Debugging PHP Web Applications with Run Debug Configurations

Tweet

PhpStorm comes with several ways of debugging PHP Web Applications. We can use zero-configuration debugging with Xdebug and with Zend Debugger. But next to that, there are also Run/Debug configurations we can use to debug our web applications with PhpStorm. In this tutorial, we'll see how we can create a Run/Debug configuration, the various types of Run/Debug configurations and when to use them.

✓ Be sure to check the other tutorials about debugging PHP code with PhpStorm to learn more.

Prerequisites

To debug PHP code with PhpStorm, we will need Xdebug or Zend Debugger. Make sure either Xdebug or Zend Debugger are installed and configured with PhpStorm.

1. Create a Run/Debug Configuration

Run/debug configurations define how a script or application can be executed and launched from the IDE. They are not required for debugging web applications (see zero-configuration debugging with Xdebug and with Zend Debugger), but come in handy for several scenarios.

Using the Run | Edit Configurations menu, we can add a new Run/Debug configuration clicking the green + button.
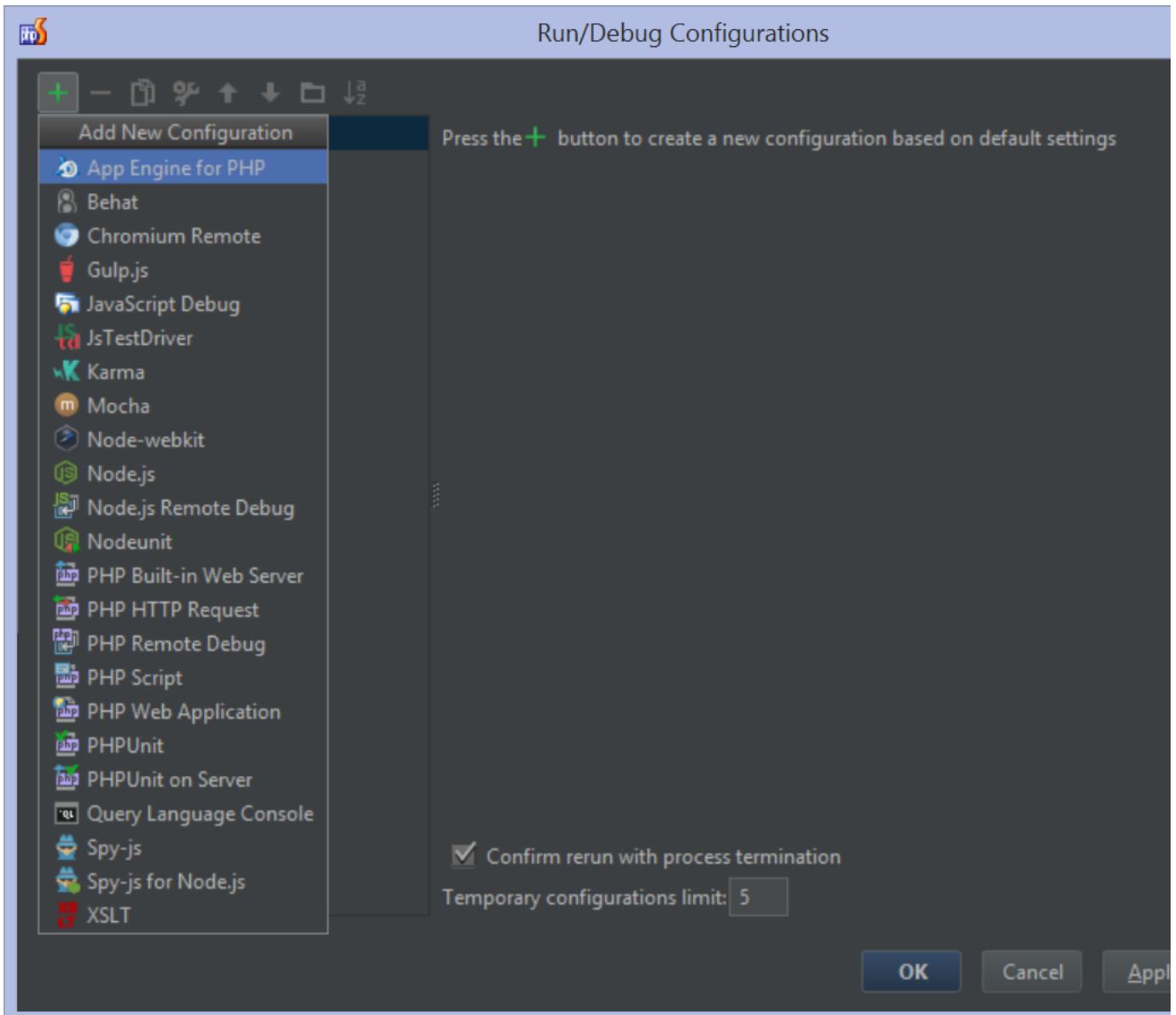
For PHP, there are several Run/Debug configuration types available. For web applications, these are the most interesting:

- PHP Built-in Web Server, which will launch our application using PHP's built-in web server. This configuration type is perfect for debugging full applications without Apache, Nginx, IIS or another full-blown web server.
- PHP Web Application, which will launch our application using Apache, Nginx, IIS or another web server that runs on our machine or remote. This configuration type is perfect for debugging full applications where a production-like setup is desired. For example when `.htaccess` support is needed or the application will be debugged end-to-end.
- PHP HTTP Request, which will execute a specific HTTP request against our application. We can define the URL as well as query string and request body for that request. This configuration type is perfect for debugging a specific page, or an API call when developing REST API's.
- PHP Remote Debug, which will make the IDE await debugger connection with specific IDE key or session id. This run configuration is universal and can be used whenever other approaches are not applicable. However, we strongly recommend using the Start Listening for PHP Debug Connections mode instead.

The other Run/Debug configuration types can be used for Debugging PHP CLI scripts with PhpStorm or Debugging and Profiling PHPUnit and Behat Tests with PhpStorm.

Each Run/Debug configuration comes with specific parameters that define how and where our application will run. Let's go over them.

PHP Built-in Web Server

The PHP Built-in Web Server configuration launches our application using PHP's built-in web server. It's perfect to quickly run (and debug) an application without having to configure a web server like Apache, Nginx or IIS. It just launches the application

