Zend Studio 3.0

Quick Start Guide

This Quick Start Guide walks you through the Zend Studio 3.0 major features, helping you to get a general knowledge on the most important capabilities of the application. A more complete Information Center is available within the Zend Studio.

The Quick Guide contains the following topics:

Installing Zend Studio

The following describes the download, installation and registration procedure of the Zend Studio application.

1. To download the Zend Studio, go to:

    http://www.zend.com/store/download_list.php

2. Select the relevant platform and click *Download*.

3. a. If you are a current Zend user, type in your Zend *User Name* and *Password* to *Login* and
       skip to step 4.

    b. If this is your first time at Zend, click *Sign Up Now* (on the lower right side of the screen)
   to *Register*.

    Complete the Registration Form and *Submit*. Mandatory fields are underlined however, any
    additional information will assist us in providing you with a better service.

Submit the form.

Upon registration, you will receive a *Welcome* e-mail, confirming your Zend Username. We recommend that you keep this for future reference.

4. You are now ready to download the Zend Studio application.
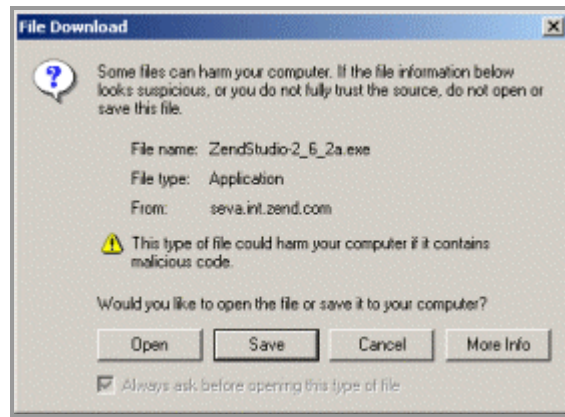
5. After downloading the file, activate the exe file in Windows or extract and activate the installation file and follow the installation process.
    Be sure to read the installation instruction. You are now ready to launch the application.

    When launching the product for the first time, you need to enter your *Registration Name* and *License Key.*
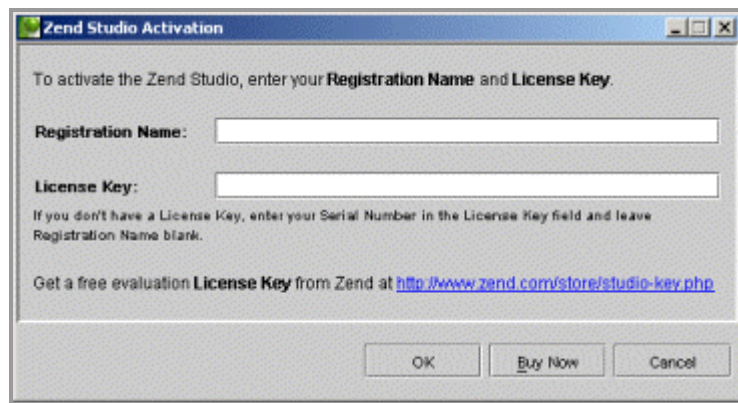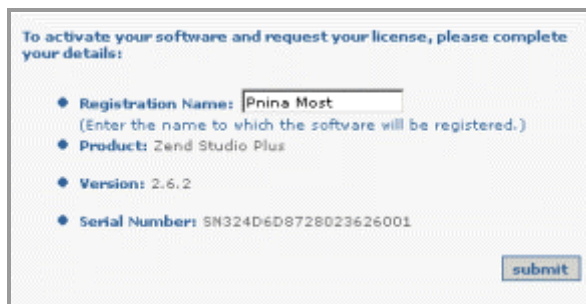
    a. If you are evaluating Zend Studio, enter the *Registration Name* and *License Key* received by e-mail and you are ready to start using the product.

    b. Otherwise, go to the following link to generate a *License Key* based on the *Serial Number* entered.

Click the link to generate a License Key.

6. If you are not logged into zend.com yet, you are requested to enter your *User Name* and *Password*.

Enter your Registration Name (any name you wish to register with) and *Submit*.

