Customer ID Documentation

Complete PDF Manual

Guide for Magento administrators, technical project owners, and implementation teams.

LAST UPDATED
2026-10-13

SOURCE
Open complete documentation online

EXTENSION
Magento 2 Extension Customer ID

Table of Contents

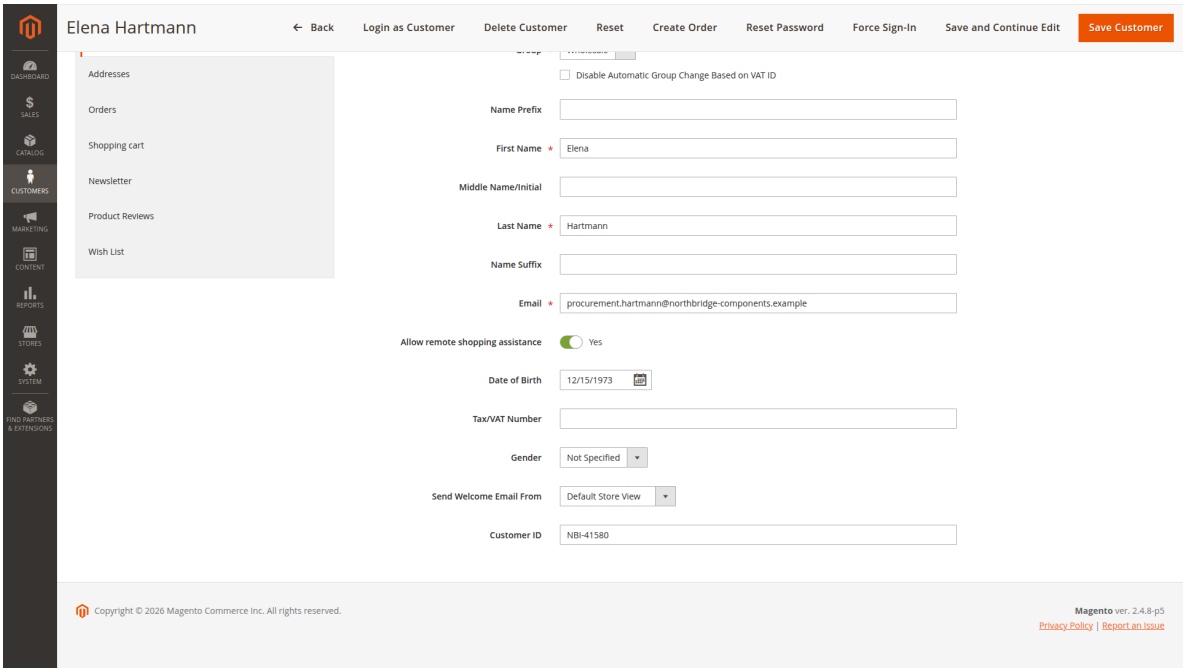
Customer ID Documentation · customer-id

Overview
Customer ID
Getting Started
Installation
Configuration
Optional Sample Data
Features
API Integration
Customer ID Login
Range Management
Template Formatting
Orders and Emails
Field names reference
Reference
Configuration Fields
Troubleshooting
Common Issues
FAQ

Customer ID

SoftwareSilo Customer ID adds a separate customer number to Magento customer accounts. Magento can assign the number automatically, format it with a prefix and zero padding, accept it as an alternative storefront login and carry it onto registered-customer orders.

The extension uses the customer attribute `customer_id`. This is separate from Magento's internal numeric customer entity ID and from the customer's email address.



What the extension does

- Adds an editable **Customer ID** field to the Magento customer form and grid.
- Assigns the next free number when automatic numbering is enabled.
- Supports formats such as `1001`, `CUST-1001` or `B2B-001001`.
- Supports a fixed numeric range such as `1000-2499`.
- Prevents the same customer ID from being assigned twice within one website.
- Lets customers sign in with either their email address or customer ID.
- Optionally displays the number on the customer account dashboard.
- Copies the number to registered-customer orders and the sales order grid.
- Makes the value available through Magento's existing customer and order service contracts.

Important boundaries

Customer ID does not create customer-group-specific number ranges, year-based templates or a separate number for each store view. It also does not certify an accounting

or compliance process. If an ERP owns the customer number, decide which system is authoritative before enabling automatic assignment in Magento.

Start here

- Install Customer ID
- Configure numbering and display
- Choose a number format
- Plan website ranges
- Enable customer ID sign-in
- Use customer IDs in orders
- Use Magento APIs
- Troubleshoot common issues
- FAQ

Configuration is under **Stores > Configuration > MageB2B > Customer ID**.

