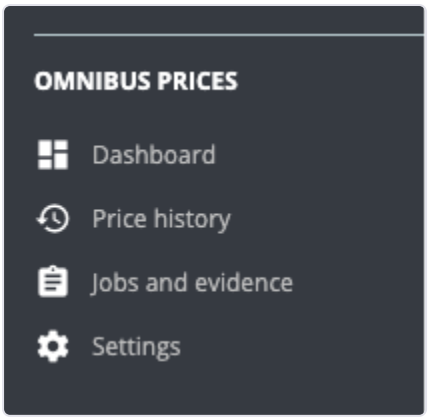
Item	Value
PrestaShop	1.6.0 to 9.x (declared range 1.6.0.0 - 9.99.99). Measured on 1.6.1.24, 1.7.8.11, 8.1.7 and 9.2.0, each on a real shop of that version.
PHP	5.6 or newer.
Themes	Any theme. The module finds the place for the line by itself; if your theme is an unusual one, there is a setting that tells it where to put it (6.4).
Your shop's data	Untouched. The module keeps its own records alongside, and removes them if you uninstall it. It never edits your products, your prices or your orders.
Languages	Back office and storefront line in Romanian and English. The line follows the language and the currency the shop is rendering.
Multistore	Yes, per shop.
Keeping it current	Nothing to set up. It can work on its own, or on a schedule if you prefer - section 7.
Speed	It does not slow your shop down. The figures are worked out in the background and stored, so a customer's page only reads the answer.

3. Installation

1. In the back office, go to **Modules > Module Manager**, click **Upload a module** and select the `gcomnibus` archive.
2. The module appears in the list as **Omnibus: lowest price in 30 days**.
3. A section of its own, **Omnibus prices**, appears in the back-office menu with four screens. On PrestaShop 1.6 the same section sits at the top level of that menu.



How it looks once installed.



And where to find it afterwards.

> **The storefront line installs switched OFF, on purpose.** A module installed five minutes ago has no history, and a reference price computed from nothing is a legal claim with no evidence behind it. Section 4 is the order in which to earn the right to switch it on.

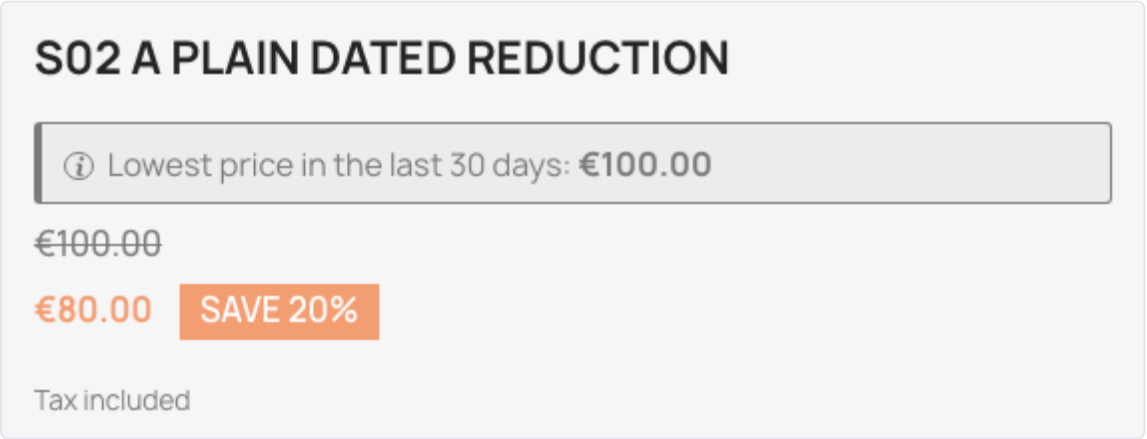
4. The first run, in order

Each step is only believable once the one before it has been done. All three buttons are on **Omnibus prices > Jobs and evidence**.

