

# PHP Code Sniffer in PhpStorm



## Redirection Notice

This page will redirect to <https://www.jetbrains.com/help/phpstorm/using-php-code-sniffer-tool.html> in about 2 seconds.

[Tweet](#)

In addition to the coding assistance and inspections provided by PhpStorm, extra checks on code style can be performed using the [PHP Code Sniffer](#) (PHPCS) tool integration. In this tutorial, we'll have a look at how we can validate our code for consistency with a coding standard of choice, and how PHP Code Sniffer integrates with PhpStorm.

In PhpStorm, we can configure code style through Settings/Preferences | Editor | Inspections | PHP | Code Sniffer and use one of the many available schemes. There are also various inspections that check our code against the code style. We can add additional code style inspections to PhpStorm using tools like PHP Code Sniffer. PHP Code Sniffer is a PHP5 script that "sniffs" PHP, JavaScript and CSS files to detect violations of a defined coding standard. By default, it will warn us whenever violations are encountered against the PEAR coding standards, but we can also specify other coding standards like PHPCS, PSR1, PSR2, Squiz and Zend. Many popular frameworks like WordPress, Drupal and Symfony2 also come with their code style rules for PHP Code Sniffer. PhpStorm integrates PHP Code Sniffer so that its warnings and errors can be shown in the same way PhpStorm inspections are shown.

- Prerequisites
  - [Installing via Composer](#)
  - [Installing via PEAR](#)
- 1. [Enable PHP Code Sniffer integration in PhpStorm](#)
  - [1.1. Specifying the path to PHP Code Sniffer](#)
  - [1.2. Configure PHP Code Sniffer as a PhpStorm inspection](#)
- 2. [Viewing phpcs warnings and errors in the editor](#)
- 3. (optional) [Checking code styles for the entire project](#)
- 4. (optional) [Installing custom code styles](#)
  - [4.1. Obtaining custom code styles](#)
  - [4.2. Using custom code styles with PhpStorm](#)

## Prerequisites

To make use of PHP Code Sniffer, we first have to install it on our system.

## Installing via Composer

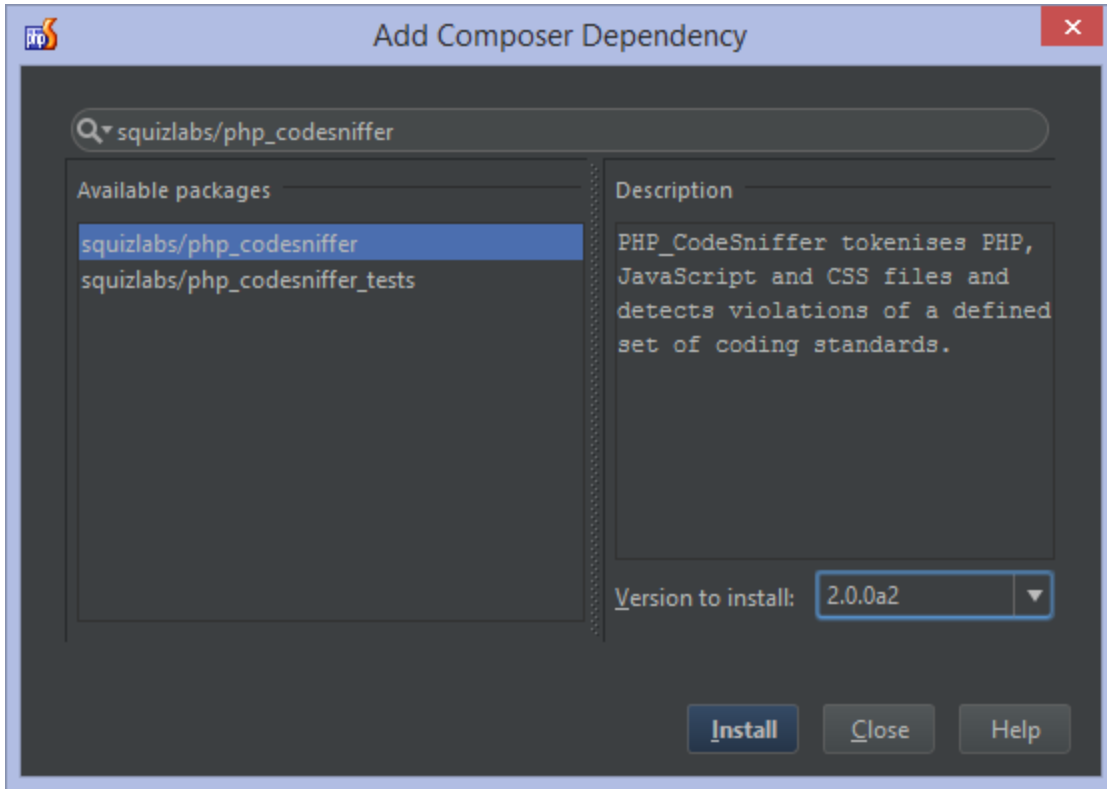
A [Composer package](#) for PHP Code Sniffer is available. It can be used to do a system-wide install, using the following command:

```
composer global require "squizlabs/php_codesniffer=*"
```

