

Coupon Code Generator

Created by Unknown User (malynow), last modified by Anton Lashuk on янв 05, 2019

- Installation
- Configuration
 - General Configuration
 - Email templates
 - Email variables
- Creating new rule
- Generating coupons
- Help Desk Ultimate integration
- Uninstallation
- Troubleshooting

Extension page: <http://ecommerce.aheadworks.com/magento-extensions/coupon-code-generator.html>

Magento store administrator has to compose numerous individual coupons on a daily basis. This routine and time consuming process can be automated but Magento native functionality is not enough to accomplish this. The Coupon Code Generator extension automates this process and lets merchants provide customers with unique codes in one click.

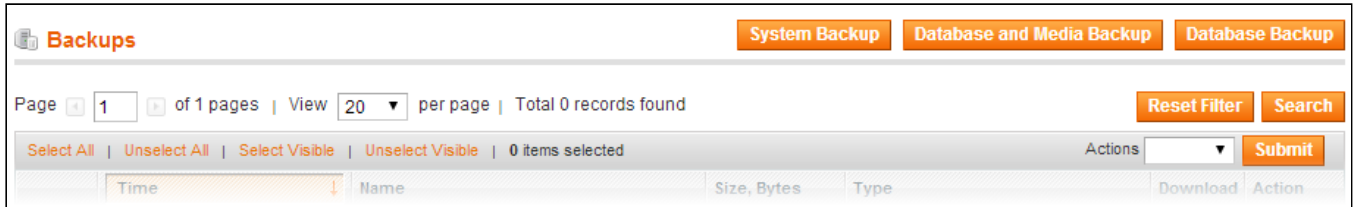
Installation

1. Backup your web directory and store database.

[Click to view details](#)

You can make backup copies with any tool you find appropriate

If you are going to use the native Magento backup function, navigate to **System -> Tools -> Backups** and perform **System** and **Database** backups



2. Log in to the Magento backend

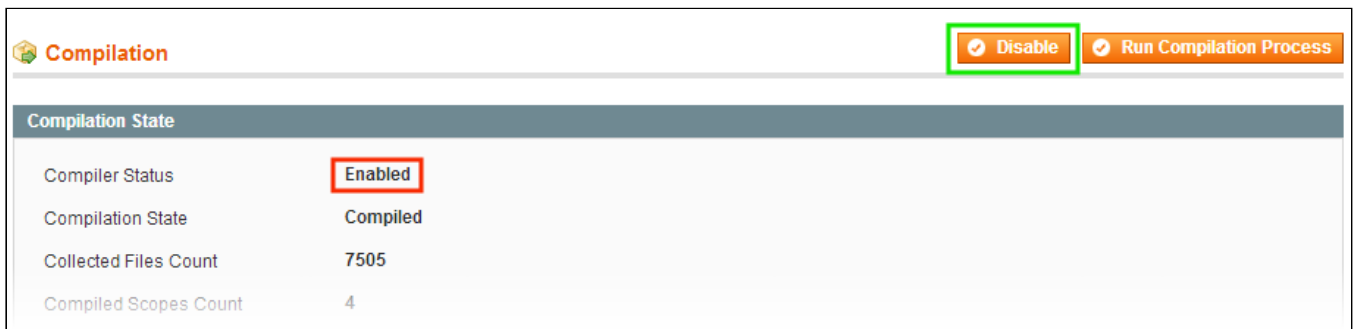
3. Disable compilation

[Click to view details](#)

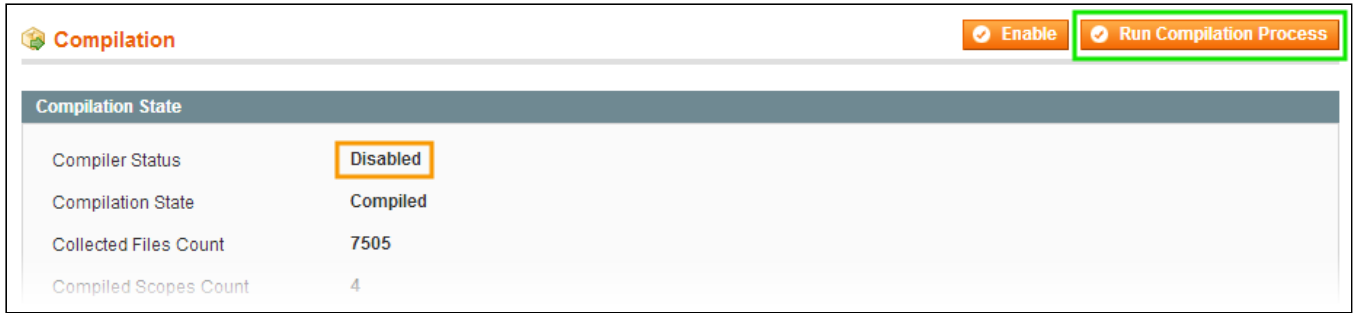
Navigate to **System -> Tools -> Compilation**.