1. **Rebuild the past window.** The job to run once, at install. It reconstructs the last 30 days from your orders, your dated reductions and your catalog price rules. It is re-runnable and resumable and never writes to a core table. On a large catalogue it reports *partial* - click again and it continues where it stopped.
2. **Recompute the decisions.** This turns the history into one decision per product and per customer group.
3. **Read the dashboard before you believe anything** - section 6.1 explains what you are looking at.

Then, and only then, **switch the line on**: Settings > Output > *Stop all front-office output* > No > Save.

5. What your customers see



The ordinary case. The badge says 20% off, 100.00 struck through, 80.00 charged - and the 30-day minimum really is 100.00. Badge and module agree.

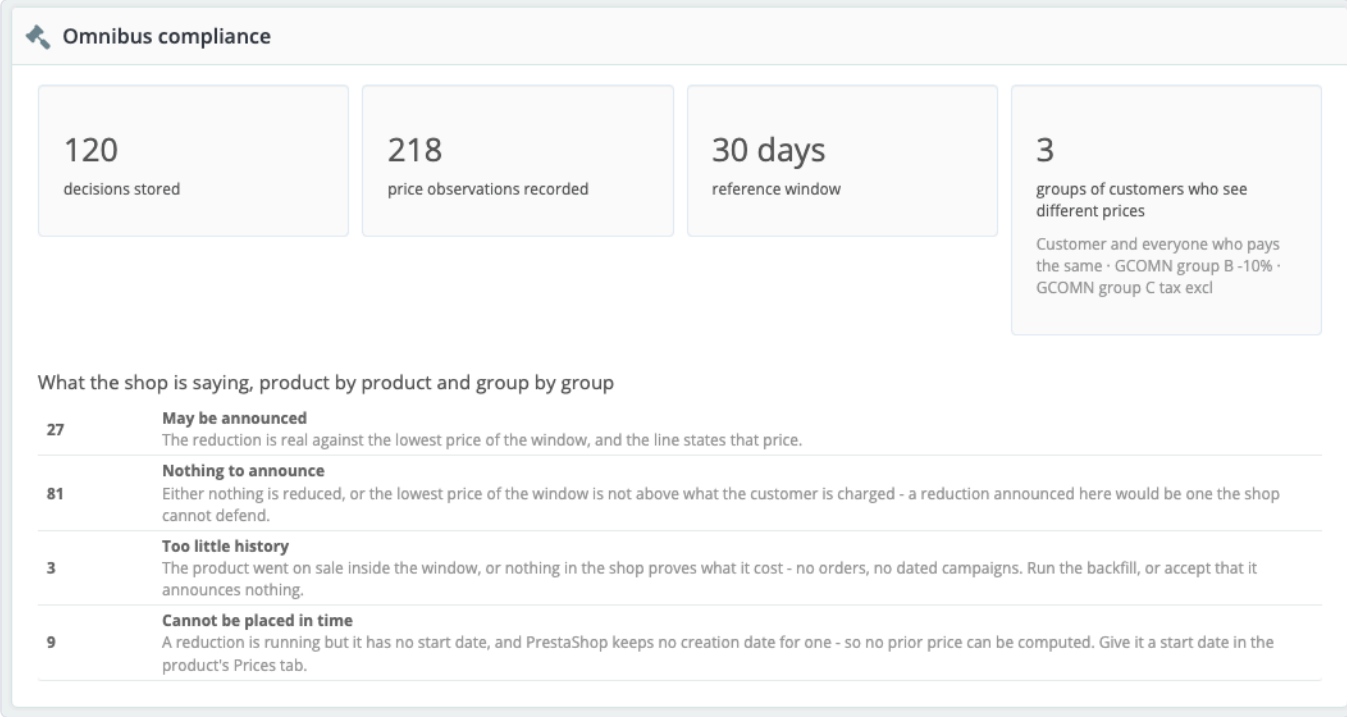


The same shop, another product. The badge says 30% off 120.00, so 84.00 is charged - but the lowest price of the last 30 days was 80.00, which is below 84.00. There is no reduction: the price was raised shortly before the sale. The line states the figure that disproves the badge, in front of the customer. This is the case the article exists for.

A product with no reduction at all gets **no line**. A module that prints something on every page is noise.

