

# Public and overloadable methods

## Table of content

- [Public and overloadable methods](#)
  - [AdminTabCore \(AdminTab.php\)](#)
  - [ConfigurationCore \(Configuration.php\)](#)
  - [CookieCore \(Cookie.php\)](#)
  - [DbCore \(Db.php\)](#)
  - [FrontControllerCore \(FrontController.php\)](#)
  - [LinkCore \(Link.php\)](#)
  - [MailCore \(Mail.php\)](#)
  - [ObjectModelCore \(ObjectModel.php\)](#)
  - [PDFCode \(PDF.php\)](#)
  - [ToolsCore \(Tools.php\)](#)
  - [ValidateCore.php](#)

## Public and overloadable methods

With its overriding system, PrestaShop makes it possible to override many of its core functions. But overriding cannot be done if you do not know which method to replace. This list should help you.

You can learn more about overriding PrestaShop by reading the following articles:

- ["Modules, Override, Web Service"](#)
- ["Modules, Classes and Controller Override"](#).

If you do not want to replace a method, but simply add to it, remember to call the parent eponymous method somewhere in your method.

```
class Product extends ProductCore
{
    public function __construct($id_product = NULL, $full = false, $id_lang = NULL)
    {
        echo 'Hello World !';
        parent::__construct($id_product, $full, $id_lang);
    }
}
```

## AdminTabCore (AdminTab.php)

| Method name and parameters  | Description  |
|---|--|
| <code>_childValidation()</code>   | Overload this method for custom checking in <code>validateRules()</code> .       |
| <code>ajaxPreProcess()</code>   | a method called in <code>ajax-tab.php</code> before <code>displayConf()</code> . |
| <code>ajaxProcess()</code>  | the default handle method for request with <code>ajax-tab.php</code> .           |
| <code>copyFromPost(&amp;\$object, \$table)</code>   | Copy datas from <code>\$_POST</code> to object.                                  |
| <code>displayFlags(\$languages, \$defaultLanguage, \$ids, \$id, \$return = false)</code>                    | Display flags in forms for translations.   |
| <code>displayImage(\$id, \$image, \$size, \$id_image = NULL, \$token = NULL, \$disableCache = false)</code> | Display image aside object form.   |
| <code>displayListHeader(\$token = NULL)</code>  | Display list header (filtering, pagination and column names).                    |
| <code>displayTop()</code>   | Overload this method for custom display in <code>displayList()</code> .          |
| <code>displayWarning(\$warn)</code>   | Display a warning message.   |
| <code>getList(\$id_lang, \$orderBy = NULL, \$orderByWay = NULL, \$start = 0, \$limit = NULL)</code>         | Get the current objects' list form the database.                                 |

|  |   |
|--|---|
| I(\$string, \$class = 'AdminTab', \$addslashes = FALSE, \$htmlentities = TRUE) | use translations files to replace english expression.                                       |
| postImage(\$id)  | Overload this method for custom checking in postProcess()'s object creation/update routine. |
| validateRules(\$className = false)   | Manage page display (form, list...).0   |

## ConfigurationCore (Configuration.php)

| Method name and parameters                   | Description  |
|--|--|
| deleteByName(\$key)                          | Delete a configuration key in database (with or without language management).                  |
| get(\$key, \$id_lang = NULL)                 | Get a single configuration value (in one language only).                                       |
| getInt(\$key)                                | Get a single configuration value (in multiple languages).                                      |
| getMultiple(\$keys, \$id_lang = NULL)        | Get several configuration values (in one language only).                                       |
| getMultipleInt(\$keys)                       | Get several configuration values (in multiple languages).                                      |
| getTranslationsFieldsChild()                 | Check then return multilingual fields for database interaction.                                |
| set(\$key, \$values)                         | Set TEMPORARY a single configuration value (in one language only).                             |
| updateValue(\$key, \$values, \$html = false) | Update configuration key and value into database (automatically insert if key does not exist). |