If Compiler status is Disabled, you can skip to the next step

If Compiler is enabled, disable it.



IMPORTANT: after the extension is installed, you can enable the compilation - again; **IT IS CRUCIAL** that you use "Run Compilation Process" function, not just "Enable button"



Installing an extension with the Compilation enabled will result in store downtime.

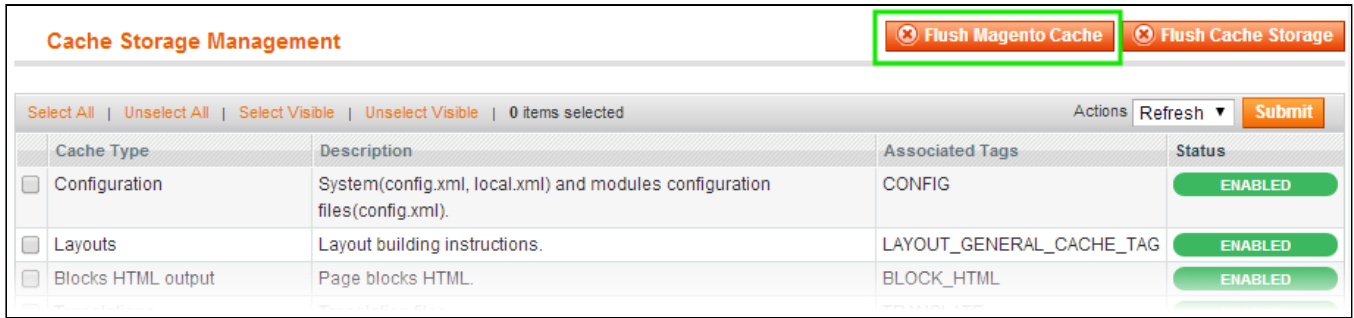
4. Flush store cache

Click to view details

You can flush the store cache in 2 ways:

- **via the backend:**

Navigate to **System -> Cache Management** menu, and click **Flush Magento Cache** button



- **via filesystem:**

On your server, navigate to Magento root folder, then proceed to **/var/cache/**; delete all the content there.

5. Download the extension package from your account and extract the downloaded archive

6. Copy the content of **/Step_1/** folder to your store's root directory

7. Copy the content of **/Step_2/** folder to your store's root directory

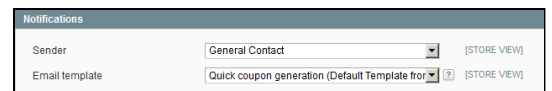
8. Flush store cache again; log out from the backend and log in again.

Configuration

General Configuration


General extension settings can be found under **System > Configuration > AheadWorks extensions > Coupon Code Generator** or **Promotions > Coupon Code Generator > Settings**.

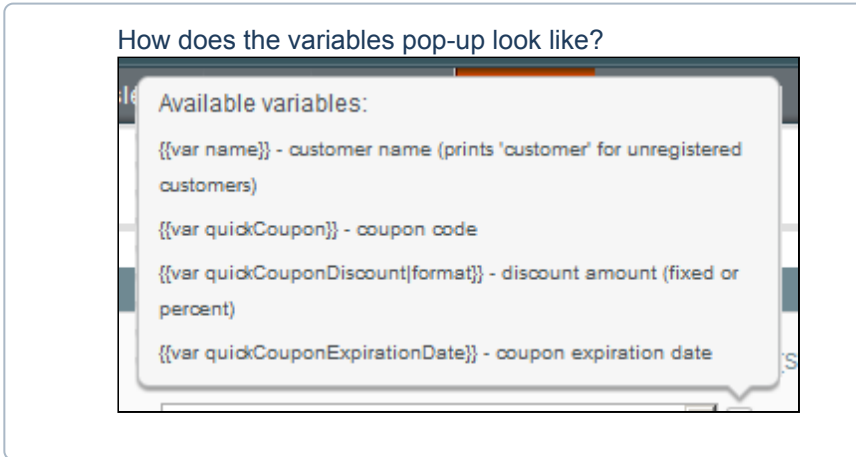
General settings are represented by the notifications. The notifications are sent in case the customer's email is submitted in the **Email** filed of



the **Generated Coupons** section. Other coupon related configuration options will be covered below in this very readme file.

- **Sender** - allows defining the email notifications sender;
- **Email template** - allows selecting the notification email template.

By clicking  button on the left of the **Email template** filed store administrator is presented with the Coupon Code Generator specific email variables pop-up.