6. The back office, screen by screen

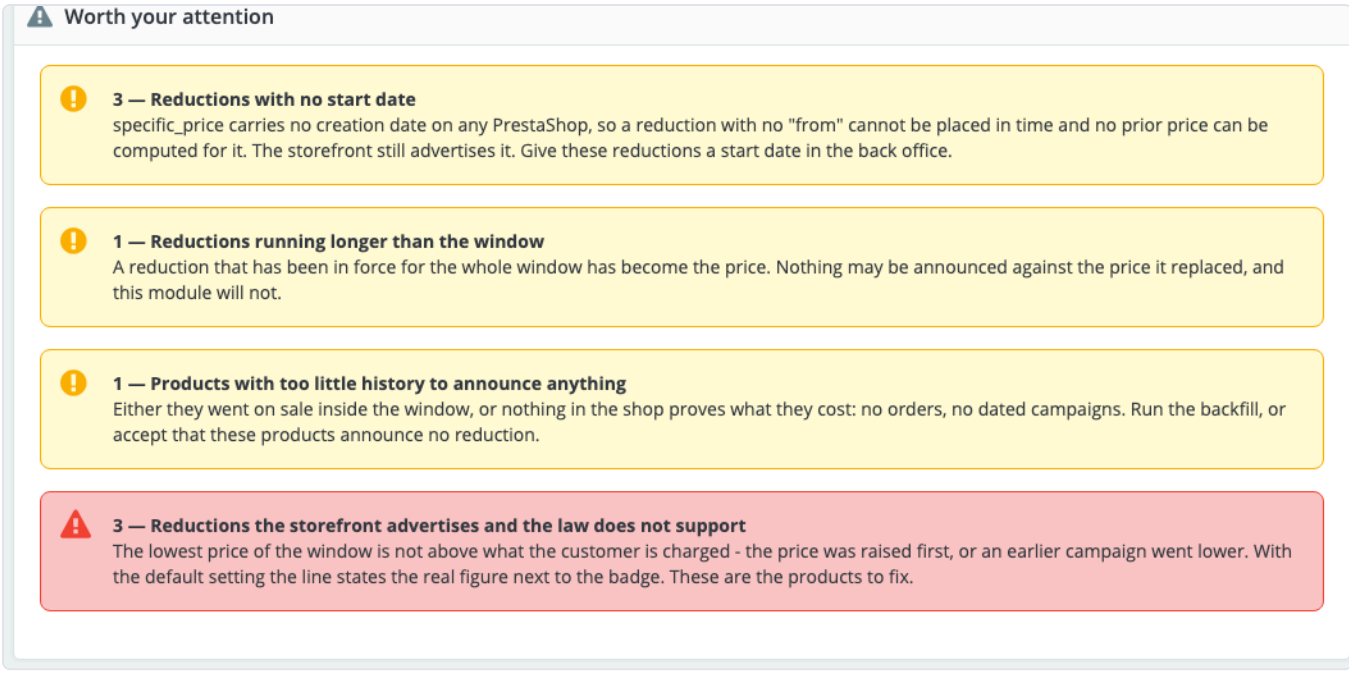
6.1 Dashboard



The four figures at the top are the size of the evidence; the breakdown underneath is what the module would let you claim.

Every product appears once per customer group, because a shop with group prices has more than one history and more than one lawful answer. Each one carries one of five statuses:

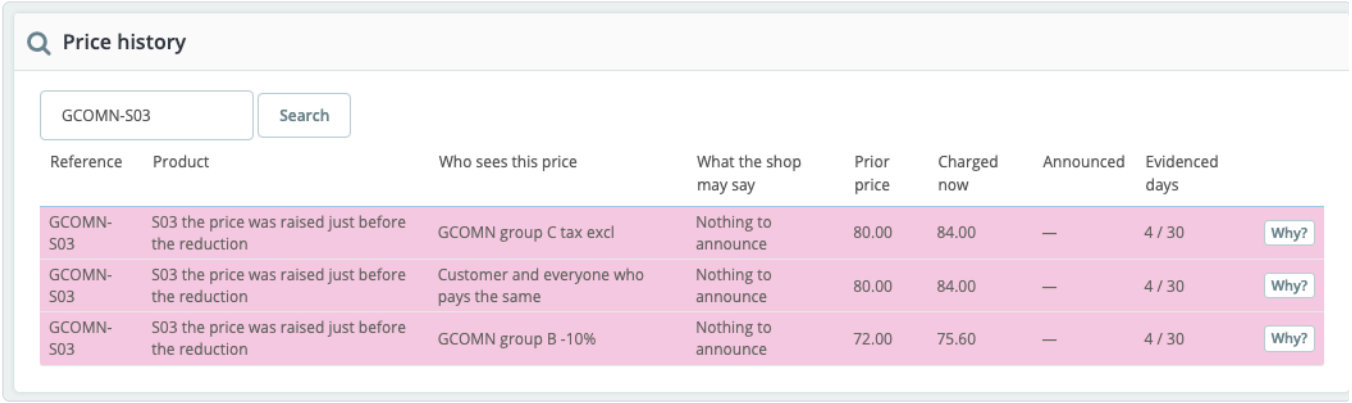
Status	What it means
May be announced	The reduction is real against the lowest price of the window, and the line states that price.
Nothing to announce	Either nothing is reduced, or the lowest price of the window is not above what the customer is charged - a reduction announced here would be one the shop cannot defend.
Cannot be placed in time	A reduction is running but it has no start date, and PrestaShop keeps no creation date for one - so no prior price can be computed. Give it a start date in the product's Prices tab.
Too little history	The product went on sale inside the window, or nothing in the shop proves what it cost - no orders, no dated campaigns. Run the backfill, or accept that it announces nothing.
On sale for less than the window	There is no 30-day price to speak of, so the lowest price since it went on sale is announced instead, with different wording.



The red one is the finding worth having: reductions your storefront is advertising that the law does not support. Those are the products to fix - before the line goes live rather than after.

