
Subject: Re: Validation plug-ins
Posted by [janalwin](#) on Thu, 07 Sep 2006 16:05:05 GMT
[View Forum Message](#) <> [Reply to Message](#)

I like option 2 the best. But is it really needed to have more than one validation method in a custom validation class? You could put every custom validation in a separate class, load this class and call the standard method that all custom validation classes should have. (for example validate())

Every validator could extend this class for example:

```
class custom_validator {  
  
    var $errors  
  
    function validate($fieldvalue) {  
  
        return;  
    } // end method  
  
    function getErrors() {  
        return $this->errors  
    } // end method  
} // end class
```

Examples of implementations

```
class dutch_postcode_validator extends validator {  
  
    function validate($fieldvalue) {  
        // check  
        if (eregi..... {  
            $this->errors[]="The postcode may only contain... "  
        } //end if  
  
    } // end method  
  
} // end class
```

```
class dutch_phonenumber_validator extends validator {  
  
    function validate($fieldvalue) {  
        // check  
        if (eregi..... {  
            $this->errors[]="A phonenumber can only contain numbers"  
        } // end if  
  
    } // end method  
  
} // end class
```

```
        if(strlen($fieldvalue)!=10) {
            $this->errors[]="A phonenumber must contain 10 characters"
        } // end if
    } // end method

} // end class
```

An other way to do it would be to define an interface, but it's only possible in PHP5 so that not an option I think.

I'm relatively new to object oriented programming. But this seems the most flexible method to me.

The indents in the code seem to get lost when I'm posting. Sorry

--

Jan Alwin