

Reference Guide To Functions

eDataCatTM

Version 2.0

Table of contents:

Link	3
Account Setup/ View Account Info	4
Login/Logout/Change User	6
List Function	8
Search Function	9
What's New/Most Pop	11
Category	12
Detail/Related Items	13
Adding Items to Cart	14
Add To Cart/Wishlist; Update Cart and View Wish/Cart	15
File Upload	17
Checkout Functions	18
Payment Processing	21
Customer Order Status	23

Link Function

Usage: `user_action=link&link=(name of/path to template)`

This function calls a user defined template and substitutes eDatCat shortcuts and system variables. Note that specific product, category, and order variables are not available. Customer variables are if customer is logged in. Featured products shortcuts are available, as well as category navigation, etc.

Main usage is for main page, entry page for multi-language sites, contact, special groupings of products, file_upload page, and anything else you might come up with.

Example: http://www.edatcat.com/cgi-bin/cgiwrap/EDCstore.pl?user_action=link&link=purchase

Account Setup/ View Account Info Function

Function names: `account_setup`, `viewacct`

These functions display the acct-setup and acct-info templates. These templates can be modified.

Hidden variables required:

- `acct-setup--` `<input type="hidden" name="user_action" value="save_account">`
- `acct-info only--` `<input type="hidden" name="edit_action" value="update">`

Input fields required and checked for by `edatcat`:

- `email`
- `address1`
- `city`
- `state`
- `zip`
- `firstname`
- `lastname`
- `password1`
- `password2`

Optional input fields:

- `midname`
- `hint`
- `address2`
- `country`
- `hphone`
- `nphone`
- `fax`
- `ship_default`
- `whsl_flag`
- `tax_exempt`
- `signon`
- `CUST1`
- `CUST2`
- `CUST3`
- `CUST4`
- `CUST5`

Templates called on completion of `save_account`:

- `account_setup`: `acct-success`
- `viewacct`: `update-success`

Special shortcuts for use on these pages:

- `%%state_list%%`
Displays drop-down of states in tax file. These choices can be modified by modifying the tax table, or hard-coding state options, or using a text input field box.
- `%%country_list%%`
Displays drop-down of countries in country file. Again, these may be modified, or hardcoded, or use a text input field box, or leave out completely.
- `%%ship_checked%%`
Used on `acct-info` to display current setting of `ship_default`.

- %%cookie%% and %%manual%%
Used on acct-info to display current setting of signon.

Notes:

- For additional field checking, javascript may be used.
- Logically, whsl_flag and tax_exempt should be modified by admin, but can be included here.
- The cookie value for signon field is initially set as checked, this may be modified, as can the ship_default.

Login/Logout/Change User

Function Names: `login`, `logout`, `change_user`

These are all similar functions, login being the most complicated and important.

However, with release 2.0, none of these functions are necessary, if you intend to use only the "nologin" option. More on doing this in the checkout section.

Login function

This displays the login.txt template. If the user is set up for signon option as cookie, they do not need to login, they are already logged in. In this case, the login function displays the already-login template. In the event of unsuccessful login, the login.txt is redisplayed with an error message or the customer's login hint where the `%%login_hint%%` shortcut appears. This can be modified in store messages on the admin page. Upon successful login, the login-success page is displayed if a return link has not been specified.

Login template

- hidden variables required on login template:
`<input type="hidden" name="user_action" value="login">`
`<input type="hidden" name="return_link" value="%%return_link%%">`
- input variables required:
`<input type="text" name="userid" value="%%userid%%" size="20">`
`<input type="password" name="lpassword" size="20">`
- Special shortcuts available to login.txt and order_login.txt
`%%login_hint%%` (see above)

Logout function (optional)

Displays login-goodbye template. Simply removes login cookie. Not effective for customers with auto logins.

Change User

Displays login.txt template, even if user is using auto cookies.

Using return_link to bypass login_success page.