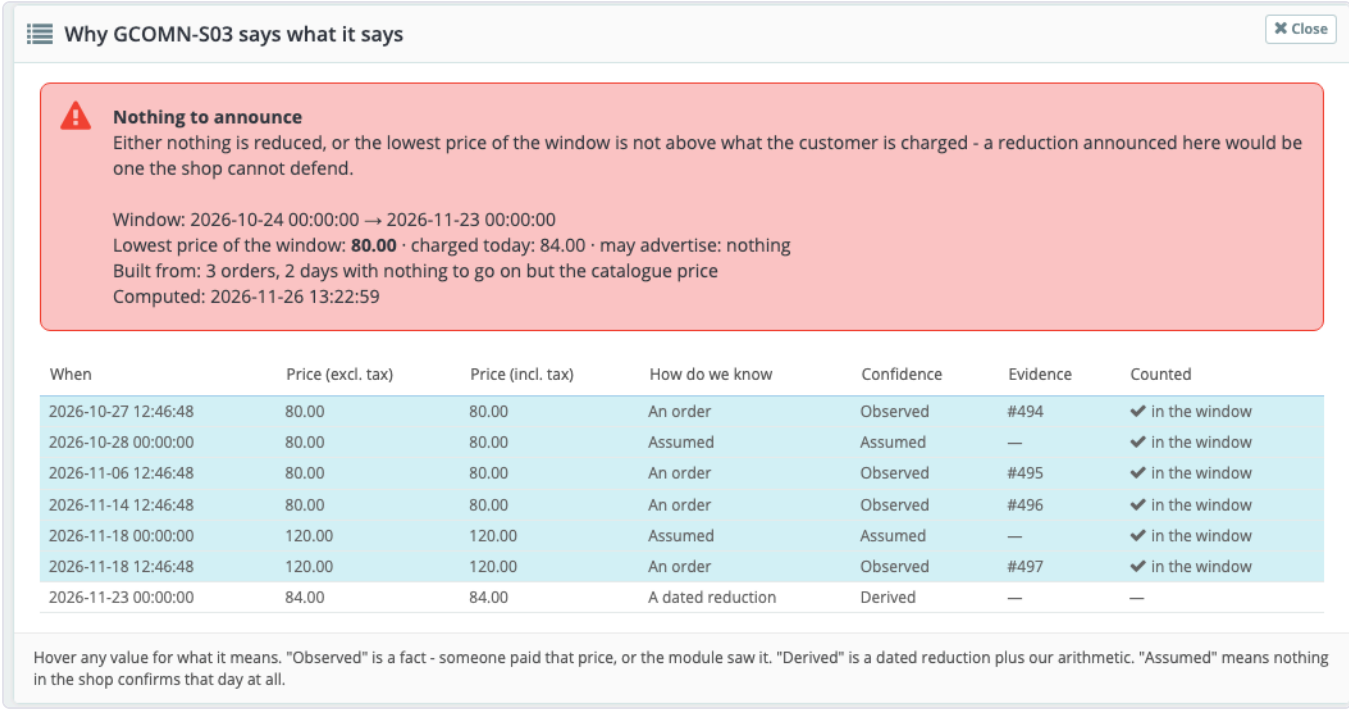
6.2 Price history

Search a product reference. Each row shows what the shop may say, the prior price, what is charged now and how many days of the window are evidenced.



One row per product and per customer group.

Press **Why** on any of them.



The panel takes the colour of the verdict and shows its work: the window, the lowest price in it, what is charged today, what may be advertised, what it was built from, and when it was computed. Below it, every observation behind that answer.

Each observation says **how we know** it:

Source	Meaning
An order	Someone paid this price on this day. The strongest evidence there is, and the only kind that survives a campaign being deleted.
Recorded by this module	A day this module watched itself.
A dated reduction	A specific price with a start date.
A catalog price rule	A rule applying to a set of products.
Assumed	Nothing proves this day; the catalogue price was used.
Entered by hand	Someone typed it.

Next to it, **confidence** - *Observed*, *Derived* or *Assumed* - and the id of the order, specific price or rule that proves it. Hover any value for a sentence explaining it. **Close**, top right, leaves you on the list you searched.

6.3 Jobs and evidence

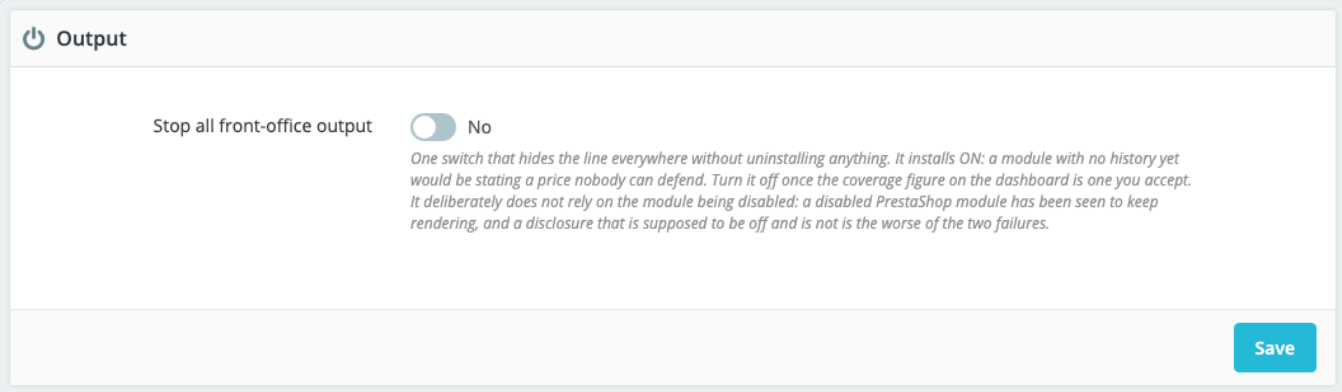
Button	What it does
Rebuild the past window	Reconstructs history from orders, dated reductions and catalog rules. The one to run at install.
Record today	Takes today's snapshot now instead of waiting for a page view.
Recompute the decisions	Turns the history into decisions. Run it after changing a setting.
Catch up new products	The same rebuild, restricted to products with no history at all - the ones added since the module went in. Runs by itself once a day.
Forget what is too old	Clears out records too old to matter, keeping whatever is still needed to read the recent months. Runs by itself once a day.