7. The following form provides you with a *License Key*, which you will use when launching the Zend Studio application.

You are now ready to use the Zend Studio application.

Using Zend Studio

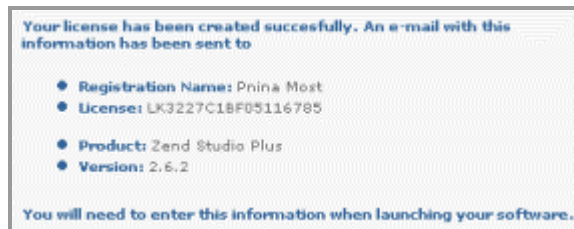The development of a web application follows a basic flow. Although the flow is somewhat straightforward, the general process can take several cycles before the application is complete. You may want to test and debug every function as it is being written, or you may want to wait until the bulk of an application is complete before giving it a test.

This being said, the basic flow of the creative process is shown in the following image.
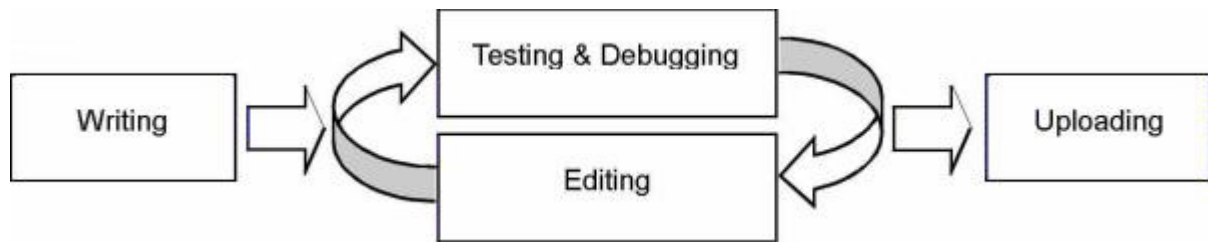
The Zend Studio installs everything you need for a full client-server development environment. The Zend Studio Client package, including the Zend Development Environment, in conjunction with the Zend Studio Server package installs your development environment and complete HTTP/PHP server with development extensions support.

The Zend Development Environment is a convenient tool that integrates an editor, debugger and project manager to help you develop your code, manage and debug it. It can connect to your own installed server or directly to the server component of the Zend Studio to allow you to debug your code in its natural environment.

- **Coding** - Zend Development Environment features, such as code completion, auto indentation, project management, cross-file searches, syntax highlighting and syntax error highlighting, help you accomplish coding related tasks.
  Once you have created a script or a section of script, you can move to the next step to see if the script is running properly.
- **Testing and Debugging** – These is how you locate errors in the code, detects bottlenecks in scripts and optimize overall performance of your applications.
- **Editing -** In this step you use the same tools used in the writing stage to help speedup your editing process. Because post-debugging edits strives to eliminate bugs in the application, a debugging session most often follows an editing session. It is common for several cycles of editing, debugging and more editing.

- **Uploading -** Uploading is the publishing of your web application to a hosting server for Web access. It is often the last step, which is done after you have determined that your application is running correctly.

The following highlights some of the main features of Zend Studio V3.0.

1. *Code Completion*

The Code Completion feature enables you to speed up development and provides you with an easy access to PHP Classes, Functions, Variables, Constants, Keywords HTML tags, attributes, attributes value and more.

There are two Code Completion types available: PHP and HTML.

Code completion includes function summary descriptions, prototype structure and tips. Code completion engine recognizes all aspects of your code according to context:

- Built in PHP functions, classes variables
- Your own functions, classes variables
- Object dereferencing for methods and properties

