



The Handbook

Codex

http://codex.wordpress.org/Author_Templates

Version Date

6 August 2005

Author Templates

WordPress allows creation of a custom Author Template for your [WordPress Theme](#). You can put information on this template file to display information about authors on the blog. All authors will use the same `author.php` template.

Using Author Template

The template, `author.php` is defined within the [Template Hierarchy](#). It is used when an author is queried. When a user clicks an author's link in their name on a page, WordPress automatically routes them to the `author.php` template file, if one is supplied. Otherwise the `index.php` will be used to fetch the list of other posts by that author. Using a custom author's template file will allow customizing the information presented.

Linking to the Author Template Page

The `author.php` won't know what data to display if you don't pass the query to it. You need to link to the `author.php`, telling it which author to display. Here are some examples of such links that query the template and specify an author:

```
?author=2
?author_name=joe
/author/joe/
```

These are requesting the author page, and telling it "give me data for Harriett".

To generate these links you can use [Template Tags](#) which list authors. See: [wp_list_authors\(\)](#) and [list_authors\(\)](#).

These links can be placed, for example, in an custom template, archive list, or sidebar. Here is an example using [wp_list_authors\(\)](#):

```
<h2>List of authors:</h2>
<ul>
<?php wp_list_authors(); ?>
</ul>
```

This will generate a list of authors, and a link to each author's page in your sidebar which would look something like:

List of Authors:

- [Joe](#)
- [Hortensia](#)
- [Mildred](#)
- [Petula](#)

Custom Author Template

Setting the Value of Current Author

This code below sets the value for a variable called `$curauth` (Current Author) which we will use in the template to indicate what data to display. This first bit of code goes INTO the `author.php` template file. It pulls in the author ID or name, and assigns that value to `$curauth`:

```
<?php
if(isset($_GET['author_name'])) :
$curauth = get_userdata_by_login($author_name);
else :
$curauth = get_userdata(intval($author));
endif;
?>
```

There are other ways to receive the query and assign the value of `$curauth`, if the above does not work for you, also try this, which will work in Word Press 1.2

```
<?php
if(isset($_GET['author_name'])) :
$curauth = get_userdata_by_login($_GET['author_name']);
else :
$curauth = get_userdata($_GET['author']);
endif;
?>
```

Or the purely 1.5 way:

```
<?php
if(get_query_var('author_name')) :
$curauth = get_userdata_by_login(get_query_var('author_name'));
endif;
```

```

else :
$curauth = get_userdata(get_query_var('author'));
endif;
?>

```

Using Current Author Outside of the Loop

Once you have the `$curauth` value, you can use it to get other data from the author profile. You must use it outside of the Loop. For example:

```
<p>This is <?php echo $curauth->user_nickname; ?>'s page</p>
```

Will say:

This is Joe's page

Listing Entries via The Loop

By using the loop on the `author.php`, you can list all entries by the queried author. This will list all posts from the queried author:

- The title in the `h3`
- The date
- The content and a read more link
- A comments link
- And if there are no posts, it will say so

The Loop code looks like this:

```

<?php if (have_posts()) : ?>
    <?php while (have_posts()) : the_post(); ?>
        <h3>
            <a href="<?php the_permalink() ?>"><?php the_title(); ?></a></h3>
            <small><?php the_time('F jS, Y') ?> </small>
            <?php the_content('Read the rest of this entry »'); ?>
        <p>
<?php comments_popup_link('No Comments »', '1 Comment »', '% Comments »'); ?>
</p>

    <?php endwhile; ?>

    <?php else : ?>
        <p>No posts by this author</p>
    <?php endif; ?>

```

About Current Author

It is important to know that `$curauth` is not a WordPress variable, but a variable generated by PHP code and sent via a link as a **query**. Therefore, the `author.php` file

must be requested via a link to the author page, which specifies an author. If you request the `author.php` page directly, you'll end up with no data.

Here is all the user data you can access through `$curauth`:

- `$curauth->user_aim;`
- `$curauth->user_email;`
- `$curauth->user_firstname;`
- `$curauth->user_icq;`
- `$curauth->user_lastname;`
- `$curauth->user_level;`
- `$curauth->user_login;`
- `$curauth->user_msn;`
- `$curauth->user_nickname;`
- `$curauth->user_description;`
- `$curauth->user_url;`
- `$curauth->user_yim;`

Here are examples of how it would be used on the `author.php` page:

- `<?php echo $curauth->user_nickname; ?>` - Displays the user's nickname.
- `<?php echo $curauth->user_description; ?>` - Displays the user's profile information (as filled in from their user account.)

Custom Author Templates Examples

The following is a list of instructional examples for linking to and creating custom author templates.