and handles HTTP requests to it.
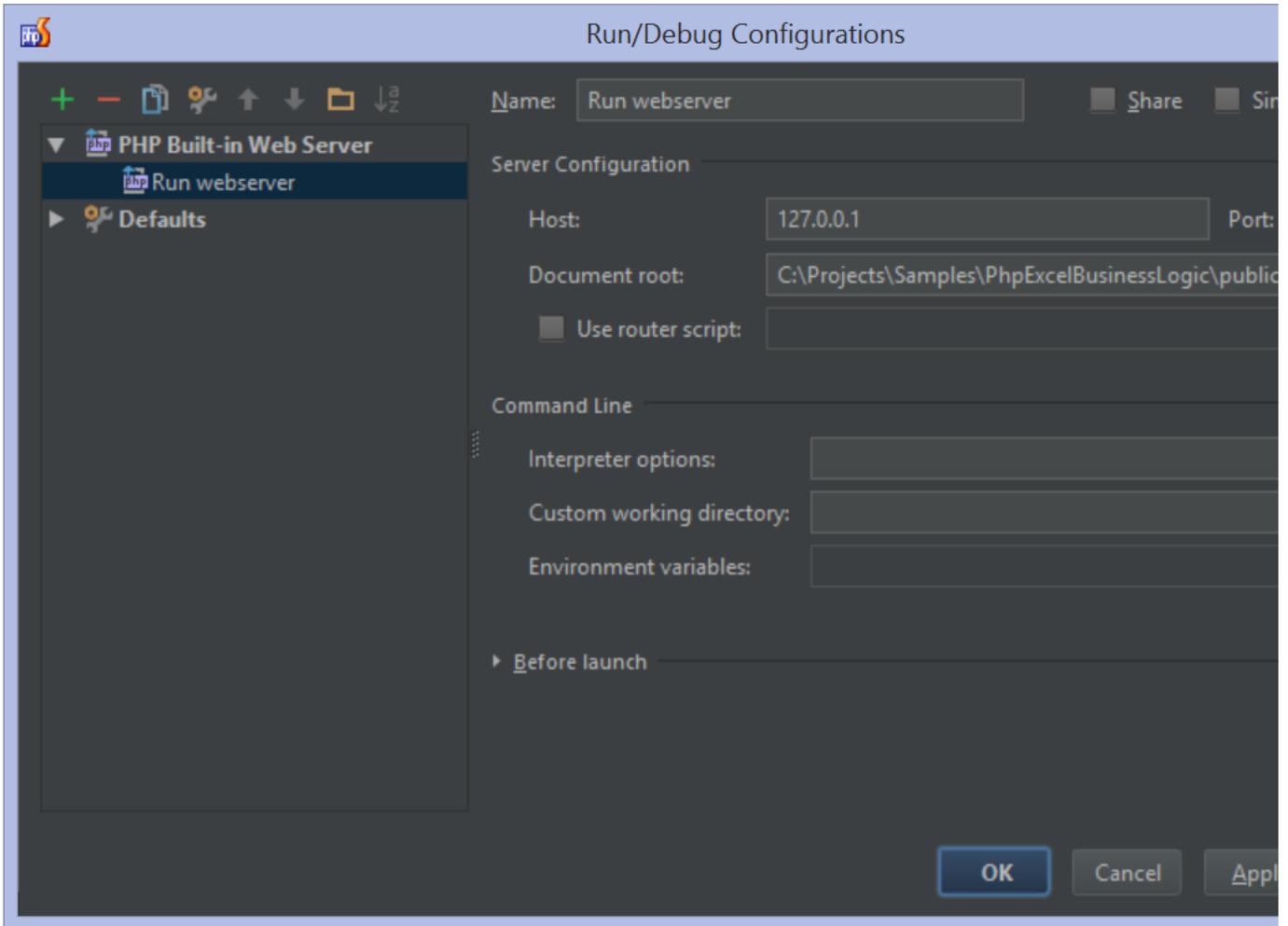
The PHP Built-in Web Server configuration requires two main settings:

- The host and port it will run on
- The document root for the application

Optionally, a router script can be specified as well. Similar to URL rewriting, the web server will route all requests through the specified router script.