The eDatCat script sets return_link to "checkout" during the checkout process. If a customer has to login to finish ordering, they will be taken directly back to the checkout process. The viewacct, account_setup, and ordertrack functions are also setup to do this. If your site does not do this, check for the hidden variable return_link on the order_login template and login template. Also, be sure you are running the most current version of eDatCat. You may also use this technique to return to other pages as well, but it is maybe not worth the trouble. You'll have to pass "return_link" and any other needed parameters to the login function.

Example:

- To return to the list category page, the link to login on the category template would be:
``
- To detail page:
``

- To cart page, the link to login on any template would be:

List Function

Usage: `user_action=list&category=%%urlcategory%%`
or
`user_action=list&category=ALL`
or
`user_action=list&category=hardcoded-category-name`

This function lists all the products in the specified category, using the listpage and listrow templates. Urlcategory is the escaped name of category. Mandatory when categories contain spaces or other special characters.

The resulting HTML can varied. It can be:

- a simple list with links to detailed info about each product.
- a simple list with links to add a single item to the cart and/or wishlist.
- a series of forms to add the product, quantity, selected options, and text input to the cart/wishlist.
- one form to add multi selected items, with quantity and options to the cart/wishlist.

For more details on these techniques, see section on Adding items to cart.

Special shortcuts to use on listpage template.

- Any category field.
- %%categoryname%%
Current immediate part of category name, without upper hierarchy.
- %%resultstatus%% and %%pagestatus%%
Displays numeric results of findings. Customizable in storemessages in the admin menu. See article on customizing this and %%navbar%%
- %%navbar%%
Displays page navigation of results. Customizable in storemessages.
- %%item_rows%% Marks placement of detail rows.

Special shortcuts to use on listrow template:

- all product fields.
- %%resultno%%
Displays the result #.

You can also choose to display results in more than one column by setting this value in Sorting/Listing in admin. For more info see the special instructions in the customization Guide.

Search Function

Usage: `user_action=search`

The search function calls the edatcat product search. Without the parameter "screen=1", `user_action=search` will display the search.txt template. With the parameter, it performs the search and displays the results using searchpage and search_row templates.

The search function looks for an occurrence of the passed string anywhere in the specified field. Catalogno and category is an exception to this! It looks for an exact match. So, if the customer puts an entry in catalogno, and something in description, it looks for an exact match for catalogno.

There are several alternatives for calling the search function.

Default edatcat search.txt

Use the default template for searching--search.txt. This page can be modified to display only the search fields desired. It is called using the `user_action=search`. For more details on modifying, see the contents of the default template.

Using any user template

You can use any user template, and use the link to call the page. The requirements are basically the same as the search.txt template. This might be handy for two different types of searches you might offer your shopper.

Using short form anywhere

You can use just a form on any template, heading, or navbar with the following parameters:

- a) Hidden variables: `<input type="hidden" name="user_action" value="search">` and `<input type="hidden" name="user_action" value="screen=1">`
- b) One or more search parameters. These are normally user input fields, but can be a set hidden variable. The name must be field name to use. Ex.: `<INPUT TYPE="text" NAME="item" SIZE=30>`
- c) Optional is: c) `sortorder=(name of field to sort by)`

Hardcoded Link

You can also use a hardcoded link with specified parameters. This might be used to list all the products manufactured by "Sony". (manuf is a custom database variable.)

example: `See our line of Sony Products`

Advanced Notes:

Searching on category. When searching on category with conjunction with other possible parameters, there is a special shortcut used to display a drop-down of categories and the option of "include all". This label can be modified in store messages.

Searching on price range: Hardcode your dropdown with any combination of price ranges. It must be called "pricerange" and follow a certain format. Be sure to include an all option if using. See example below.

```
<tr><td>Price Range</td>
<td><select name="pricerange" size="4">
```

```
<option selected value="none">All Prices</option>
<option value=".01;5.00">Incredibly Cheap (Under $5)</option>
<option value="5.01;20.00">Real Bargains ($5-$20)</option>
<option value="20.01;50.00">Still a Bargain ($20-$50)</option>
<option value="50.01;100.00">Still in my Budget ($50-$100)</option>
<option value="100.01;200.00">Stretching my Budget ($100-$200)</option>
<option value="200.01;300.00">Good Stuff ($200-$300)</option>
<option value="300.01;9999.99">Must be Art (over $300)</option>
</select></td></tr>
```

What's New/Most Pop

These are special listing functions that work much like list and search.