Download CSV exports every observation and every decision with its source, its confidence and the id of the thing that proves it. This is the file to attach to an answer - for ANPC, or for a customer asking why their discount shrank.

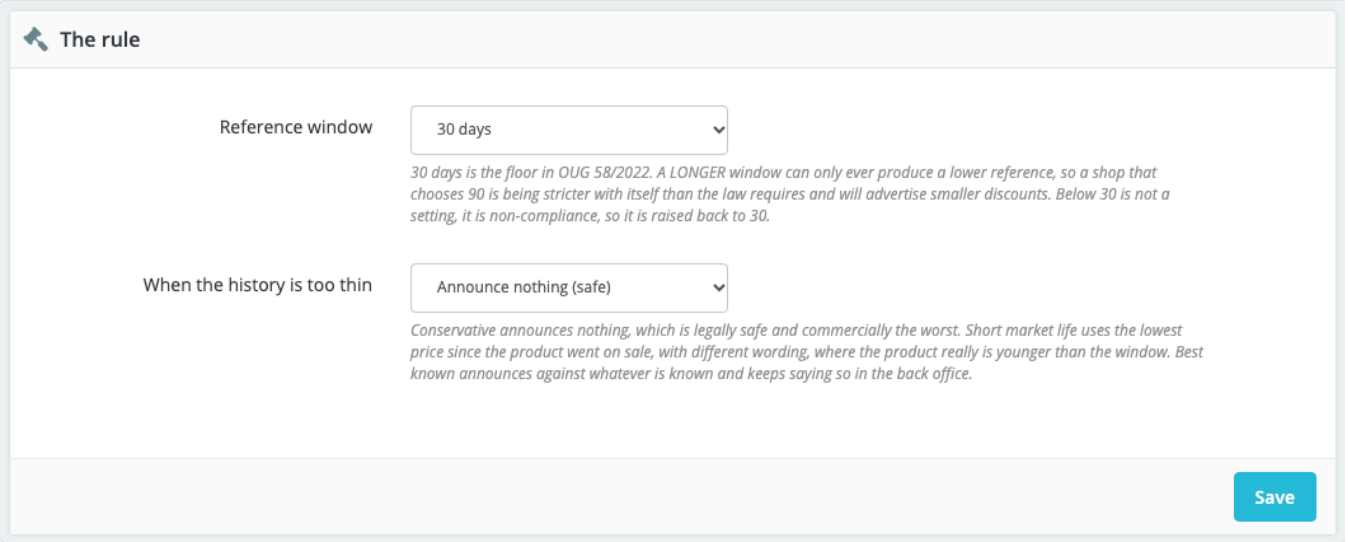
A click is still a web request, so each job takes a bounded chunk and reports *partial* if it did not finish. Click again to continue.

6.4 Settings

Ten settings in four panels. Every one has a sentence underneath explaining what the choice costs; that sentence is the point of the screen. Two of the four panels are below the fold, so scroll.



Output	Details
Stop all front-office output	One switch that hides the line everywhere without uninstalling anything. **On at install.** Set it to No once the coverage on the dashboard is one you accept. It is a switch of its own rather than relying on you disabling the module, because a disabled PrestaShop module has been seen to keep rendering.



The rule	Details
Reference window	30 to 90 days; 30 is the floor in OUG 58/2022. A longer window can only ever produce a lower reference, so choosing 90 is being stricter with yourself than the law requires and will advertise smaller discounts. Below 30 is not a setting, it is non-compliance, and is raised back to 30.
When the history is too thin	**Announce nothing (safe)** - legally safe, commercially the worst. **Lowest price since the product went on sale** - different wording, for a product that really is younger than the window. **Announce against the best known price** - announces against whatever is known and keeps saying so in the back office.