Installation

Install the package from the SoftwareSilo Composer repository in the same way as your other SoftwareSilo extensions. Take a database backup and test the change in staging before running it on a live store.

Requirements

- Magento 2.4
- PHP `8.1` or newer within the PHP version supported by your Magento release

Install the package

Replace `repo.softwaresilo.io`, `softwaresilo`, `https://repo.softwaresilo.io/` and `<token>` with the repository details shown under **Account > Licenses** in your SoftwareSilo account — see Installation & Updates.

```
composer config bearer.repo.softwaresilo.io <token>
composer config repositories.softwaresilo composer
https://repo.softwaresilo.io/
composer require mageb2b/customer-id
php bin/magento module:enable MageB2B_ExtensionManager
MageB2B_CustomerId
php bin/magento setup:upgrade
php bin/magento cache:flush
```

Run dependency injection compilation and static-content deployment when your deployment mode and release process require them.

Verify the installation

```
php bin/magento module:status MageB2B_CustomerId
```

Then check these two places in Magento Admin:

1. Open **Stores > Configuration > MageB2B > Customer ID**.
2. Open **Customers > All Customers**, edit a customer and confirm that the **Customer ID** field is present.

The installed customer attribute code is `customer_id`. The extension also adds the `customerid` field to sales orders and the sales order grid. These names are intentionally different because Magento already uses `customer_id` for its internal order-to-customer relation.

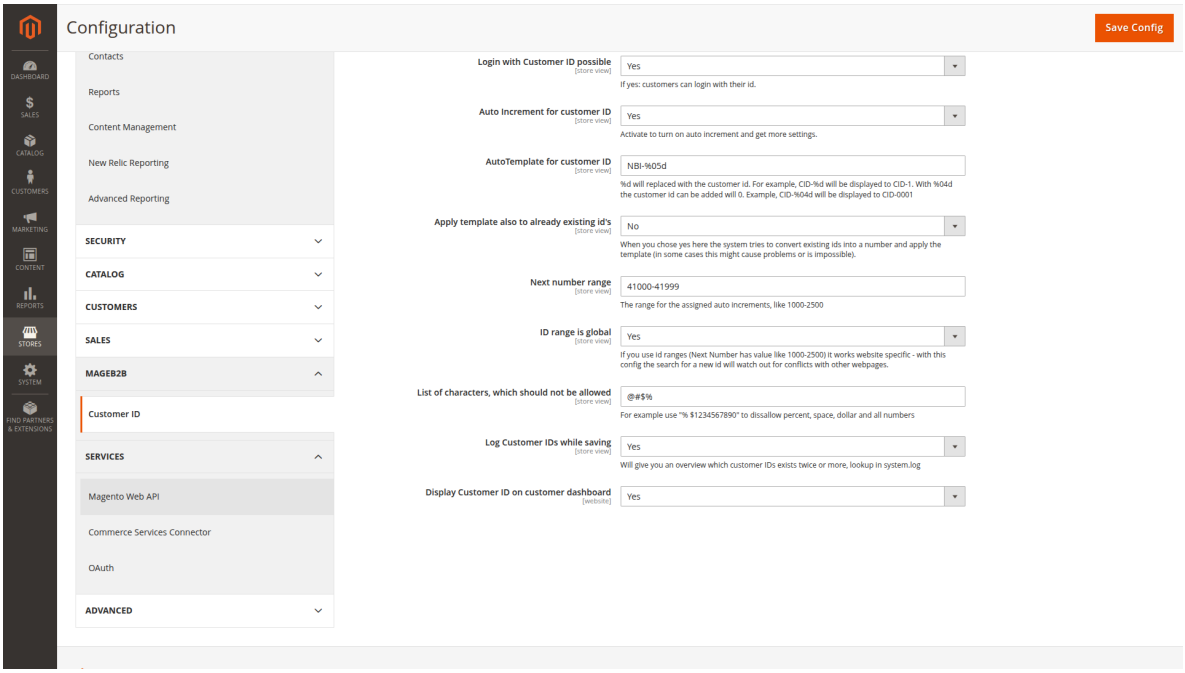
Before enabling automatic assignment

Decide whether Magento or another system owns customer numbers. If you are migrating existing values from an ERP, import and verify them first, choose the desired website scope and only then enable automatic numbering for new accounts.

Continue with Configuration.

Configuration

Open **Stores > Configuration > MageB2B > Customer ID > General settings**. Set the configuration scope deliberately: numbering is resolved from the default store view of the customer's website.



Shipped defaults

A fresh installation enables **Login with Customer ID possible** and **Auto increment for customer ID**, starts numbering at `1` with the plain `%d` template, and leaves dashboard display and duplicate logging off. Review these before your first customer is created — the login option is on by default.

