### Debugger Console

**See also** [Kint](#)

<table>
<thead>
<tr>
<th>Rating</th>
<th>Subject</th>
<th>Submitted by</th>
<th>Importance</th>
<th>Easy to solve?</th>
<th>Priority</th>
<th>Category</th>
<th>Volunteered to solve</th>
<th>Created</th>
<th>LastModif</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>★★☆☆☆</td>
<td>smarty (debug) function gives PHP error &quot;E_USER_WARNING&quot;</td>
<td>sinnlos</td>
<td>7</td>
<td>35</td>
<td>• Error</td>
<td>2009-06-26</td>
<td>2009-08-18</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>★★☆☆☆</td>
<td>debugger console breaks the admin panels</td>
<td>pkdille</td>
<td>25</td>
<td>• Error</td>
<td>2010-02-13</td>
<td>2010-05-06</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>★★☆☆☆</td>
<td>Debugger console no longer appears</td>
<td>luciash d'</td>
<td>7</td>
<td>35</td>
<td>• Regression</td>
<td>2014-01-29</td>
<td>2014-01-30</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>★★☆☆☆</td>
<td>Bad contrast for debug console using the default theme</td>
<td>Bernard Sfez / Tiki Specialist</td>
<td>4</td>
<td>10 easy</td>
<td>• Usability</td>
<td>2018-11-25</td>
<td>2018-11-28</td>
<td>2</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Also see `{debug}` which can be used in Smarty Templates as described in [template tricks](#).

**General Description**

For a TikiAdmin this tool is a great first step towards making Tiki as easy to debug as it is to extend and modify!

**Overview**

When processing a page, your code can send output to the debug console with commands such as `$debugger->msg("your message")`. This writes the output to the debug console which you can then view by...
displaying the console.

When executing debugging commands, the page is re-loaded with the output of the debugging commands written to the debugconsole. This may pose problems for pages which are processing user input, uploading files, or that crash and do not return.

The debug console is embedded within a div class="debugconsole" id="debugconsole" which is turned on by clicking on the link to toggle the console. Careful inspection will note that the footer.tpl includes tiki-debug_console.php which handles the processing after most of the rest of the page has already executed.

General Notes

- Enable debugger console by selecting checkbox on Admin -> Features -> Administration Features.
- Debugger can be extended with commands and/or interface extensions (tabs).
- Every command in debugger is a PHP class.
- Sources lib/debug
- Search for debug-command_*.php it is filenames mask... for debugger commands
- one command one class one file

Key Function and sub-features

**General info**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>*help</td>
<td>Display list of commands or help for specified command (help print for example)</td>
</tr>
<tr>
<td>*tikitables</td>
<td>Show list of Tiki tables in DB schema</td>
</tr>
<tr>
<td>*features</td>
<td>Show features on/off state</td>
</tr>
<tr>
<td>*perm</td>
<td>Show current permissions in a convenient way</td>
</tr>
<tr>
<td>*print</td>
<td>Print PHP variable. Indexes are OK.</td>
</tr>
<tr>
<td>*slist</td>
<td>Display list of Smarty variables. Better to specify partial name or very long list of vars will return.</td>
</tr>
<tr>
<td>*sprint</td>
<td>Print Smarty variable</td>
</tr>
<tr>
<td>*watch</td>
<td>Manage variables watch list</td>
</tr>
<tr>
<td>*sql</td>
<td>Exec SQL query on Tiki DB</td>
</tr>
</tbody>
</table>

How to extend debugger

Look for example skeleton lib/debug/debugger-command_test.php

Related Links

Typical Uses

**Debug messages on console**

It is possible to print debug info to special debug tab called 'Debug messages' during page generation. It is possible from PHP and Smarty as well.

Dump Smarty variable from template:
```
{var_dump var=left_modules}
```

Dump Smarty variable from template using modificator:
```
{$description|str_replace:"\n":"<br "/"|dbg}
```

Of course modificator `dbg` can be used more than once:
```
{$description|dbg|str_replace:"\n":"<br "/"|dbg}
```
... so you will get 2 messages before and after modification of $description.

```php
// PHP code
global $debugger;
// Print message
$debugger->msg('Value of $myvar = "'.$myvar.'"');
// Print variable
$debugger->var_dump('$myvar');
```

---

**Case Studies**

---

**Bugs**

Non command-line commands should not print help. — Fixed in CVS.

**Support Requests**

We need more examples of the commands that are being passed here...

---

**TikiTeam**

zaufi

---

For more information

lookup sources lib/debug

---

**Related links**

- [Debugging tools deployable within Tiki](#)

---

**alias**

- [TikiDebuggerConsole](#)