Email templates

The notification email templates are caught from **System > Transactional Emails**. One can create and edit any number of templates to be available for notifications.

Additionally Coupon Code Generator comes with a pre-defined notification email template that can be found and edited under: **app/locale/en_US/template/email/aw_coupongenerator/**

Email variables

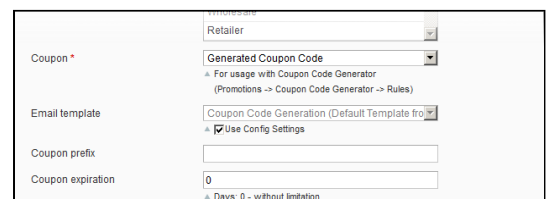
The extension introduces notification email specific variable for further convenience.

Variable	Function
var_name	customer name
var quickCoupon	coupon code
var quickCouponDiscount format	discount amount (flat or percent)
var quickCouponExpirationDate	coupon expiration date

Creating new rule

Coupon Code Generator extension introduces it's own Magento backend section **Promotions > Coupon Code Generator** represented by the following pages:

- Rules;
- Generated Coupons;
- Settings.



New coupon rules can be added under the **Rules** page by clicking

 button.

The rule creation process is similar to creating native Magento Shopping Cart Price Rule with few exceptions.

To create new coupon rule follow these steps:

1. In the Coupon filed dropdown select **Generated Coupon Code**
2. Decide if the rule should generate coupons with particular prefix with **Coupon prefix** field;
3. If required, set the coupon life time using **Coupon expiration** field;
4. Fill in other required fields.

Once done with editing the rule click  button.

Coupon expiration field defines for how many days the coupon will be active starting from the coupon generation date.

Coupon prefix is letter-digital only, no spaces or special symbols allowed. **Coupon prefix** can be edited later on. Any changes of the prefix will not affect coupons which were already generated.

All the coupon rules will be available under **Generated Coupons** page.

Generating coupons

Coupons can be generated form the **Promotions > Coupon Code Generator > Generated Coupons** page.

Generated Coupons page allows managing all the coupons generated and provides a related info overview.

To generate a coupon code:

1. Select the rule;
 - a. Click the **Select rule** menu (refer to the screenshot on the right);
 - b. In the drop down menu select coupon rule;
2. Type in the recipient email address;

Type in email filed is optional. If you want the coupon code to be emailed to a particular customer using the **template** specified previously - fill in the customer's email. Leave the field blank so no email notifications will be sent. When generated the coupon code will be available in the coupon table below for you to deliver it to customer by any means necessary. *Please note: coupon code can only be sent to a single email address.*
3. Click **Generate** button.

Once you click **Generate** button the coupon with **required settings** is created. If the recipient email address has been specified the coupon code will also be emailed to the customer.

Additionally, coupons can be generated from under the customer's account.

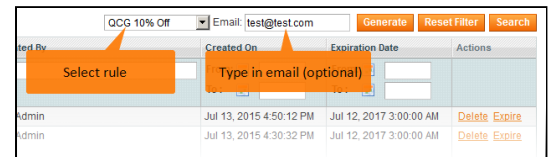
Coupon Code Generator extension introduces the namesake section to the customer account area of the store backend.

In this area it is possible to quickly generate and email a coupon code.

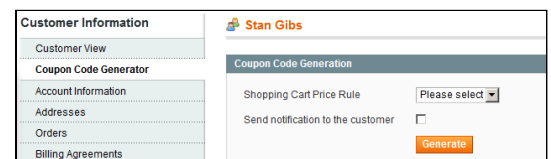
IMPORTANT

Uses per Coupon setting defines how many times **ALL** the coupons created with the rule can be used. The setting is **RETROACTIVE**.

Meaning that if **Uses per Coupon** setting value is changed for the rule created all the coupons generated with this very rule (before or after the change) will change the **Uses per Coupon** value accordingly.



Created By	Created On	Expiration Date	Actions
admin	Jul 13, 2015 4:50:12 PM	Jul 12, 2017 3:00:00 AM	Delete Expire
admin	Jul 13, 2015 4:30:32 PM	Jul 12, 2017 3:00:00 AM	Delete Expire



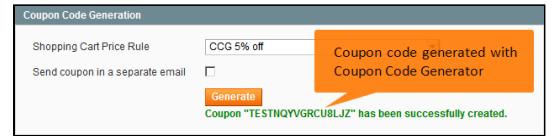
Help Desk Ultimate integration

Coupon Code Generation block can be also found at the [Help Desk](#)