If you prefer to install into the vendor directory, a dependency can be added to composer.json instead. This can be done using [Composer Support in PhpStorm](#) or manually. Composer will install PHP Code Sniffer under the following path in your project: ./vendor/bin/phpcs and ./vendor/bin/phpcs.bat.

### ▼ [Using Composer support in PhpStorm...](#)

Use the [Composer | Add dependency...](#) context menu on your project:



#### Manually editing composer.json...

Add the following to the composer.json file:

```
"require-dev": {  
    "squizlabs/php_codesniffer": "1.*"  
}
```

## Installing via PEAR

Another way to install PHP Code Sniffer is by using PEAR:

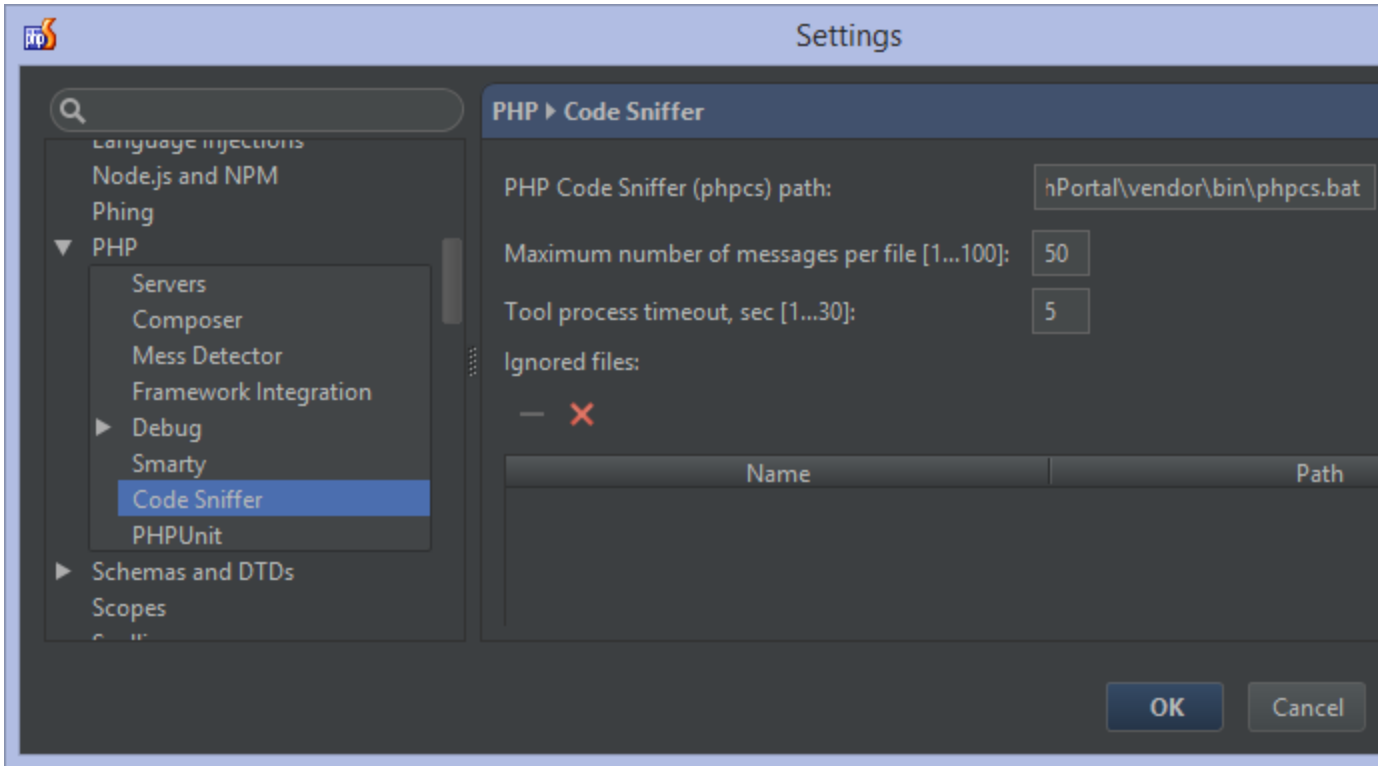
```
pear install PHP_CodeSniffer
```

This will install the PHP Code Sniffer classes into our PEAR installation folder (for example C:\PHP\5.5\pear\PHP\CodeSniffer.php) and will create a bash / batch script in the PHP installation folder (for example C:\PHP\5.5\phpcs and C:\PHP\5.5\phpcs.bat).

## 1. Enable PHP Code Sniffer integration in PhpStorm

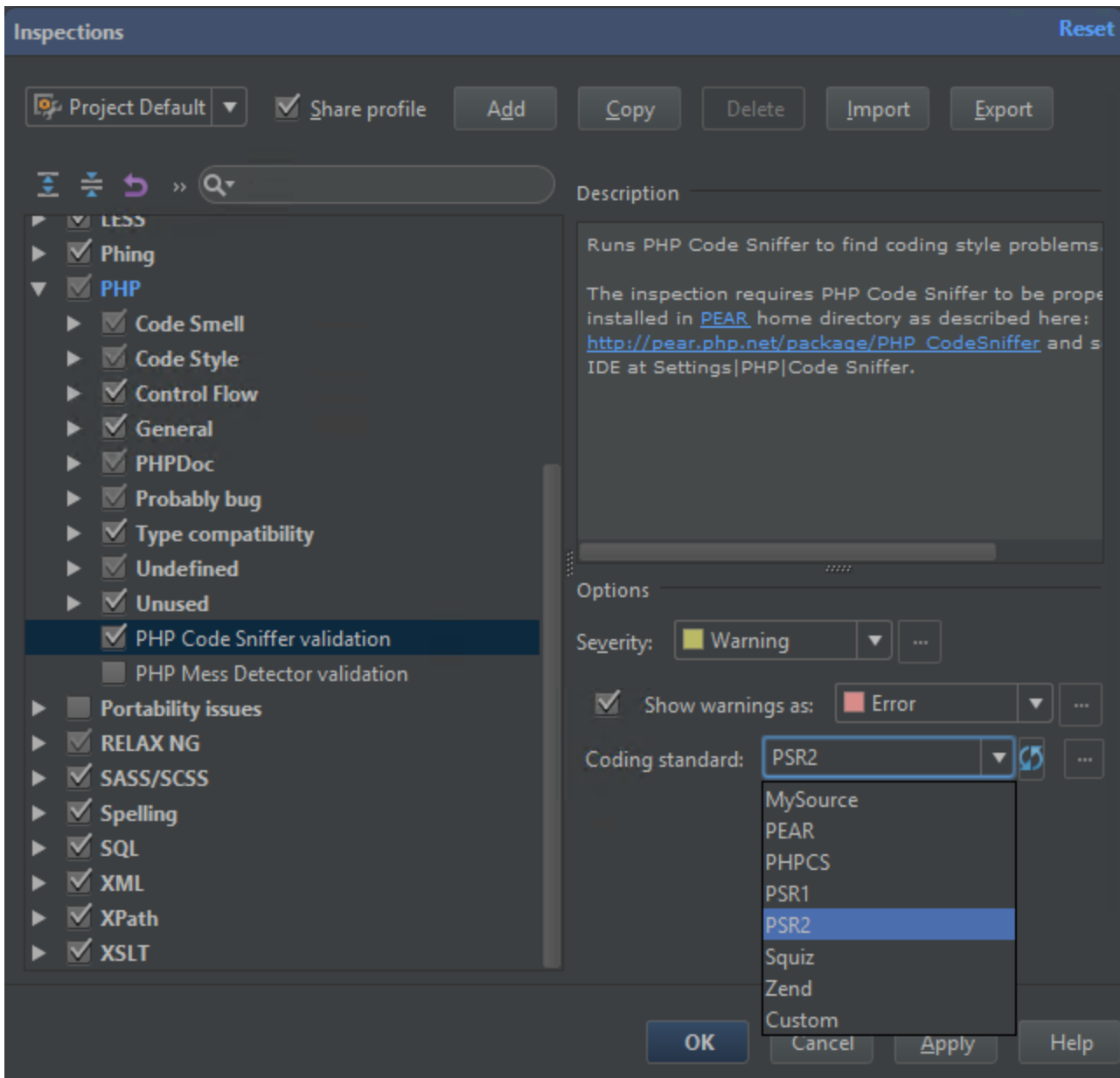
### 1.1. Specifying the path to PHP Code Sniffer