## CookieCore (Cookie.php)

| Method name and parameters      | Description  |
|---------------------------------|--|
| __get(\$key)                    | Magic method wich return cookie data from _content array.                                      |
| __isset(\$key)                  | Magic method which check if key exists in the cookie.  |
| __set(\$key, \$value)           | Magic method wich add data into _content array.  |
| __unset(\$key)                  | Magic method wich delete data into _content array.   |
| getFamily(\$origin)             | Get a family of variables (e.g. "filter_").  |
| isLoggedIn(\$withGuest = false) | Check customer informations saved into cookie and return customer validity.                    |
| isLoggedInBack()                | Check employee informations saved into cookie and return employee validity.                    |
| logout()                        | Delete cookie.   |
| mylogout()                      | Soft logout, delete everything links to the customer but leave there affiliate's informations. |
| setExpire(\$expire)             | Set expiration date.   |
| update(\$nullValues = false)    | Get cookie content.  |
| write()                         | Save cookie with setcookie().  |

## DbCore (Db.php)

| Method name and parameters  | Description  |
|---|--|
| autoExecute(\$table, \$values, \$type, \$where = false, \$limit = false, \$use_cache = 1) | Filter SQL query within a blacklist.                 |
| autoExecuteWithNullValues(\$table, \$values, \$type, \$where = false, \$limit = false)    | Filter SQL query within a blacklist.                 |
| getInstance(\$master = 1)   | Get Db object instance (Singleton).                  |
| nl2br2(\$string)  | Convert \n and \r\n and \r to <br />.                |
| pSQL(\$string, \$htmlOK = false)  | Sanitize data which will be injected into SQL query. |

## FrontControllerCore (FrontController.php)

| Method name and parameters | Description  |
|----------------------------|--|
| preProcess()               | Overload this method for custom display. This runs between the initialization of the page and the display of the header. |
| process()                  | Overload this method for custom display. This runs between the display of the header and the display of the content.     |

## LinkCore (Link.php)

| Method name and parameters  | Description  |
|---|--|
| addSortDetails(\$url, \$orderby, \$orderway)  |  |
| getCategoryLink(\$id_category, \$alias = NULL, \$id_lang = NULL)  |  |
| getCatImageLink(\$name, \$id_category, \$type = null)   |  |
| getCMSCategoryLink(\$id_category, \$alias = NULL, \$id_lang = NULL)   |  |
| getCMSLink(\$cms, \$alias = null, \$ssl = false, \$id_lang = NULL)  |  |
| getCustomLink(\$id_custom, \$page, \$prefix = '~', \$alias = NULL, \$id_lang = NULL)                        |  |
| getImageLink(\$name, \$ids, \$type = NULL)  | Returns a link to a product image for display. Note: the new image filesystem stores product images in subdirectories of img/p/. |
| getLangLink(\$id_lang = NULL)   |  |
| getLanguageLink(\$id_lang)  | Create link after language change, for the change language block.  |
| getManufacturerLink(\$id_manufacturer, \$alias = NULL, \$id_lang = NULL)                                    |  |
| getMediaLink(\$filepath)  |  |
| getPageLink(\$filename, \$ssl = false, \$id_lang = NULL)  |  |
| getPaginationLink(\$type, \$id_object, \$nb = false, \$sort = false, \$pagination = false, \$array = false) |  |
| getProductDeletePictureLink(\$product, \$id_picture)  | This function returns a link to delete a customization picture file.   |
| getProductLink(\$id_product, \$alias = NULL, \$category = NULL, \$ean13 = NULL, \$id_lang = NULL)           | Return the correct link for product/category/supplier/manufacturer.  |
| getSupplierLink(\$id_supplier, \$alias = NULL, \$id_lang = NULL)  |  |
| goPage(\$url, \$p)  |  |
| preloadPageLinks()  |  |

## MailCore (Mail.php)

| Method name and parameters   | Description   |
|--|---|
| l(\$string, \$id_lang = null)  | This method is used to get the translation for email Object. For an object is forbidden to use htmlentities, we have to return a sentence with accents. |
| Send(\$id_lang, \$template, \$subject, \$templateVars, \$to, \$toName = NULL, \$from = NULL, \$fromName = NULL, \$fileAttachment = NULL, \$modeSMTP = NULL, \$templatePath = PS_MAIL_DIR, \$die = false) |   |

|   |  |
|---|--|
| sendMailTest(\$smtpChecked, \$smtpServer, \$content, \$subject, \$type, \$to, \$from, \$smtpLogin, \$smtpPassword, \$smtpPort = 25, \$smtpEncryption) |  |
|---|--|

## ObjectModelCore (ObjectModel.php)

This is the main object from PrestaShop's object model. Any overriding of its methods is bound to influence how all the other classes and methods act. Use carefully.