Recommended first setup

For a single website that wants numbers such as `B2B-01000`:

Setting	Value
Login with Customer ID possible	Yes, if customers should use the number to sign in
Auto increment for customer ID	Yes
Auto template for customer ID	`B2B-%05d`
Apply template also to already existing IDs	No
Next number range	`1000-9999`

Setting	Value
ID range is global	No for one website; review carefully for several websites
Disallowed characters	Leave empty unless your integration requires a restricted character set
Log Customer IDs while saving	No; enable temporarily when investigating duplicates
Display Customer ID on customer dashboard	Yes if customers need to quote it to your team

Create a staging customer after saving and confirm the resulting ID before importing or editing production accounts.

Setting reference

Login with Customer ID possible

When enabled, a storefront username without `@` is treated as a possible customer ID. Email login remains available and Magento still validates the normal customer password.

Auto increment for customer ID

When enabled, a new customer without a number receives the next free formatted ID. When disabled, Magento leaves the field empty unless an administrator or integration supplies a value.

Auto template for customer ID

The template must contain one integer placeholder such as `%d`, `%04d` or `%06d`. Prefixes and suffixes may be literal text. See Template formatting.

Apply template also to already existing IDs

When enabled, the Admin customer form attempts to extract a number from an existing value and presents it with the current template. For example, `42` can appear as `B2B-0042`. Values without a usable numeric part remain unchanged.

Treat this as a migration aid, not a bulk conversion tool. Test representative IDs first and check downstream systems before saving changed customers.

Next number range

Use `1000-2500` to restrict automatic allocation to that inclusive range. Magento searches for the first free number in the configured scope. When the range is full, the customer save shows a warning and no new number is assigned.

In non-range mode, the live next number is held as internal increment state per website's

default store code. Do not assume that changing a single number in this field resets an established counter. If a live sequence must be moved, verify the current state in staging and contact support instead of editing database records directly.

ID range is global

This setting affects the search for automatically generated IDs. When disabled, another website may reuse the same generated value. When enabled, allocation checks other websites as well.

When enabled, manual duplicate validation becomes global as well — a value already used on any website is rejected. When disabled, manual edits are only checked within the customer's website, so imports and integrations must enforce any stricter company-wide rule themselves.

Disallowed characters

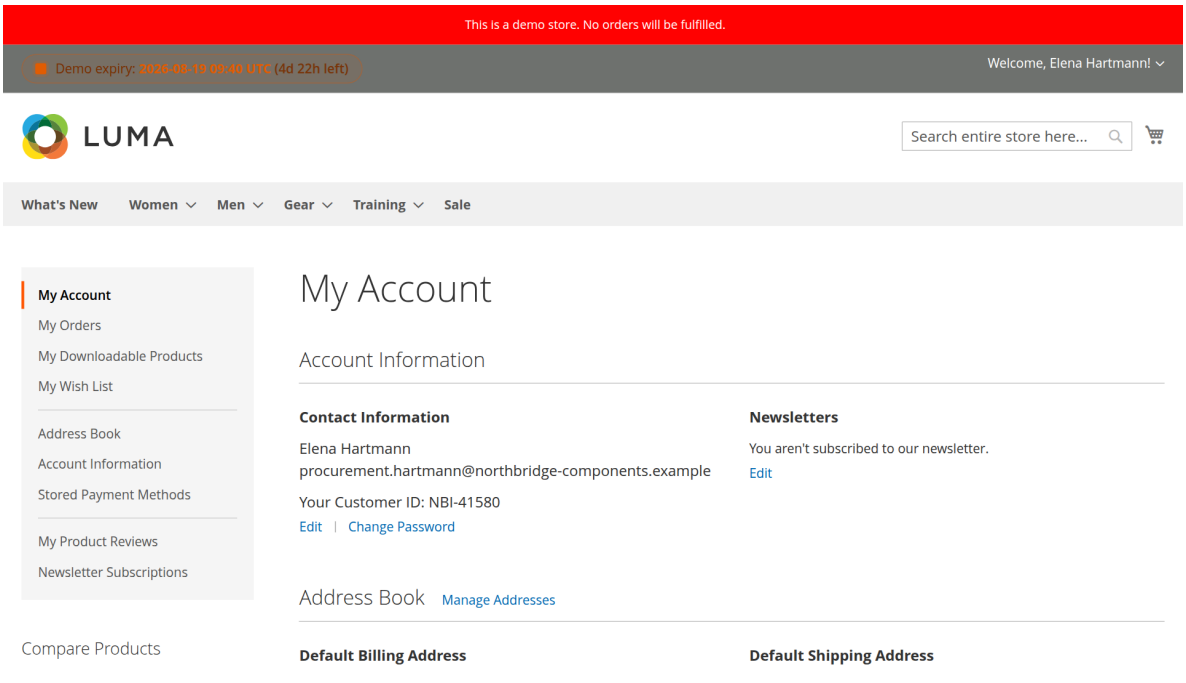
Enter the individual characters that must not appear. A value such as `% $` blocks the percent sign, spaces and the dollar sign. The field is not a regular expression. If an edited value contains a blocked character, Magento keeps the previous customer ID and shows a warning.