Through Project Settings | PHP | Code Sniffer, we can specify the path to PHP Code Sniffer, as well as some additional options (such as timeout) or files to ignore.



## 1.2. Configure PHP Code Sniffer as a PhpStorm inspection

Once PhpStorm knows the path to the PHP Code Sniffer executable, we can configure it as a PhpStorm inspection through Project Settings | Inspections and then enabling the PHP Code Sniffer validation inspection.



We can configure several additional parameters:

- Severity: the severity of the inspection that will be shown in PhpStorm. This defaults to Weak warning; in the screenshot above we bumped it up to be a Warning instead.
- Show warnings as: how should PHP Code Sniffer warnings be shown? This defaults to Warning, in the screenshot above we bumped it up to be an Error instead.
- Coding standard: the coding standard PHP Code Sniffer should check. PHP Code Sniffer ships with support for PEAR, PHPCS, PSR1, PSR2, Squiz, Zend coding standards and can easily be extended with [your own coding standard](#) or well-known standards like [Drupal](#), [Wordpress](#), [Joomla](#), [Symfony2](#) and many more.

## 2. Viewing phpcs warnings and errors in the editor

The results of running the PHP Code Sniffer validation inspection will immediately be visible in the editor when opening a PHP, JavaScript or CSS file. Just like regular PhpStorm inspections, warnings and errors that were captured by PHP Code Sniffer will be shown in the right gutter and as tooltips inside our code:

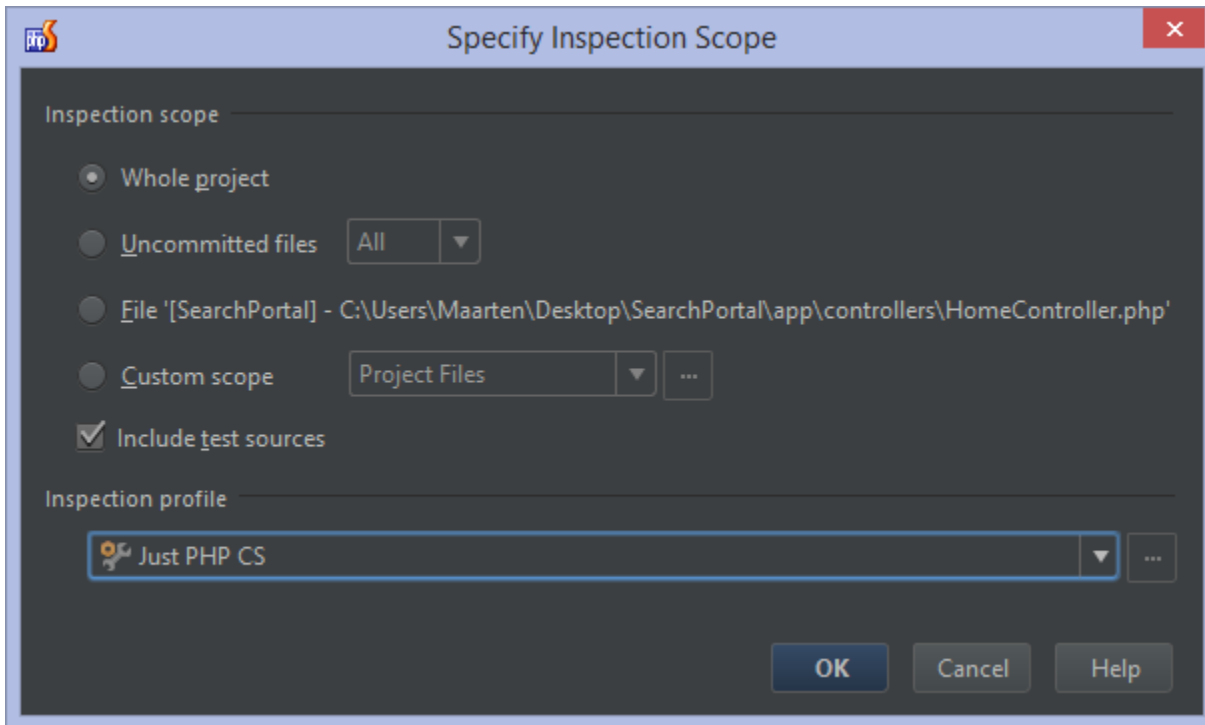
```
HomeController.php x
HomeController search
$plugins = (array)json_decode(file_exists($fileName) ? file_get_contents($fileN
foreach ($plugins as $plugin) {
    $pluginInstance = new $plugin->class;
    $searchengines[] = (object)array('slug' => $pluginInstance->getSlug(), 'nam
}
return View::make('home')->with('searchengines', $searchengines);
}
}

public function search() {
    $searchengines = array();
    $searchengines = array();

    // find all search engine plugins
    $fileName = realpath(dirname(__FILE__) . '/../plugins/plugins.json');
```