| Method name and parameters                          | Description  |
|---|--|
| __construct(\$id = NULL, \$id_lang = NULL)          | Build object.  |
| add(\$autodate = true, \$nullValues = false)        | Save current object to database (add or update).       |
| delete()  | Delete current object from database.                   |
| deleteSelection(\$selection)                        | Delete several objects from database.                  |
| getFields()   | Prepare fields for ObjectModel class (add, update).    |
| getValidationRules(\$className = <i>_CLASS_</i> )   | Return object validation rules (fields validity).      |
| save(\$nullValues = false, \$autodate = true)       | Save current object to database (add or update).       |
| toggleStatus()                                      | Toggle object status in database.                      |
| update(\$nullValues = false)                        | Update current object to database.                     |
| validateFields(\$die = true, \$errorReturn = false) | Check for fields validity before database interaction. |

## PDFCode (PDF.php)

| Method name and parameters | Description     |
|----------------------------|-----------------|
| Footer()                   | Invoice footer. |
| Header()                   | Invoice header. |

## ToolsCore (Tools.php)

| Method name and parameters                                     | Description  |
|--|--|
| addCSS(\$css_uri, \$css_media_type = 'all')                    | Add a stylesheet at any time.  |
| addJS(\$js_uri)  | Load a javascript file in the header.  |
| cccCss()   | Combine Compress and Cache (ccc) CSS calls.  |
| cccJS()  | Combine Compress and Cache (ccc) JS calls.   |
| convertBytes(\$value)  | Convert a shorthand byte value from a PHP configuration directive to an integer value. |
| convertPrice(\$price, \$currency = NULL, \$to_currency = true) | Return price converted.  |
| dateDays()   |  |
| dateFormat(\$params, &\$smarty)                                | Display date regarding to language preferences.  |
| dateFrom(\$date)   |  |

|   |  |
|---|--|
| dateMonths()  |  |
| dateTo(\$date)  |  |
| dateYears()   | Generate date form.  |
| deleteDirectory(\$dirname, \$delete_self = true)  | Delete directory and subdirectories.   |
| dieObject(\$object, \$kill = true)  | Display an error with detailed object.   |
| displayDate(\$date, \$id_lang, \$full = false, \$separator = '-')                             | Display date regarding to language preferences.  |
| displayError(\$string = 'Fatal error', \$htmlentities = true)                                 | Display an error according to an error code.   |
| displayPrice(\$price, \$currency = NULL, \$no_utf8 = false)                                   | Return price with currency sign for a given product.   |
| encrypt(\$passwd)   | Encrypt password.  |
| getAdminToken(\$string)   | Encrypt password.  |
| getCurrentUrlProtocolPrefix()   | Get the current url prefix protocol (https/http).  |
| getHomeMetaTags(\$id_lang, \$page_name)   | Get meta tags for a given page.  |
| getHttpHost(\$http = false, \$entities = false)   | Return the <b>current</b> host used, with the protocol (http or https) if \$http is true. This function should not be used to choose http or https domain name. Use Tools::getShopDomain() or Tools::getShopDomainSsl instead. |
| getMetaTags(\$id_lang, \$page_name)   | Get meta tages for a given page.   |
| getPath(\$id_category, \$path = "", \$linkOnTheLastItem = false, \$categoryType = 'products') | Get the user's journey.  |
| getProductsOrder(\$type, \$value = null)  | Get products order field name for queries.   |
| getProtocol(\$use_ssl = null)   | <a href="#">Return the set protocol according to configuration (https).</a>  |
| getRemoteAddr()   | Get the server variable REMOTE_ADDR, or the first ip of HTTP_X_FORWARDED_FOR (when using proxy).   |
| getServerName()   | Get the server variable SERVER_NAME.   |
| getShopDomain(\$http = false, \$entities = false)   | Returns domain name according to configuration and ignoring ssl.   |
| getShopDomainSsl(\$http = false, \$entities = false)  | returns domain name according to configuration and depending on ssl activation.  |
| getToken(\$page = true)   | Get token to prevent CSRF.   |
| getValue(\$key, \$defaultValue = false)   | Get a value from \$_POST / \$_GET. If unavailable, take a default value.   |
| isSubmit(\$submit)  | Check if submit has been posted.   |
| jsonDecode(\$json, \$assoc = false)   | Convert json string to php array / object.   |
| link_rewrite(\$str, \$utf8_decode = false)  | Return the friendly url from the provided string.  |
| passwdGen(\$length = 8)   | Random password generator.   |
| redirect(\$url, \$baseUri = _PS_BASE_URI_)  | Redirect user to another page.   |
| redirectAdmin(\$url)  | Redirect user to another admin page.   |
| redirectLink(\$url)   | Redirect url wich allready PS_BASE_URI.  |
| safeOutput(\$string, \$html = false)  | Sanitize a string.   |
| secureReferrer(\$referrer)  | Secure an URL referrer.  |
| setCookieLanguage()   | Change language in cookie while clicking on a flag.  |
| setCurrency()   | Set cookie currency from POST or default currency.   |