What's New

Usage: `user_action=whatsnew`
or
`user_action=whatsnew&category=Accessories`

This function will return listing of products that have been added within a certain number of days set in admin. The optional category parameter will narrow listing to the specified category. The processing uses the date-added field in the product record. This can be set on import or through the admin product screen. Note that this uses an index and so the regenerate indexes must be used after import and any product additions/changes concerning date-added.

Templates Used: `newpage` and `newrow`
(see list function for details on templates)

Most Pop

Usage: `user_action=mostpop`
or
`user_action=mostpop&category=Accessories`

This function will return listing of products that have been ordered the most within a certain number of days set in admin. The optional category parameter will narrow listing to the specified category. The processing uses the date-added field in the product record. This can be set on import or through the admin product screen. Note that this uses an index and so the regenerate indexes must be used after import and any product additions/changes concerning date-added.

Templates Used: `newpage` and `newrow`
(see list function for details on templates)

Adding Items to Cart/Wishlist

There are several methods to add items to cart. This section also applies to adding to wishlist, but we will only refer to as "cart".

Using a Link.

Usage: `Add to Cart`

Note: using a link is not appropriate for items with product options. You can also hardcode the catalogno.

Using a form.

For list pages, do not include form tags on the page template, include form tags on the appropriate item row template using unique form names. Include form tags on detail page.

Include the hidden field `user_action` and `catalogno`.

```
<input type="hidden" name="user_action" value="addtocart">
<input type="hidden" name="catalogno" value="%%catalogno%%">
```

Include input fields for quantity and options.

You can add checkboxes for other items, naming the checkboxes "catalogno" and setting the value to the desired catalogno. But beware, that you cannot include input fields such as quantity and options for any of the extra items. To accomplish this, you must use the next technique!

Single form for multiple items

From list pages, or related items on detail page, or custom hard-coded web page.

This technique is used in the R2.0 default listrow template and can be utilized on the detail template when you wish to display related items with input fields for quantity and options.

On the row template you will need:

- `<input type="hidden" name="catalogno" value="%%catalogno%%">`
- `<input type="checkbox" name="add%%catalogno%%" value="YES">Select to Buy Qty:<input type="text" name="quantity" size="3">`
- Plus any dispooption and disptext shortcuts as needed.

On the page template you will need:

- `<input type="hidden" name="user_action" value="addtocart">`
- `<input type="hidden" name="catalogno" value="%%catalogno%%">`

Do not use the shortcut for related items on any row template. It may interfere with the listing process.

On the detail page, if you are adding multi items using related items, **you must alter the templates `relitems` and `relrow`**

Add To Cart/Wishlist; Update Cart and View Wish/Cart

Function names: addtocart, addtowish, updatecart, updatewish, viewcart, viewwish

These functions all call the cartpage, cartrow or wishpage, wishrow or empty_cart, empty_wish templates.

Addto.. will add the specified catalogno, quantity, and options to the appropriate list. For more info on this function, see section on "Adding items to the cart."

Updatecart/updatewish is a function called from the list and must be called from a form. It either deletes, or moves the items selected.

Important note, that this is new to R2.0. The old style links in 1.5 (editcart/editwish) are no longer used. The update process should have installed the above templates over your existing ones! Viewcart and viewwish simply call these templates, without any additional processing. If the list is empty, the empty_cart/wish template is displayed.

Cart and wishlist templates.