Log Customer IDs while saving

Duplicate IDs are always rejected within the current website. This switch only controls whether the blocked attempt is also written as a warning to Magento's standard log, normally `var/log/system.log`.

Display Customer ID on customer dashboard

Adds the value to the signed-in customer's account dashboard. It does not automatically add the number to every account page, invoice PDF or email template.



Safe rollout checklist

1. Export or otherwise record existing customer numbers.
2. Decide whether numbers may repeat across websites.
3. Test the template and range with new staging customers.
4. Verify manual edits, storefront login and an order placed by a registered customer.
5. Confirm that ERP, CRM and email templates read the correct field names.

See Range management for multi-website examples.

Optional Sample Data

The separate `mageb2b/customer-id-sample-data` package is intended for development and disposable demonstrations. It creates a small set of sample customer accounts with `CID-...` values when Magento runs `setup:upgrade`.

Do not install it on a production store. The package contains fixed demonstration identities and passwords and is not a migration tool.

Install in a development environment

```
composer require mageb2b/customer-id-sample-data
php bin/magento module:enable MageB2B_CustomerIdSampleData
php bin/magento setup:upgrade
```

Verify the fixture records

```
php bin/magento sampledata:verify:customer-id
```

The command compares the installed records with the package CSV and reports missing or mismatched customer IDs.

Remove the fixture records

```
php bin/magento sampledata:reset:customer-id
```

The reset command asks for confirmation, deletes only the customer emails listed in its fixture CSV and removes the package's patch state so `setup:upgrade` can install the records again. Review the command output before continuing; deleting customers is destructive even in a development environment.

For migration of real customer numbers, use a controlled customer import or integration process instead.

API Integration

The extension does not add a dedicated Customer ID REST route or GraphQL field. It extends Magento's existing data instead:

- Customer records use the EAV custom attribute `customer_id`.
- Orders expose the copied value as the OrderInterface extension attribute `customerid`.

Access remains subject to Magento's normal API authentication, ACLs and website scope.

Read a customer

Magento's standard customer endpoint can include the value under `custom_attributes`:

```
GET /rest/V1/customers/123
Authorization: Bearer <integration-token>
```

Relevant response fragment:

```
{
  "id": 123,
  "email": "purchasing@northstar-industrial.example",
  "custom_attributes": [
    {
      "attribute_code": "customer_id",
      "value": "B2B-01042"
    }
  ]
}
```

Customer search uses Magento's standard customer service contracts with the same attribute code. Writing `customer_id` through `custom_attributes` is restricted: only admin and integration API tokens may set it — customer and guest tokens get rejected on account create and customer/address repository saves. See Protected customer attributes for the shared mechanism. Test the exact payload against your Magento release and integration permissions before building a synchronization job.

Read an order

For a registered-customer order, the copied business number can appear in order extension attributes:

```
{
```

```
  "extension_attributes": {
    "customerid": "B2B-01042"
  }
}
```

The name is `customerid`, without an underscore, because Magento already uses `customer_id` for the internal customer entity relation. Guest orders normally have no copied Customer ID. The field names reference compares all three identifiers used by the extension.

Integration ownership

Choose one owner for customer-number creation:

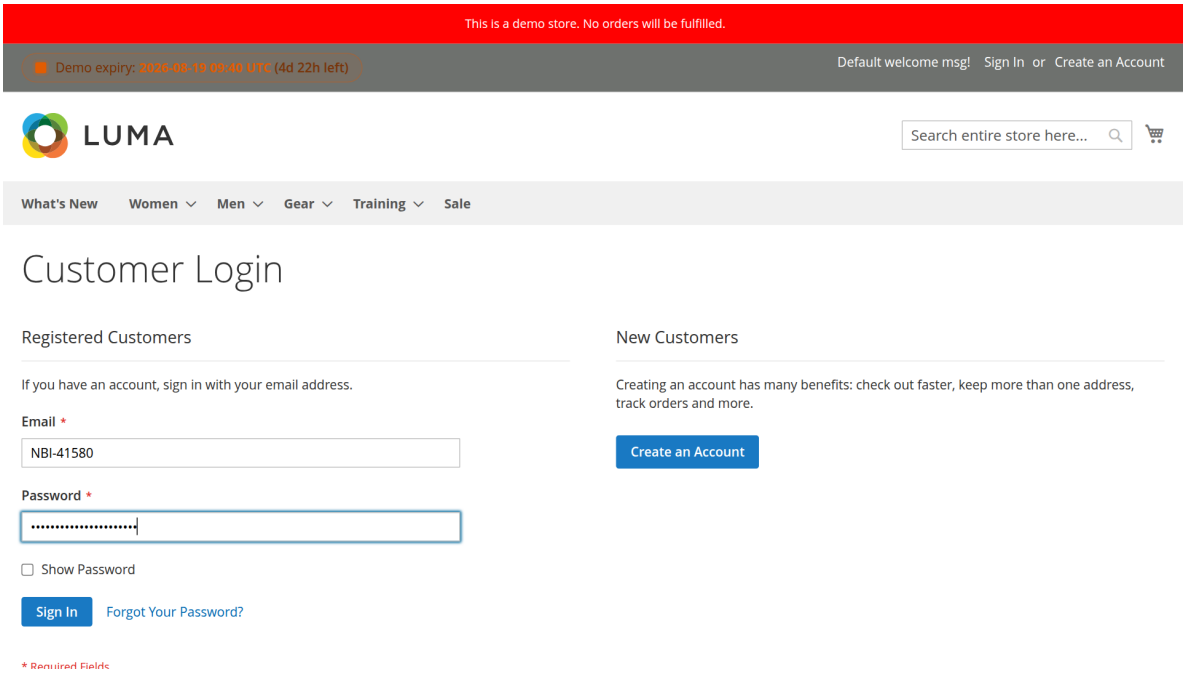
- If Magento owns it, let automatic assignment run and send the resulting `customer_id` to the ERP or CRM.
- If the ERP owns it, leave automatic assignment off during import and write the ERP value to `customer_id`.

Duplicate values are rejected — across all websites when **ID range is global** is enabled, otherwise within the customer's website. Imports should handle the validation error instead of silently generating a replacement.

There is no module-specific GraphQL schema. A GraphQL integration needs a separately implemented field or another supported integration layer.

Customer ID Login

Enable **Login with Customer ID possible** under **Stores > Configuration > MageB2B > Customer ID**. Customers can then enter either their email address or customer ID with the same Magento customer password.



The extension changes the login input to accept text. A username containing `@` follows Magento's normal email-login path. A username without `@` is looked up as a customer ID within the current website; when a match is found, the extension passes the associated email to Magento's standard authentication process.

This works for the regular customer login and Magento's AJAX authentication form.

Supported ID formats

Both numeric and prefixed IDs work:

- `1042`
- `00001042`
- `CUST-01042`
- `B2B-001042-EU`

Customers should enter the stored value as shown in their account. The lookup can also reconcile a numeric value with the configured template in common cases, but published instructions should use the complete customer ID to avoid ambiguity.

Security behavior

Customer ID login does not bypass Magento authentication:

- The normal customer password is still required.
- Magento's account status, CAPTCHA and lockout behavior still apply.
- Lookup is limited to the current website.
- Email login remains enabled.

Sequential numbers may be easier to guess than email addresses. Treat a customer ID as a convenient username, not a secret.

Test the workflow

1. Create a staging customer with a known ID.
2. Sign out of every customer session.
3. Sign in once with the email address and once with the complete customer ID.
4. Repeat the test from any checkout authentication popup used by the storefront.
5. If the store has several websites, confirm that the ID resolves only on the intended website.

If the login fails, see Common issues.

Range Management

Set **Next number range** to two positive numbers separated by a hyphen, for example `1000-2499`. The range is inclusive. When Magento creates a customer without an ID, the extension scans from the lower bound and assigns the first free formatted number.

One website

A single B2B website might use:

```
Template: B2B-%05d
Range:    1000-9999
```

The first free number becomes `B2B-01000`, followed by `B2B-01001` and so on. A gap left by a removed or renumbered customer can be reused because range mode searches for the first free value.

Several websites

Choose the configuration scope for each website and set its template and range. For example:

Website	Template	Range
Industrial sales	`IND-%05d`	`1000-4999`
Workplace supplies	`WRK-%05d`	`5000-8999`

With **ID range is global** set to **No**, automatic allocation checks customers from the current website. Other websites can therefore reuse the same formatted value.