|   |  |
|---|--|
| str2url(\$str)                                    | Return a friendly url made from the provided string. If the mbstring library is available, the output is the same as the js function of the same name. |
| switchLanguage()                                  | Set cookie id_lang.  |
| toCamelCase(\$str, \$capitaliseFirstChar = false) | Translates a string with underscores into camel case (e.g. first_name -> firstName).   |
| truncate(\$str, \$maxLen, \$suffix = '...')       | Truncate strings.  |
| usingSecureMode()                                 | Check if the current page use SSL connection on not.   |
| ZipExtract(\$fromFile, \$toDir)                   | Extract a zip file to the given directory.   |
| ZipTest(\$fromFile)                               | Try to open a zip file in order to check if it's valid.  |

## ValidateCore.php

| Method name and parameters            | Description   |
|---------------------------------------|---|
| IsSortDirection(\$value)              | Check if the value is a sort direction value (DESC/ASC)   |
| isAbsoluteUrl(\$url)                  | Check object validity   |
| isAddress(\$address)                  | Check for a postal address validity   |
| isArrayWithIds(\$ids)                 | isArrayWithIds()  |
| isBirthDate(\$date)                   | Check for birthDate validity  |
| isCarrierName(\$name)                 | Check for a carrier name validity   |
| isCatalogName(\$name)                 | Check for product or category name validity   |
| isCityName(\$city)                    | Check for city name validity  |
| isCleanHtml(\$html)                   | Check for HTML field validity   |
| isColor(\$color)                      | Check object validity   |
| isConfigName(string \$configName)     | Check for configuration key validity  |
| isCookie(mixed \$data)                | Check if \$data is a PrestaShop cookie object   |
| isCoordinate(string \$data)           | Check for Latitude/Longitude  |
| isCountryName(string \$name)          | Check for a country name validity   |
| isDate(string \$date)                 | Check for date validity   |
| isDateFormat(string \$date)           | Check for date format   |
| isDiscountName(string \$discountName) | Check for discount coupon name validity   |
| isDistanceUnit(\$unit)                | isDistanceUnit()  |
| isDniBool(string \$dni)               | Check for Dni validity  |
| isEan13(string \$ean13)               | Check for barcode validity (EAN-13)   |
| isEmail(string \$email)               | Check for e-mail validity   |
| isFileName(string \$name)             | Check for standard name file validity   |
| isGenderIsoCode(string \$isoCode)     | Check for gender code (ISO) validity  |
| isLangIsoCode(string \$iso_code)      | Check for Language Iso Code   |
| isLinkRewrite(string \$link)          | Check for a link (url-rewriting only) validity  |
| isMd5(string \$md5)                   | Check for MD5 string validity   |
| isPasswd(string \$passwd, \$size)     | Check for password validity   |
| isPhoneNumber(string \$phoneNumber)   | Check for phone number validity   |
| isPhpDateFormat(string \$date_format) | Check date formats like <a href="http://php.net/manual/en/function.date.php">http://php.net/manual/en/function.date.php</a> |

|                                    |  |
|------------------------------------|--|
| isPostCode(string \$postcode)      | Check for postal code validity               |
| isSerializedArray(string \$data)   | Check for PHP serialized data                |
| isSha1(string \$sha1)              | Check for SHA1 string validity               |
| isUpc(string \$upc)                | Check for barcode validity (UPC)             |
| isUrl(string \$url)                | Check url validity (disallowed empty string) |
| isUrlOrEmpty(string \$url)         | Check url validity (allowed empty string)    |
| isZipCodeFormat(string \$zip_code) | Check for zip code format validity           |