The line	Details
When to show it	**Beside any announced reduction** is the reading of art. 6a: the prior price is required whenever a reduction is announced, and the cases where the announcement is wrong are exactly the ones where a customer needs the figure. The quieter mode shows it only where the module agrees the reduction may be announced, and leaves an overstated badge standing alone.
How loud	**Boxed, with the price in bold** - the default, because a disclosure nobody notices is not a disclosure; it still borrows your theme's colours. Or **One quiet line of text**.
Wording	Empty for the built-in wording. `%1$d` is the window in days and `%2$s` the price, for example `Cel mai mic pret din ultimele %1$d de zile: %2$s`. The words are your legal exposure, which is why they are a setting.
CSS selector for themes that offer no hook	The setting is named for what it takes, but you probably never need it: it is only for a theme that gives the module no place for the line. Leave it empty first - the module tries the usual places by itself. If the line still does not appear, your developer can put the exact spot here.

These last three are the technical corner of the screen. **You will probably never need to touch them**; they are here so nobody has to edit anything by hand to change them.

Engine	Details
Products per chunk	How much work one heartbeat request does. It rides on a customer's page view, so it is deliberately small; a big shop finishes over several requests. No PHP limit is ever raised.
Seconds between heartbeats	How long the module waits before letting another page view do any work. One bounded job per beat, never two, so a burst of visitors is not a burst of jobs. It changes nothing if your shop runs the work on a schedule instead.
Products re-priced through the core per run	A self-check: each snapshot re-prices a random sample through PrestaShop itself and fails the job on a mismatch, rather than writing a history you would have to defend. Zero would switch that off, so zero is refused.

7. Keeping it up to date

The module has to look at your prices once a day. There are three ways, and you only need one.

How	What it means for you
Leave it alone	It does the work a little at a time, carried by the visits your shop already gets, until the day is done. Nothing to set up. This is the normal choice.
Put it on a schedule	If your shop has a daily task scheduler, the Settings page shows one line to give to whoever manages your hosting. The work then happens at night instead of during the day.
Call it from a web address	If your host offers scheduled web calls instead, the Settings page shows the address to use. **Treat it like a password** - anyone who has it can make your shop do this work.

Changing a price starts the day again. If you save a product, or add, edit or delete a reduction, the module notices and looks again - otherwise a price you cut at noon and put back at six would never be recorded, and that low is exactly what a 30-day minimum is made of.

If you are checking by hand, open the page twice. The moment a price is saved is too early for the module to read it; the next visit records it.

8. Turning it off and uninstalling

To hide the line for a while, set **Stop all front-office output** to Yes. The history keeps being recorded and nothing is lost; set it back to No and the line returns.

Uninstalling removes its records, its menu entries and all its settings. Nothing of it is left behind.

> **It also deletes the recorded price history**, and that history cannot be rebuilt beyond what your orders and specific prices still prove. If you may want it later, export the CSV from **Jobs and evidence** first.

9. When something looks wrong

Problem	What to check
No line anywhere	Is **Stop all front-office output** still on? It is on at install. Then clear the shop's cache.
Every product says "Nothing to announce"	The history has not been built. Run **Rebuild the past window**, then **Recompute the decisions**.
No line on one product	Open **Price history**, search its reference and press **Why** (6.2). The panel names the status and the reason.
A product sits in "Cannot be placed in time"	Its reduction has no start date, and PrestaShop keeps no creation date for one. Give it a start date in the product's Prices tab and recompute.
A job says "partial"	Normal, not an error - it did a bounded chunk. Click it again.
A setting changed nothing	Settings change what the engine decides, not what is already stored. Run **Recompute the decisions**.
The line is in the wrong place on the page	Your theme puts its price block together differently. Use the setting at the end of 6.4 to say where the line belongs.

10. Help

Write to **office@gurucoders.ro**.

If a product is not showing what you expected, send two things: the product reference, and a screenshot of the **Why** panel for it (6.2). That panel contains the whole answer - the window, the figures and the evidence behind them - and it usually saves a round trip.

If your question is a technical one - how it hangs off PrestaShop, which files it adds, where the records are kept - ask and we will send the technical notes; they are deliberately not in this manual.