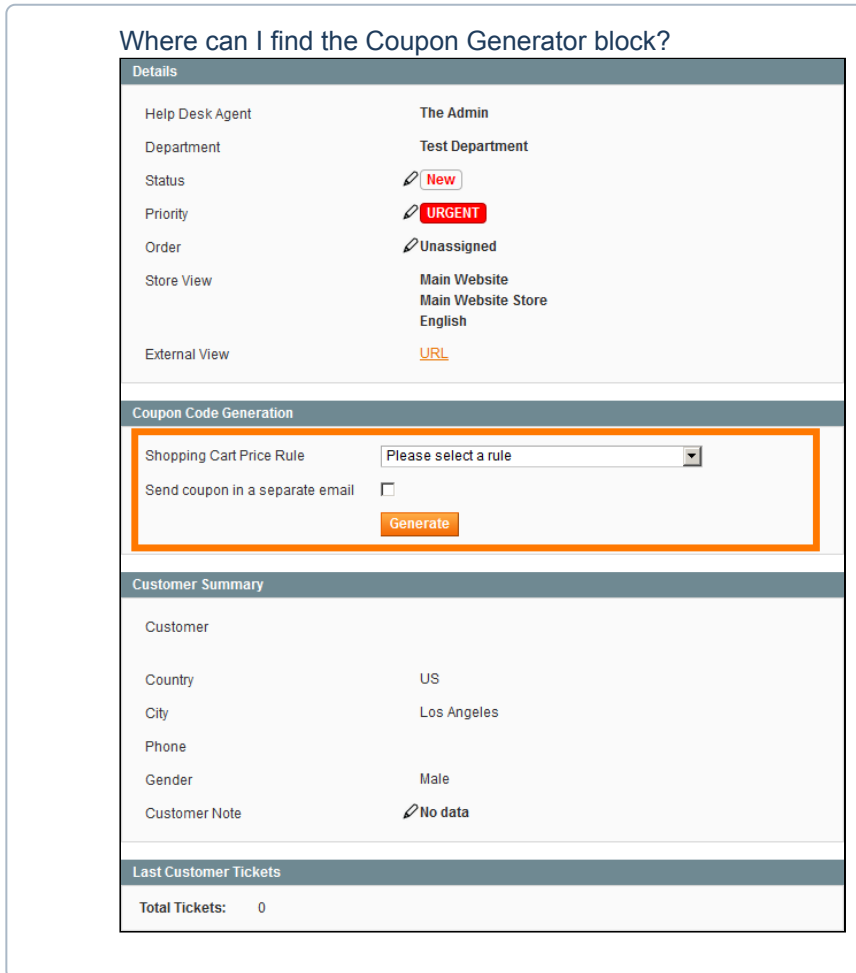
Ultimate 3.x ticket screen, providing Help Desk extension installed.

Coupon Code Generation block allows generating coupon codes with the coupon rules defined when replying to the customer's ticket.

- **Shopping Cart Price Rule** - allows defining the rule coupon should be generated with;
- **Send coupon in a separate email** - allows sending generated coupon code in a separate email with pre-defined email template.



Once help desk agent clicks **Generate** button, depending on **Send coupon is a separate email** setting, the coupon code is either generated and displayed below the **Generate** button (see screenshot attached), or sent to the customer.



Uninstallation

If you just need to temporary disable the extension, you can just do the steps 1-5, without deleting the files of the extension.

- Disable compilation, in case it is enabled.
- Login to your FTP, navigate to **app/etc/modules/**
- Open the file **AW_Coupongenerator** and change the following line:

```
<active>>true</active>
```

to

```
<active>>false</active>
```

Now your Magento is unaware of the existence of this module.

- Clear the cache under var/cache
 - Make sure that the site is working properly, otherwise roll back the changes and apply to our [technical support](#).
 - If everything works fine, it is safe to delete the files of the extension.
-

Troubleshooting

After the extension installation the store gives an error, or blank page, or suggests to start Magento installation procedure.

Change the owner of the extracted extension files to the web server user and set 775 permissions on them. Clear the store cache and try again.

There is no aheadWorks extensions under my configuration section, or having the extension tab clicked I get a blank page, or Access Denied error.

Clear the store cache, browser cookies, logout and login again.

I've set up everything correctly, inserted the HTML code but there is nothing on that page.

Clear the store cache, clear your browser cache and domain cookies and refresh the page.

My configuration changes do not appear on the store.

Clear the store cache, clear your browser cache and domain cookies and refresh the page.

You can always find the latest version of the software, full documentation, demos, screenshots and reviews at <http://ecommerce.aheadworks.com>

License agreement: <http://ecommerce.aheadworks.com/LICENSE-M1.txt>

Contact us: <http://ecommerce.aheadworks.com/contacts/>

Copyright © 2019 aheadWorks Co. <http://www.aheadworks.com>

Нравится Станьте первыми кому понравится это

Ни одной