Linking to Author Page from Sidebar

To list authors in your sidebar, you need to create a list of author links. When the visitor clicks on an author's name, it brings up a page listing all posts by that author. The list will have a link to post, show the date and show the category of post. This example will work with the default WordPress theme.

First, generate a list of authors and links in the `sidebar.php` template file, then prepare the `author.php` to display the list of posts.

1. Open `sidebar.php` in a [text editor](#).

We can generate a list of authors, and links by using the default settings available from [wp_list_authors\(\)](#). **Note:** You could add this to any other page, such as the `archives.php`.

```
<h2>List of authors:</h2>
<ul>
<?php wp_list_authors(); ?>
</ul>
```

For more information, see [wp_list_authors\(\)](#) and [list_authors\(\)](#).

2. Create a new blank text file, and save it as `author.php` in your Theme directory. Paste the following code into the new `author.php` file (this will include the

Theme's header, sidebar and footer):

```
<?php get_header(); ?>
<div id="content" class="narrowcolumn">
    <!-- This sets the $curauth variable -->
    <?php
    if(isset($_GET['author_name'])) :
    $curauth = get_userdata_by_login($author_name);
    else :
    $curauth = get_userdata(intval($author));
    endif;
    ?>

    <h2>Posts by <?php echo $curauth->user_nickname; ?>:</h2>

    <ul>
    <!-- The Loop -->
    <?php if ( have_posts() ) : while ( have_posts() ) : the_post(); ?>
        <li>
        <a href="<?php the_permalink() ?>" rel="bookmark" title="Permanent Link:
        <?php the_title(); ?>">
        <?php the_title(); ?></a>,
        <?php the_time('d M Y'); ?> in <?php the_category('&'); ?>
        </li>

        <?php endwhile; else: ?>
            <p><?php _e('No posts by this author.');
        <?php endif; ?>
    <!-- End Loop -->
    </ul>
</div>
<?php get_sidebar(); ?>
<?php get_footer(); ?>
```

Including Profile Information

You can include more profile information, outside of [The Loop](#), using the variables described above for \$curauth. For example, above the loop which displays the lists of posts add:

```
<h2>About: <?php echo $curauth->user_nickname; ?></h2>
<dl>
<dt>Website</dt>
<dd><a href="<?php echo $curauth->user_url; ?>"><?php echo $curauth->
user_url; ?></a></dd>
<dt>Profile</dt>
<dd><?php echo $curauth->user_description; ?></dd>
</dl>
```

Linking Options

You may want to change the way the list of authors appears, using the arguments in [wp_list_authors\(\)](#). For example, the admin account is excluded by default. You can include the admin account this way:

```
<ul>
<?php wp_list_authors('exclude_admin=0'); ?>
</ul>
```

Also, you can combine arguments. By default, authors without posts are eliminated, but you may want to link to their profiles anyway. So we can include admins, as above, and also include authors with no posts.

```
<ul>
<?php wp_list_authors('exclude_admin=0&hide_empty=0'); ?>
</ul>
```

Author Data on Other Templates

You can include author data on other templates.

For example, if you use the Template Tag, [the_author_posts_link\(\)](#), from within The Loop, clicking on it will bring the user to a list of all posts by a specified author. This points to the `archive.php`, *unless* you have `author.php` installed.

Therefore, you can choose to include the author data on an [Archive Index](#). You can have it read, for example: "Harriett Smith's page of posts". So, you need the same `$curauth` code at the top to pull in the current author data; and you'd need to print the page title.

But, if your Archive Index is used for different purposes, you'd want to check first: is this archive being access from an author link? To do this, you can use an `if()` statement to check if the author data was passed. Then you can assign the variable, and print `$curauth` data where you like. This code below will accomplish that for you.

```
<!-- check if author is passed (if it was an author link to archive), then
define the $curauth variable -->

<?php if(isset($_GET['author_name']) || isset($_GET['author'])) :
if($_GET['author_name']) :
$curauth = get_userdata_by_login($_GET['author_name']);
else :
$curauth = get_userdata($_GET['author']);
endif;
?>

<!-- print the page title, including author's name, only if above check is true
-->

<h3>You're currently viewing <?php echo $curauth->user_firstname; ?> <?php echo
$curauth->user_lastname; ?>'s page of posts:</h3>

<?php endif; ?>
```

References

- [Get Author Profile Plugin](http://guff.szub.net/get-author-profile|Kafkaesqui) (<http://guff.szub.net/get-author-profile|Kafkaesqui>) will also access the author information and allows you to provide author information outside of the Loop, such as for a sidebar intro of the blog owner or to list contributors to your blog.
- [WordPress Support Forum Thread on Author Templates](http://wordpress.org/support/topic/27554#post-154787) (<http://wordpress.org/support/topic/27554#post-154787>)
- [Another WordPress Support Forum Thread on Author Templates](#)