With it set to **Yes**, automatic allocation also considers values assigned on other websites, and manual duplicate validation becomes global too — an ID already used on any website is rejected. This is useful when several websites feed one ERP namespace. With **No**, manual edits are only checked within the customer's website, so imports and integrations must follow their own stricter policy if needed.

Ranges are not assigned per customer group or per store view.

Range exhaustion

When every number is used, the Admin save shows **Could not find a free ID in range …** and the new customer remains without an automatically assigned number. The extension does not send an exhaustion email or create a separate monitoring alert.

Plan enough room, review range consumption periodically and extend the range before it fills. Test a range change in staging, especially when prefixes or website scope also

change.

Range versus continuing sequence

Range mode repeatedly searches the configured interval. Without a hyphenated range, the extension uses internal increment state and skips collisions. Saving a manual ID whose numeric part is at or above the current counter advances the counter beyond that number. The live counter is not a customer-facing configuration value, so avoid database edits intended to force the next number.

Template Formatting

Customer ID uses PHP-style integer formatting. The template must contain an integer placeholder; the most useful forms are `%d` and a zero-padded form such as `%05d`.

Template	Number	Result
`%d`	42	`42`
`%05d`	42	`00042`
`CUST-%05d`	42	`CUST-00042`
`B2B-%06d-EU`	42	`B2B-000042-EU`

What is supported

- Literal prefixes and suffixes.
- Decimal integer substitution with `%d`.
- Minimum width and zero padding, for example `%04d`.

Only one numeric sequence is generated. Placeholders such as `{year}`, `{month}`, `{store}` and `{group}` are not supported, and there are no separate padding settings outside the template.

Changing an existing format

Changing the template affects newly generated values. Existing values remain as stored unless **Apply template also to already existing IDs** is enabled and the Admin form can extract a numeric part.

Before changing a live format:

1. Check whether customers use their IDs to sign in.
2. Check ERP, CRM, exports and custom email templates.
3. Try examples with and without leading zeros.
4. Keep the old value unchanged when another system treats it as a permanent key.

Customer numbers are identifiers, not presentation-only labels. A cosmetic format change can break lookups in another system.

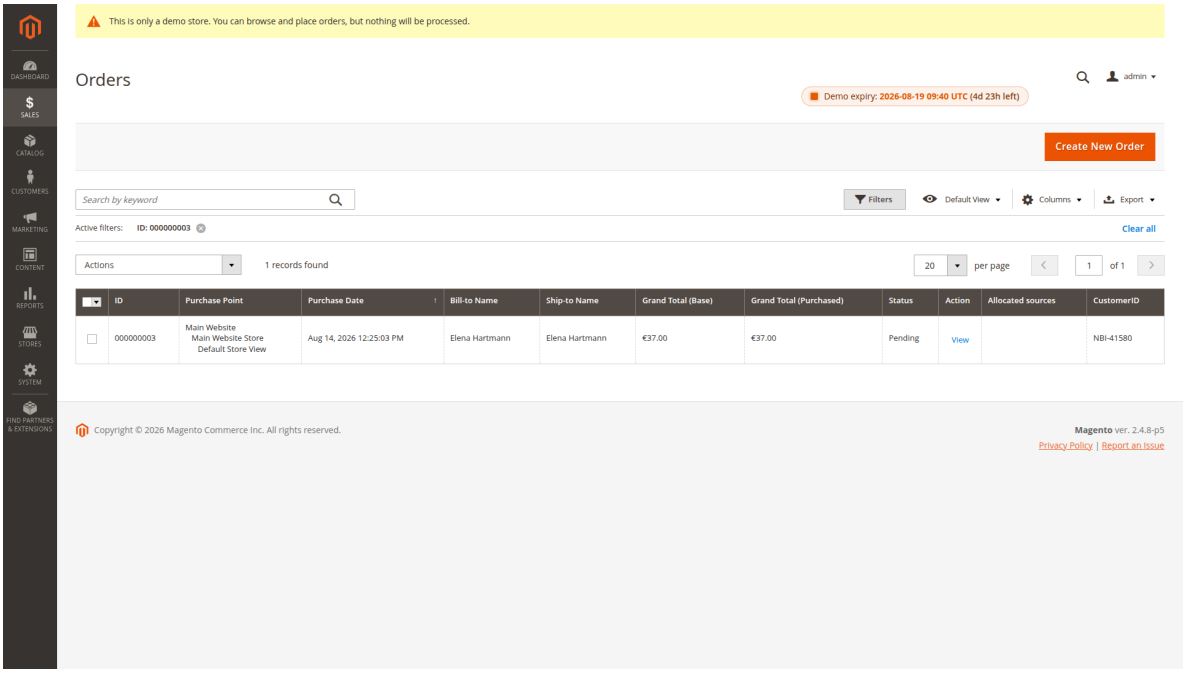
Orders and Emails

When a signed-in customer places an order, or an administrator creates an order for a customer account, the extension copies the account's Customer ID into the order field `customerid`.