If you will be allowing updates to the list, which you probably will, you will need to start the form on cartpage/wishpage with these tags:

- `<form action="%%script_loc%%" method="POST" align="center">`
- `<input type="hidden" name="user_action" value="updatecart">` (updatewish for wishlist)

then end it, again on cartpage/wishpage with your submit button (or image):

- `<input type="submit" name="Update Price" value="Update Price">`
(The name and value are unimportant to processing)

On the row templates, you will need these tags to complete the form operability:

- `<input type="checkbox" value="YES" name="delete%%itemno%%">Delete Item
`
- `<input type="checkbox" value="YES" name="move%%itemno%%">Move to Wishlist` (or
Cart)
- `<input type="text" size="3" name="%%catalogno%%" value="%%quantity%%">`

All of the above are necessary, except the checkbox to move to wishlist. Note: if you do not wish to use wishlists, be sure to remove this checkbox!

Shortcuts to use on row templates:

- `%%seloptions%%`
Displays all selected options
- `%%selttext%%`
Displays all text options. If you do not like the way these format your options, you can place each selected option individually. Use `%%selopt1%%,etc.` and `%%text1%%,etc.`
- `%%price%%`
Displays discounted price.
- `%%discount%%`
Displays the discount on that item.

- %%ext_price%%
Displays price times quantity, PLUS any setup or onetime charges. Plus, any product variable such as status, free-freight, or setup.

Shortcuts to use on page templates:

- Any customer variables, if customer is logged in.
- %%tot_price%%
Displays total extended price.
- %%tot_discount%%
Displays store discount. (Item discount is already included in tot_price)
- %%grand_total%%
Displays (\$tot_price-\$tot_discount) Freight is not calculated at this time.

File Upload Function

To utilize the new file upload function, first use link to link to a specific page for file upload or use multi-part form and submit button on existing template.

On the multi-part form, the file input field is named "upload_file". This hidden variable is required:
<input type="hidden" name="user_action" value="file_upload">

Optional hidden fields: With Release 2.0d (Dec. 1)

- return_loc=(URL of return location) to go to at completion of download. If return_loc is omitted, script will display the upload_success template.
- The uploaded file will use save directory and max file size, specified in admin menu. fileupl_dir and maxuplsize: will override admin settings.

The uploaded file will be saved in save directory using sessionid as file name. In the event one already exists, the file will be saved with a letter appendix. A limit of 49 files per sessionid can be uploaded.

Default templates: fileupload and upload_success

Check Out Functions

Although there are many functions related to checking out, all of them are initiated using the function name checkout

Overview of process.

A. When the checkout function is called, the edatcat script determines if the customer is logged in or not. If not, the **order_login** screen is displayed. (See information on login template for more info on this template. In addition, the order_login screen presents 3 methods of continuing with the order. These are designated by the hidden value, "user", and can actually be placed as shortcuts on merchant email confirmation.

1. The user has an existing account and wishes to login at this time. user="returning"
2. The user needs to set up an account at this time. user="new"
3. The user doesn't want to set up an account at all. user="nologin"

The merchant can choose to eliminate either #3, and force customer registration, by eliminating that checkbox and prompt.

Or, the merchant can choose to eliminate accounts entirely. To accomplish this, all links to login and account functions should be removed. Customer order status would not be operative, so all links to that function should be removed. To further streamline the checkout process, the order_login can be bypassed by calling the checkout function by passing user as a parameter. Example: `` Form action would require the two as hidden fields.

B. The next screen is the billing screen which uses the template, **billing.txt**
This template accepts billing information. See account setup/account info for more details, since this template works much like the acct_info.

Important differences are required hidden variables:

```
<input type="hidden" name="user_action" value="checkout">
<input type="hidden" name="user" value="%%user%%">
```

Another important difference is the use_billing input field. This gets its default value from the ship_default customer field. It can be overridden for this order, but won't be saved as the default. The shortcuts, %%yeschecked%% and %%nochecked%% place the checked status on the template. Merchant can force input of shipping address, by simply removing this checkbox and supplying the hidden field use_billing with value set at "yes"

User Order Fields: At this point, the user order fields (USER1,USER2, etc.) can be collected and will be stored with order info. You might want to include input fields for extra shipping instructions or other comments here.

C. If use_billing is "yes", the next screen is the shipping screen, template **shipping.txt**. Billing fields are brought over to the shipping template, where the customer can overwrite the values with correct shipping name and address. Again this template shares much in common with the account_setup template.