### 3. (optional) Checking code styles for the entire project

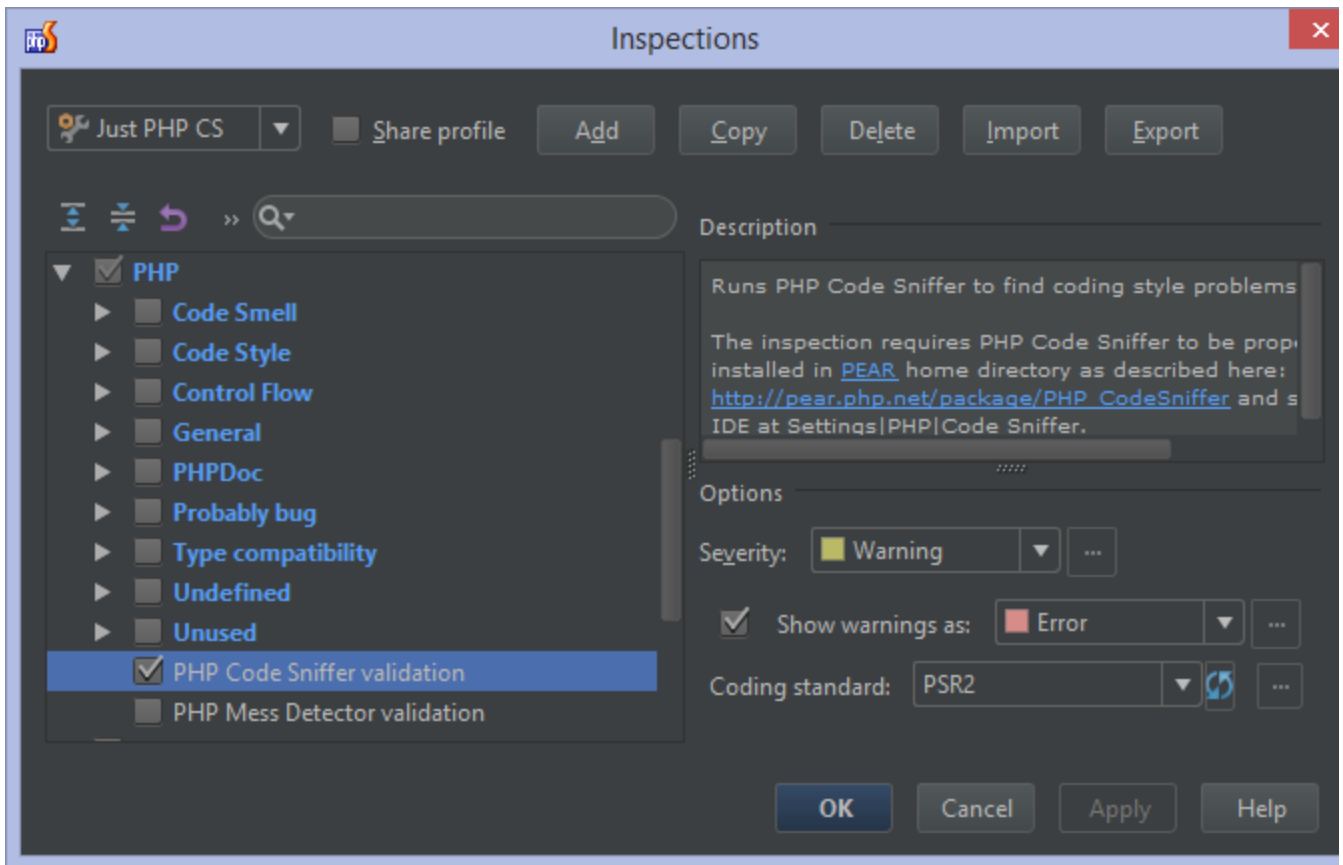
Whenever we want to learn about how well (or how poorly) we are adhering to the selected coding standard, we can run PHP Code Sniffer in "batch mode." Using the Code | Inspect code menu, we can let PhpStorm run the PHP Code Sniffer validation on all files in our project (or on a subset). We will have to specify the inspection profile to use, which defaults to the inspection profile that is active for the current project.



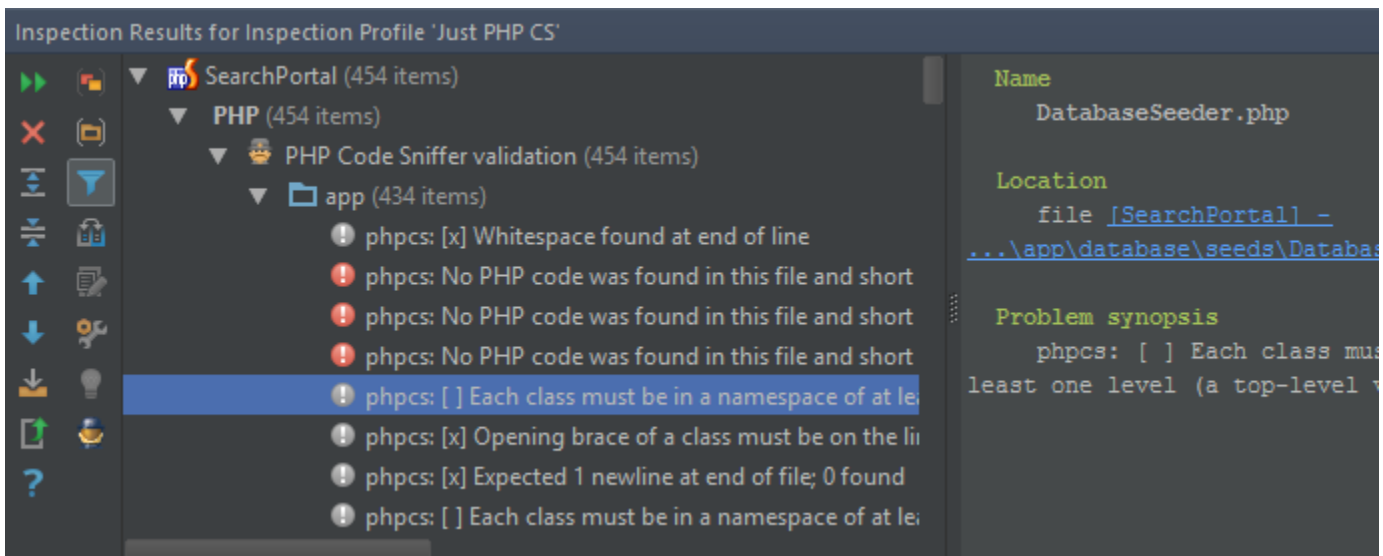
In the screenshot above, I have created a customized inspection profile which will only run PHP Code Sniffer and skips all other PhpStorm inspections.