This preserves the business number used at order creation. Changing the customer account later does not rewrite historical orders.

Magento Admin

The sales order grid includes a **Customer ID** column. Use it to match Magento orders with ERP, service-desk or telephone enquiries without confusing the business number with Magento's internal customer ID.



Guest orders normally leave the field empty because no customer account supplies a Customer ID.

APIs

Loaded orders expose the value through the OrderInterface extension attribute `customerid`. See API integration for the field-name distinction.

Order emails

The extension adds the copied value to the order object as `mageb2b_customer_id` before Magento renders an order email. It does not automatically change every email template. See the field names reference before mapping customer and order data in an integration.

If your customized order template needs the value, add and test the variable in staging. Keep the template change in your theme or email customization rather than editing vendor files.

What is not changed

Customer ID does not alter order numbering, invoice numbering or credit memo numbering. It also does not place the value into invoice PDFs by itself.

Field names reference

The extension uses three field names for the same business identifier. They belong to different Magento data contracts and are not interchangeable.

Context	Field name	Where it appears
Customer account	`customer_id`	Customer EAV attribute, customer forms, and customer API `custom_attributes`
Sales order	`customerid`	`sales_order`, `sales_order_grid`, and the OrderInterface extension attribute
Order email	`mageb2b_customer_id`	Additional value placed on the order object before Magento renders the email template

Magento already uses `customer_id` on an order for the internal customer entity relation. The extension therefore stores the copied business identifier as `customerid` without an underscore.

The order value is a snapshot. Changing `customer_id` on the customer account does not rewrite `customerid` on existing orders.

Integration checklist

1. Read and update the customer value through the `customer_id` custom attribute.
2. Read the historical order value through the `customerid` extension attribute.
3. Use `mageb2b_customer_id` only in the order email rendering context.
4. Do not map the business identifier to Magento's native order `customer_id` column.

Related pages

- API Integration
- Customer IDs on Orders
- Template Formatting

Configuration Fields

This reference lists the settings exposed by `mageb2b/customer-id` in Magento Admin.

Configuration paths are included for controlled deployments and `config:set`. Use Magento Admin for normal changes so field validation and scope inheritance still apply. The extension status display is not listed because it reports installation state rather than a configurable value.

General settings

Field	Configuration path	Input	Scope
Login with Customer ID possible	`customer_id/customer_id/login`	Select	Default, Website, Store view
Auto increment for customer ID	`customer_id/customer_id/auto_increment`	Select	Default, Website, Store view
Auto template for customer ID	`customer_id/customer_id/id_template`	Text	Default, Website, Store view
Apply template also to already existing id's	`customer_id/customer_id/template_force_apply`	Select	Default, Website, Store view
Next number range	`customer_id/customer_id/next_increment`	Text	Default, Website, Store view
ID range is global	`customer_id/customer_id/idrange_global`	Select	Default, Website, Store view
List of characters, which should not be allowed	`customer_id/customer_id/disallowed_characters`	Text	Default, Website, Store view
Log Customer IDs while saving	`customer_id/customer_id/log_customerids`	Select	Default, Website, Store view
Display Customer ID on customer dashboard	`customer_id/customer_id/display_customer_id_on_customer_dashboard`	Select	Default, Website

Visibility dependencies:

- `customer_id/customer_id/id_template` is available when

`customer_id/customer_id/auto_increment` is `1`.

- `customer_id/customer_id/template_force_apply` is available when `customer_id/customer_id/auto_increment` is `1`.
- `customer_id/customer_id/next_increment` is available when `customer_id/customer_id/auto_increment` is `1`.
- `customer_id/customer_id/idrange_global` is available when `customer_id/customer_id/auto_increment` is `1`.

Paths without an Admin field

These paths ship `config.xml` defaults but have no Admin field. Change them only via `config:set` when you know what you are doing:

Purpose	Configuration path	Default
Collision retry limit for non-range auto-allocation	`customer_id/security/max_recursion_attempts`	`1000`

In non-range mode the generator retries each colliding candidate. When retries exceed this limit the customer save fails with **Unable to generate unique customer ID after N attempts**. The usual fix is more ID space (longer template numbers or a configured range), not a higher limit.

Common Issues

A new customer has no Customer ID

Confirm that **Auto increment for customer ID** is enabled in the scope resolved for the customer's website. Then check whether the configured range is full and whether the template contains a valid integer placeholder.