PHP Web Application

The PHP Web Application configuration lets us run our application using Apache, Nginx, IIS or another web server that runs on our own machine or remote. It is well-suited for running and debugging complete PHP web applications that need to run in a production-like setup.
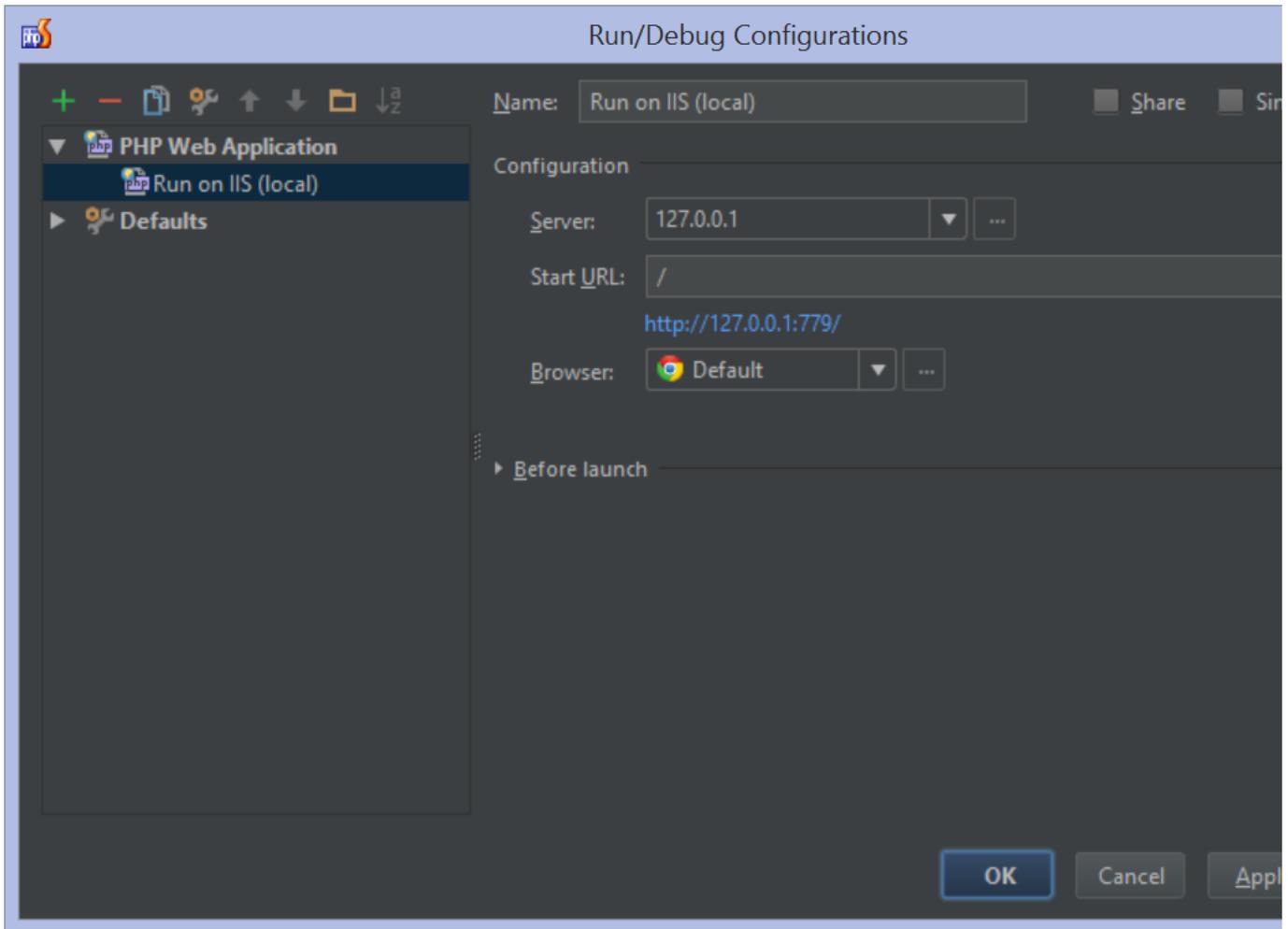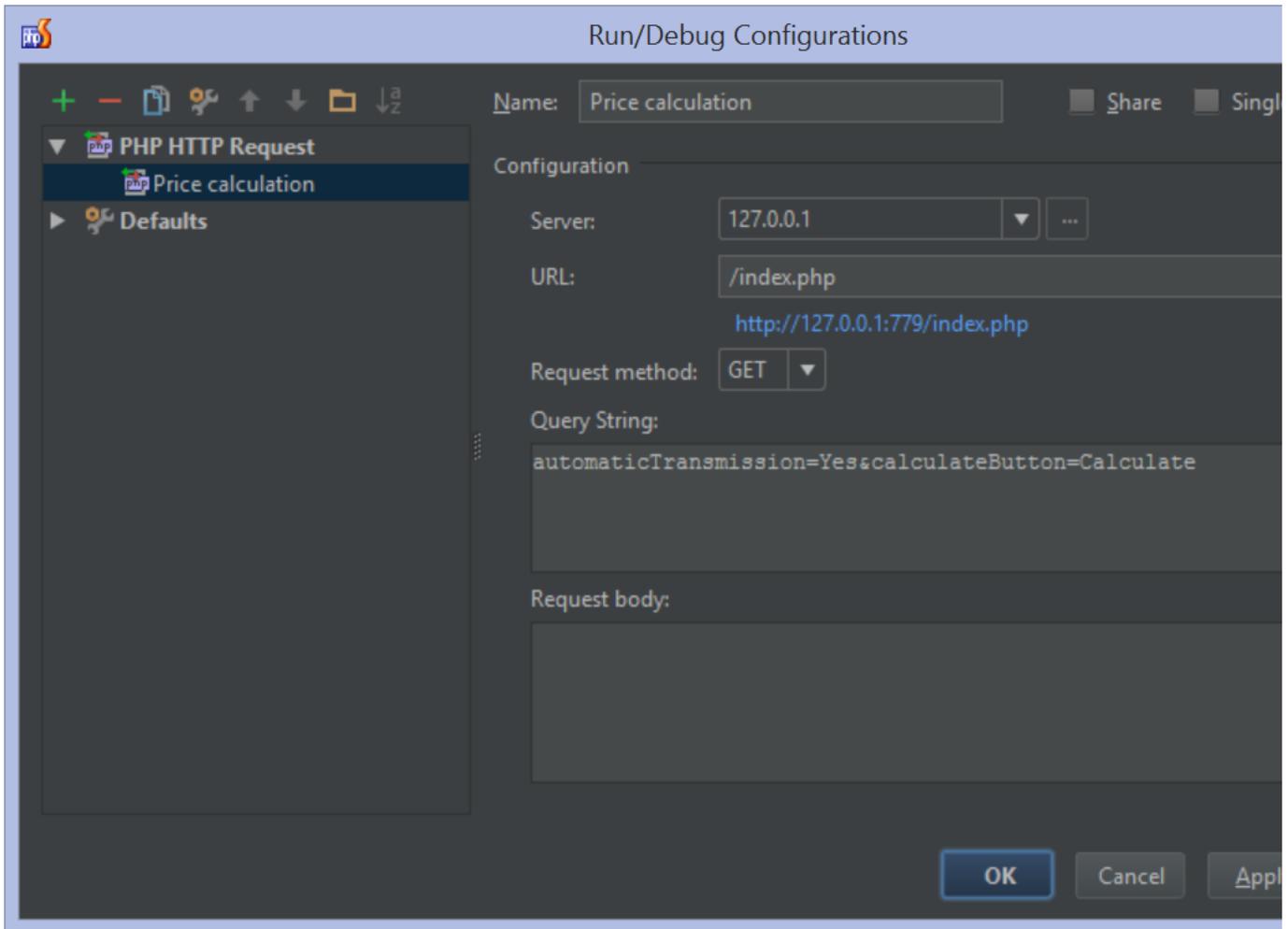
For this configuration, we have to specify:

- The server where the application is deployed and runs. See Deploying PHP applications with PhpStorm for more information about setting up a deployment and path mappings.
- The URL to launch the browser with.

PHP HTTP Request

The PHP HTTP Request configuration executes a specific HTTP request against our application that runs either local or remote, on the PHP Built-in Web Server or on a real web server. It's a great configuration type to run and debug specific application scenarios (like requesting a product listing) and API calls when running REST API's.
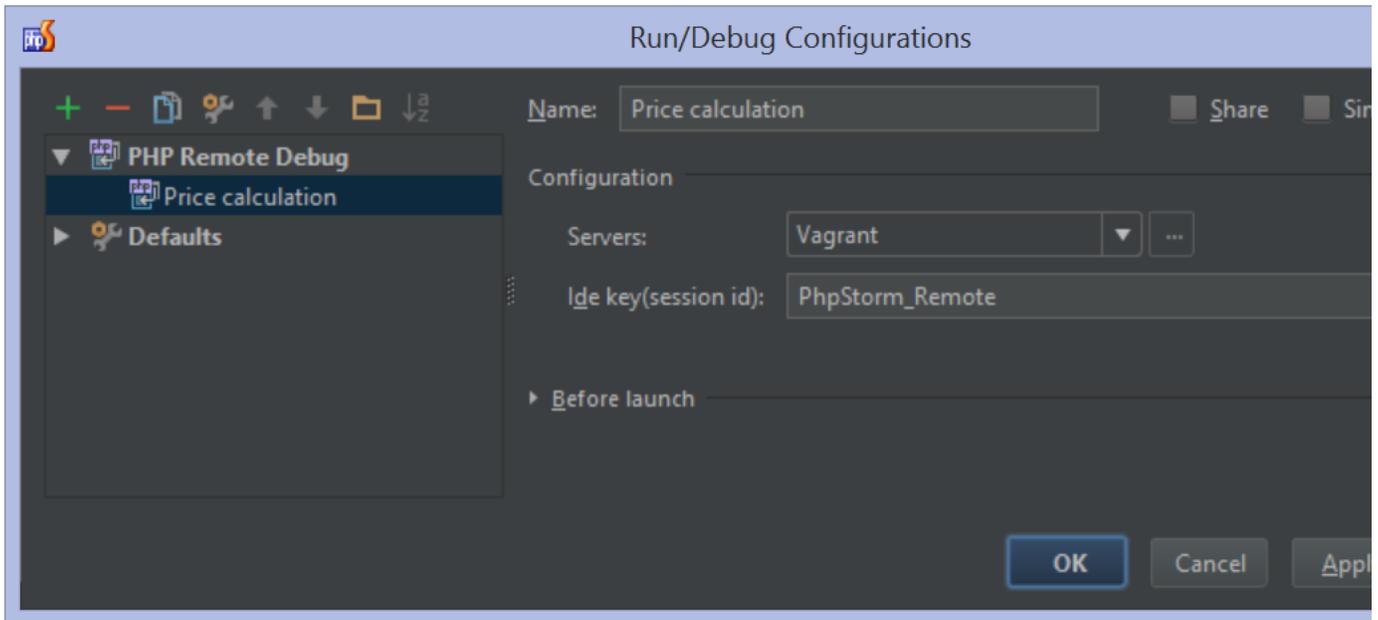
For this configuration, we have to specify:

- The server where the application is deployed and runs. See Deploying PHP applications with PhpStorm for more information about setting up a deployment and path mappings.
- The URL to request. Note that no browser is opened: PhpStorm executes the request to this URL for us when running or debugging.
- The request method (GET or POST)
- An optional query string to append to the request URL
- An optional request body to send when executing the request

PHP Remote Debug

The PHP Remote Debug configuration will make the IDE await a debugger connection with specific IDE key or session id.

This run configuration is universal and can be used whenever other approaches are not applicable. However, we strongly recommend using the Start Listening for PHP Debug Connections mode instead.

For this configuration, we have to specify:

- The server where the application is deployed and runs. See Deploying PHP applications with PhpStorm for more information about setting up a deployment and path mappings.
- IDE key or session id.