Important differences are required hidden variables:

```
<input type="hidden" name="user_action" value="shipping">
<input type="hidden" name="user" value="%%user%%">
<input type="hidden" name="email" value="%%email%%">
```

When passing the USER fields from the billing template, include them as hidden variables. Note: these are not in the default template.

Example: `<input type="hidden" name="USER1" value="%%USER1%%">` etc.

Again, these fields can be collected on the shipping template as well. Maybe to provide an extra line in shipping address.

D. After billing and shipping info has been collected, the order is saved in a temporary text file in the carts directory. Should it ever be needed for "lost" details. The file to look for will have an "O" prefix followed by invoice #. Also at this time, the following temporary files are generated and saved in the cart directory. They are deleted upon payment completion.

- customer email: using **econfirm** and **emailrow**
- merchant email: using **mconfirm** and **emailrow**
- order record for SSL site: using **orderrec** and **emailrow**
- final confirmation page: **confirm_page** and **order_rows2** (if it exists, otherwise **order_rows**) (although this page is generated on the regular site, it will be displayed on SSL site, so should be coded accordingly. Variables available are the same for the checkout template below.

Note: **orderrec** and email templates work best as text, not HTML code.

E. Last screen before payment processing is the confirm order screen. Templates used are: **checkout** and **order_rows**. These templates are very similar to cartpage and cartrow.

An important difference is there is no update order rows function, as in the cart.

Required Elements:

`%%payaction%%` and `%%payfields%%`

Generates hidden fields and "post" action necessary to initiate payment process. If using "other" payment processing, you will need to insert these fields manually.

There are articles available for Worldpay and Paypal implementation, and instructions for combining methods.

UPS shipping options:

You will need to include a form with these elements to allow, your customer an option for UPS shipping types. If you use only UPS Ground, you can eliminate this form.

```
<form action="%%script_loc%%" method="post">
<input type="hidden" name="user_action" value="checkout">
Here are your shipping options:<br>
<small>%%ups_shipping_opts%%</small><br>
<small><input type="submit" name="update" value="Update Shipping"> </small>
</form>
```

USER1, etc. can also be collected on this template, and they will be passed on to the payment screen. Also, the option to send email confirmation to the customer can be checked or hard-coded here. The variable is `sendemail` and values are YES or NO. If no further payment processing will take place, this would be the template to give the customer that option. Also, if you want to give an option of various off-line payment methods, the variable `paytype` can be included here.

Any other variable may be either collected in input fields or passed on as hidden fields to the payment template and used as shortcuts on final confirmation screen and email confirmation. Among these is "user" which will reflect choice made on login screen.

example: `<input type="hidden" name="user" value="%%user%%">`

Variables available on order_row template:

- Same variables available on cart record.

Variables available on checkout template:

- invno, invdate, invuserid, user
- shipping address: shaddress1, shaddress2, shcity, shstate, shzip, shcountry
- shipping name: shfirstname, shlastname, shmidname
- all customer fields
- ups_packages, ups_type
- total_weight, tot_price, tot_discount, tax_total, vat_tax,
- shipping_total, tot_item_shipping, handling_total
- grand_total

Payment Processing

The %%payaction%% shortcut on the checkout template determines where payment processing will take place. Or in the case of other types of payment, the hard-coded post link initiates the process. Here is an overview of the different types of payment processing selected in admin settings.

No SSL Option Utilized

No payment processing

For checks, COD, invoicing, or phone orders where no on-line payment processing is required, Edatcat links directly to the XXXchkout script where the order transaction is completed. (see below). In the checkout template, paytype can be set to any descriptive value for inclusion in shortcuts, and saved in the order database for further reference.

Payment Processing Services

With this configuration, Edatcat links directly to the payment processing service. Some payment processing services, like authorizenet and payflow link, will “post” back to a specified script and send a response code of whether the transaction was successful, before displaying it’s own confirmation page. This ensures that the edatcat transaction is completed. When you select one of these specific services, edatcat automatically forms the parameters needed and processes this response. The transaction is completed as below.