For a full range, Magento shows **Could not find a free ID in range ...** during the save. Extend the interval before creating more accounts.

Magento rejects the template

Use a template containing `%d` or a padded variant such as `%05d`. Values such as `{id}`, `{year}` and plain text without an integer placeholder are invalid.

Editing the next number does not move the sequence

In non-range mode, the active counter is held in internal increment state. The configuration value is not a reliable live reset control after installation. Do not edit Magento flag tables directly. Reproduce the desired sequence change in staging and contact support for a controlled adjustment.

A manual ID is rejected as a duplicate

Another customer already has that value — anywhere when **ID range is global** is enabled, otherwise within the same website. Search **Customers > All Customers** using the **Customer ID** column. Duplicate validation is always active; the logging option only adds a diagnostic warning to the standard Magento log.

An edited ID reverts to its previous value

The proposed value contains a character listed under **Disallowed characters**. That setting is a literal list of individual characters, not a regular expression. Remove the character from the ID or revise the policy.

The save reports "Invalid characters in customer ID"

The built-in validation also rejects values longer than 255 characters or containing script-style markup such as `<script` or `javascript:` — independently of the **Disallowed characters** list. This applies to manual edits and integration writes alike.

Auto-allocation dies after many retries

The save fails with **Unable to generate unique customer ID after N attempts**. In non-range mode the generator retries colliding candidates up to a hidden limit of 1000

attempts (`customer_id/security/max_recursion_attempts`, no Admin field). Reaching it means the sequence space is nearly exhausted — widen the template (more digits) or configure a number range rather than raising the limit.

Customer ID login fails

Check all of the following:

- **Login with Customer ID possible** is enabled in the current store scope.
- The customer belongs to the current website.
- The entered value matches the stored Customer ID.
- The customer password is valid and the account is not locked.
- A custom login theme or checkout popup has been tested, including AJAX login where applicable.

Try the email address with the same password. If email works but Customer ID does not, compare the stored `customer_id`, configured template and current website.

The ID is missing from an order

Guest orders do not normally receive a Customer ID. For a registered-customer order, confirm the customer already had a value when the order was created. Historical orders are not backfilled when the account changes later.

The ID is missing from an email or PDF

Dashboard display does not change every document. Order email rendering receives `mageb2b_customer_id`, but a customized template must use it. Invoice PDFs are not modified by this extension.

API data uses an unexpected field name

Customer data uses `customer_id` as a custom attribute. Order data uses the extension attribute `customerid`. There is no built-in `customer_number`, `customer_id_number` or module-specific GraphQL field.

FAQ

Is the Customer ID Magento's internal customer ID?

No. `customer_id` is a separate business identifier stored as a customer attribute. Magento still keeps its own numeric entity ID internally.

Can Magento assign the number automatically?

Yes. It can use a continuing sequence or the first free number in a configured range.

Which template variables are supported?

Use an integer placeholder such as `%d` or `%05d`, with optional literal text around it. Dynamic variables such as year, month, website or customer group are not supported.

Can different customer groups have different ranges?

No. Configuration can differ by Magento scope, but the extension does not assign ranges by customer group.

Can two websites use the same number?

Yes, when global range checking is disabled. Enable it if automatic allocation must avoid numbers used by other websites — it also makes manual duplicate validation global. With the setting off, manual edits are only checked within the customer's website, so integrations must enforce any stricter company-wide rule.

What happens when a range is full?

Magento shows a warning during the customer save and leaves the account without an automatically assigned number. The extension does not send a separate exhaustion notification.

Can customers still sign in with email?

Yes. Enabling Customer ID login adds an alternative username; it does not disable email login or password checks.

Do prefixed IDs work for login?

Yes. A stored value such as `B2B-01042` can be used. Give customers the complete value to avoid ambiguous lookups.

Is uniqueness optional?

No. A duplicate customer ID is rejected within the current website. The logging switch only controls whether the blocked attempt is also written to the Magento log.

Does the extension put the ID on orders?

Yes, for orders tied to a customer account that already has an ID. The order keeps its copied value even if the account changes later.

Does it change order or invoice numbers?

No. Customer ID is independent of Magento order, invoice and credit memo sequences.

Is there a dedicated REST or GraphQL API?

No. Magento's customer service contract can expose `customer_id` as a custom attribute, and orders expose `customerid` as an extension attribute. The package declares no GraphQL field of its own.

Does dashboard display add the number to emails and PDFs?

No. It only controls the customer account dashboard. Order emails can use the provided order variable after template customization; PDFs are not changed by the extension.

Should I install the sample-data package in production?

No. It is for disposable development and demo environments only.