2. Debug!

From the toolbar, click the "bug" icon to start the debugger. We can also use the Run | Debug menu or the Shift+F9 keyboard shortcut (CMD+F9 on Mac OS X). This will launch the selected Run/Debug configuration and pauses execution at the first breakpoint.
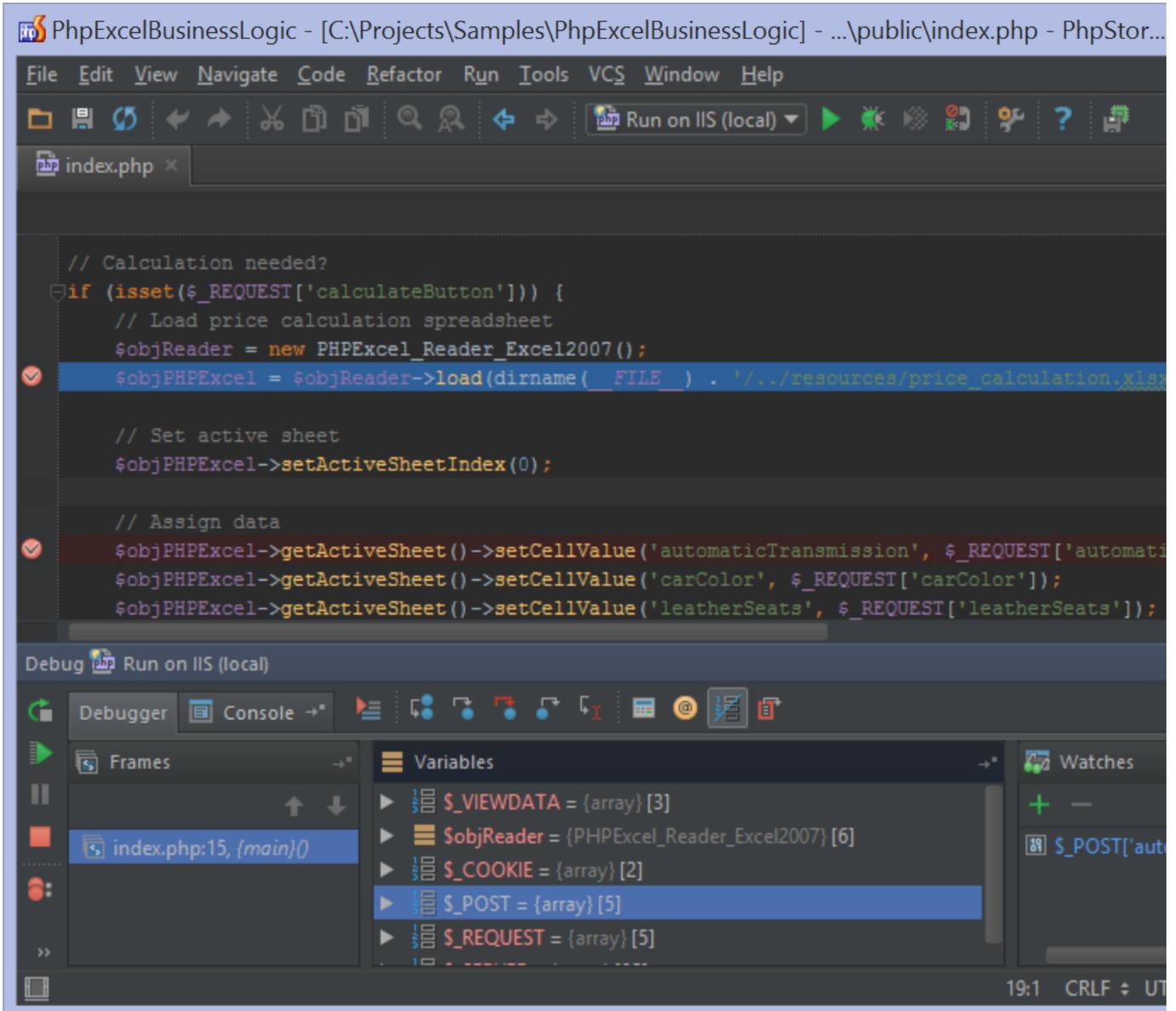
We can now continue debugging as described in Using the PhpStorm Debugger.

> ✅ When launching the debugger, PhpStorm appends a query string to the URL that is requested (for example `http://12`
> `7.0.0.1:779/?XDEBUG_SESSION_START=10409`). This is a request variable that's required by Xdebug or Zend
> Debugger) to launch the debugger. The PhpStorm bookmarklets and Browser Debugging Extension do a similar thing
> when using zero-configuration debugging with Xdebug and with Zend Debugger.

Tweet