#### ▼ (optional) Creating a custom inspection profile...

1. From the Specify Inspection Scope dialog, click the ... button to open the available inspection profiles.
2. In the toolbar, click Add to create a new Inspection Profile.
3. Give the custom Inspection Profile a name and click OK.
4. Disable all inspections that are not required, and enable the ones that are.



Once PhpStorm and PHP Code Sniffer finish their work, the inspection results will be shown. Warnings and errors found by PHP Code Sniffer will be listed, grouped by folder. Clicking a warning on the left will display details on the right.




Using the toolbar on the left, we can change filtering and grouping options, as well as export the inspection results as XML or HTML.

#### 4. (optional) Installing custom code styles

While PHP Code Sniffer ships with many supported code styles, sometimes we have to use a custom code style for the project we are working on. For example, Drupal projects typically adhere to the [Drupal Code Standard](#). Luckily, PHP Code Sniffer supports adding custom code styles.

## 4.1. Obtaining custom code styles

 There is no formal way of obtaining custom code styles. Most ship with a readme that explains how to install the required files on our system or in our project.

Even though there is no standard way of obtaining additional code style configurations, installing them is more or less the same. Typically installation consists of:

- Downloading the code styles
- Copying the code styles into PHP Code Sniffer's installation directory

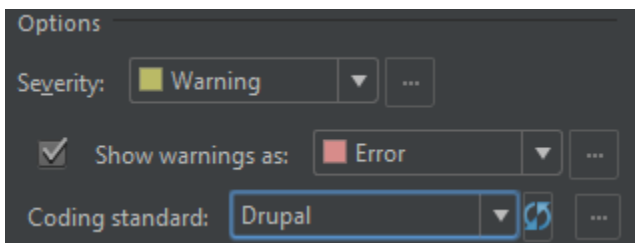
### ▼ Example: installing Drupal code style...


The [Drupal code style](#) for PHP Code Sniffer ship as a Composer package. We can add it to our project by installing the drupal/coder package, either system-wide or in our project. When installing Drupal's Coder package, code styles will reside in the `./vendor/drupal/coder/coder_sniffer/Drupal` folder.

To install these code styles, we can copy the `./vendor/drupal/coder/coder_sniffer/Drupal` folder to the Standards folder of the PHP Code Sniffer installation. When installed through Composer, this would mean copying the folder to `./vendor/squizlabs/php_codesniffer/CodeSniffer/Standards/Drupal`.

## 4.2. Using custom code styles with PhpStorm

We can configure PhpStorm to make use of custom code styles through Project Settings | Inspections, updating the PHP Code Sniffer validation inspection settings. From the Coding Standard dropdown list, we can select the coding standard we just copied into PHP Code Sniffer's installation directory.



 Depending on how the code style was developed, it is not always required to copy the code style into the PHP Code Sniffer installation folder. For many code styles, we can select the Custom code style through Project Settings | Inspections, and specify the path to the code style we want to use in PhpStorm.

