Zend_View_Helper_IndentLines - Jon Whitcraft

<ac:macro ac:name="unmigrated-inline-wiki-markup"><ac:plain-text-body><![CDATA[

<ac:macro ac:name="unmigrated-inline-wiki-markup"><ac:plain-text-body><![CDATA[

Zend Framework: Zend_View_Helper_IndentLines Component Proposal

Proposed Component Name	Zend_View_Helper_IndentLines
Developer Notes	http://framework.zend.com/wiki/display/ZFDEV/Zend_View_Helper_IndentLines
Proposers	Jon Whitcraft
Zend Liaison	TBD
Revision	1.0 - 6 Feb 2009: Initial Draft. (wiki revision: 11)

Table of Contents

1. Overview
2. References
3. Component Requirements, Constraints, and Acceptance Criteria
4. Dependencies on Other Framework Components
5. Theory of Operation
6. Milestones / Tasks
7. Class Index
8. Use Cases
9. Class Skeletons

1. Overview

Zend_View_Helper_IndentLines

2. References

3. Component Requirements, Constraints, and Acceptance Criteria

- This helper **will** add whitespace to each line of anything passed to it.
- This helper **will** be extendable to allow you to easily add this functionality to other view helpers

4. Dependencies on Other Framework Components

- Zend_View_Exception

5. Theory of Operation

6. Milestones / Tasks

- Milestone 1: design notes will be published here
- Milestone 2: Working prototype checked into the incubator supporting use cases
- Milestone 4: Unit tests exist, work, and are checked into SVN.
- Milestone 5: Initial documentation exists.

7. Class Index

- Zend_View_Helper_IndentLines

8. Use Cases

UC-01

```
$content = 'Hello
World';

print $this->indentLines(4, $content);

// outputs
    Hello
    World
```

9. Class Skeletons

```php
<?php

class Zend_View_Helper_IndentLines extends Zend_View_Helper_Abstract
{

    protected $_indentCharacter = ' ';

    protected $_endOfLineCharacter = PHP_EOL;

    public function indentLines($indentLength, $content)
    {
        if(func_num_args() == 0) {
            return $this;
        }

        $lines = preg_split('/\r\n|\r|\n/', $content);

        $text = '';
        $indent = str_repeat($this->_indentCharacter, $indentLength)
        foreach($lines as $line) {
            $text .= $indent . $line . $_endOfLineCharacter;
        }

        return $text;
    }

    public function setIndentCharacter($newCharacter)
    {
        $this->_indentCharacter = $newCharager;

        return $this;
    }

    public function getIndentCharacter()
    {
        return $this->_indentCharacter;
    }

    public function setEndOfLineCharacter($newCharacter)
    {
        $this->_endOfLineCharacter = $newCharacter;

        return $this;
    }

    public function getEndOfLineCharacter()
    {
        return $this->_endOfLineCharacter;
    }
}
```

]]></ac:plain-text-body></ac:macro>
]]></ac:plain-text-body></ac:macro>