

# Retrieving WordPress Posts

## TechRounder PDF Edition

Live article: <https://www.techrounder.com/webtips/retrieving-wordpress-posts/>

---

By Vipin PG | Published May 8, 2012 | Updated January 4, 2026 | Format: Article | 2 min read

### In brief

WordPress posts can be retrieved by using the function `query_posts()`. Different posts can be displayed on different pages, even based on different categories, by using this function.

WordPress posts can be retrieved by using the function `query_posts()`. Different posts can be displayed on different pages, even based on different categories, by using this function.

For example, on the homepage, you may see the latest ten posts. If you want to show only five posts, it can be done using `query_posts()`

```
query_posts( 'posts_per_page=5' );
```

If you use `query_posts` within a template page, WordPress will have already executed the database query and retrieved the records by the time it gets to your template page (that's how it knew which template page to serve up!). So when you over-ride the default query with `query_posts()`, you're essentially throwing away the default query and its results and re-executing another query against the database.

This is not necessarily a problem, especially if you're dealing with a smaller blog-based site. However, developers of large sites with big databases and heavy visitor traffic may wish to consider alternatives, such as modifying the default request directly (before it's called). The request filter can be used to achieve precisely this.

The `'parse_query'` and the `'pre_get_posts'` filters are also available to modify the internal `$query` object used to generate the SQL to query the database.

```
// The Query query_posts( $args ); // The Loop while ( have_posts() ) : the_post(); the_title(); endwhile;
// Reset Query wp_reset_query();
```

Here parameters are combined with an ampersand (&), like so:

```
'query_posts( 'cat=3&year=2004' );'
```

Posts for category 13, for the current month on the main page:

```
'if ( is_home() ) { query_posts( $query_string . '&cat=13&monthnum=' . date( 'n', current_time(
'timestamp' ) ) ); }'
```

this combination will return posts belong to both Category 1 AND 3, showing just two posts, in descending order by the title:

```
'query_posts( array( 'category__and' => array(1,3), 'posts_per_page' => 2, 'orderby' => 'title', 'order' =>
'DESC' ) );'
```

The following returns all posts that belong to category 1 and are tagged "apples"

```
'query_posts( 'cat=1&tag=apples' );'
```

You can search for several tags using + `'query_posts( 'cat=1&tag=apples+apples' );'`