Note that some payment processing services do not provide for a “forced” post and rely on the customer to hit the return button, for the edatcat transaction to be completed. It is quite possible that the customer might not complete this step. In this case, the merchant will need to recover details of the order from the cart directory. Or contact the customer.

SSL Options

When merchant is collecting credit card or secure payment information on their own SSL site, the checkout script calls the XXXSSL script using the SSL address. This script displays some form of the payment template.

Local

Local SSL sites reside physically on the same server where the regular site resides. Data paths, templates, scripts and images are found in the same location as the regular site.

Most, if not all, of the same shortcuts are available from a local SSL site, as is from the regular. But being a secure site, the addresses of the template elements, should also be secure to avoid browser warnings to the customer.

Remote

Remote secure site does not mean a remote processing service! It resides on a completely different server in service for the merchant's use. Processing for remote is slightly more limited, because the edatcat databases are not available. Many shortcuts are not available. You will need to design your payment template and and SSL version of carterror accordingly.

No Payment Processing

This processing option is chosen when the merchant will be collecting credit card information on the secure site and processing them in house. For this option, the payment.txt template is used. For the required elements on payment tmlplate, please refer to the default template. The payment template for this option, returns to the SSL script to verify card info and if valid, completes the transaction as below.

Making modifications to payment.txt:

You may want to update the dropdown for expired year, simply add/remove select options. You may

want to add/remove paytype choices. There is nothing magical about paytype values, except for credit card, where you must specify "card" as the value to initiate cc-card checking. To add a paytype of COD, simply give the checkbox a value of "COD". To display the checked value in case the screen is re-displayed use the shortcut: %%COD_checked%% Although this is not likely to happen unless custom programming is added to your script. You may wish to take the checkbox off for email option.

Shortcuts/variables concerning payment info:

- %%paytype%%
- %%cc_number%%
- %%cc_name%%
- %%cardtype%%
- %%exp_month%%
- %%exp_year%%
- %%cclist%%
Displays dropdown of available cards from selection in admin settings
- %%card_error%%
Displays appropriate error message for invalid card input. These messages can be modified in store messages.
- %%paytext%%
Formats credit card info. This is not customizable, therefore, you can use the individual shortcuts above.

Payment Processing Services

This processing option is chosen when the merchant will be collecting credit card information on the secure site and then sending that info on to a processing service like Authorizenet. For this option, please refer to the specific documentation for setting up payment templates. The payment processing service should return response to the checkout script, not the SSL script.

Order transaction Completion.

The temporary order file in the cart directory is saved to the invoice/order database. For SSL sites, payment information shortcuts are completed.

Email confirmation to merchant and customer is sent, temporary files in cart directory are deleted, and customer's shopping cart and sessionid cookie is deleted. Final confirmation screen is displayed.

Custom checkout processing: Just before temporary files are deleted, the edatcat script checks for the existence of XXXcustom.pl (or .cgi). If it exists, it will perform a subroutine called checkout_proc in that script file. This script file must be a library, not a stand-alone script.

Customer Order Status

Function name: orderstat (alias ordertrack)

Link to call function: %%script_loc%%?user_action=orderstat
or from customer email confirmation:
%%script_loc%%?user_action=orderstat&userid=%%userid%%

This function calls login function if not already logged in. Using return_link (see login) it will bypass the account_success screen. The function then displays list of orders using the templates, statuspage and ordstatow. Version 2.0d (Dec. 2001) displays order detail using template orddetrow.

Variables available for statuspage:

- All customer database fields.
- %%item_rows%%
Specifies placement for order header detail.

Variables available for ordstatow:

- All invoice database header fields.
- %%track_button%%
Displays a tracking button for UPS tracking. Button can be modified in admin menu under confirmation/tracking.
- %%detail_rows%%
Specifies placement for order detail row.

Variables available for orddetrow.

- All invoice item fields.
(Similar to cart and checkout page)
- All product fields.