

For some reason, you work on a project with a lot of PHP files and, by "chance", all the php files are in the same folder. The application does not seem to work properly due to PHP error. So the easiest way to find which files are in trouble is by calling the php client with the "-l" option. The problem is there is way too many files.

The solution is very simple:

```
for x in `ls *.php`; do php -l $x; done;
```

Let's say now you would like to perform the same operation in a SVN environment to make sure the modified or added files are valid.

First, go in the project folder:

```
cd /var/www/project_y/trunk/
```

* or can be any other folder

Get the list of modified or added files:

```
svn status
```

That will output something like:

```
M      class/Main.class.php
M      action.php
A      access.php
```

Now, unfortunately you cannot perform the `for x in `svn status`; do php -l $x; done;`. The result will output an error saying: the file M cannot be found or open.

```
M
Could not open input file: M
action.php
No syntax errors detected in action.php
```

To fix that problem, we need to use the sed command to remove the extra characters we don't really need.

```
svn status | sed -e 's/M      //g' | sed -e 's/A      //g'
```

The output result will look something like:

```
class/Main.class.php  
action.php  
access.php
```

Now, we can use the for command to validate the modified or added file:

```
for x in `svn status | sed -e 's/M      //g' | sed -e 's/A      //g'`; do php -l $x; done;
```

The output result will look something like:

```
No syntax errors detected in class/Main.class.php  
No syntax errors detected in action.php  
No syntax errors detected in access.php
```