Code completion automatically collects information from PHPDoc-style comments and embeds it in the completion suggestions
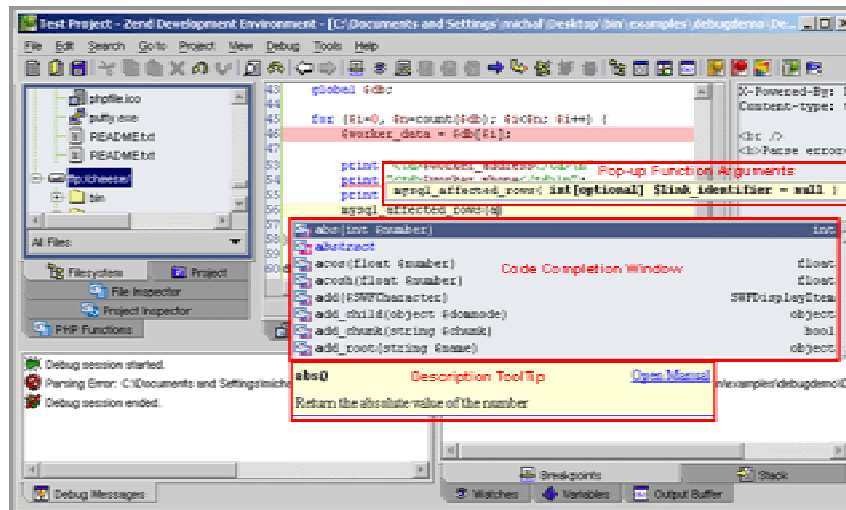
While coding – code completion window, pop-up functions arguments and description tooltips open automatically.

2. *Highlight Syntax Errors*

Immediately locate syntax errors as you type. The Syntax Error Indicator bar pinpoints syntax error location within file.
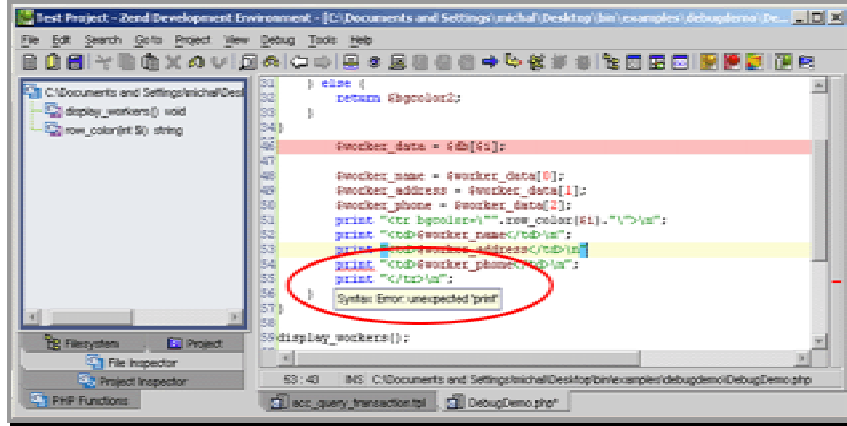
Syntax errors are underlined in red as you type. Additionally, a small red line can be found to the right of the syntax error pinpointing where it can be found in the code. To view the error description tooltip, place the cursor on the error marking.

3. *CVS Integration*

The Zend Development Environment integrates directly with the CVS version management system.
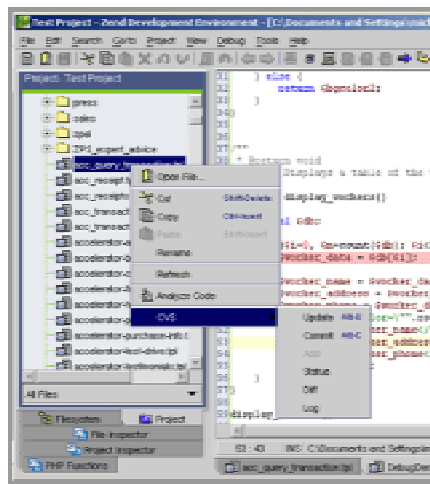
CVS integration is supported if a CVS server is installed and a CVS module has been checked out. Add files to a project and right click on file or folder for CVS options.

4. *FTP Integration*

Transparently work with files on remote FTP systems as if they were part of a local file directory.

You can use the FTP Root to read and write using to FTP severs with methods such as of copy and paste, or drag and drop files.

5. *Customization*

For your convenience and in order to maximize efficiency, you can easily personalize a variety of settings and functions.

To customize settings, go to *Tools>Customization> and select the requested tab.*

The Customization Window is divided into several tab headings as follows:

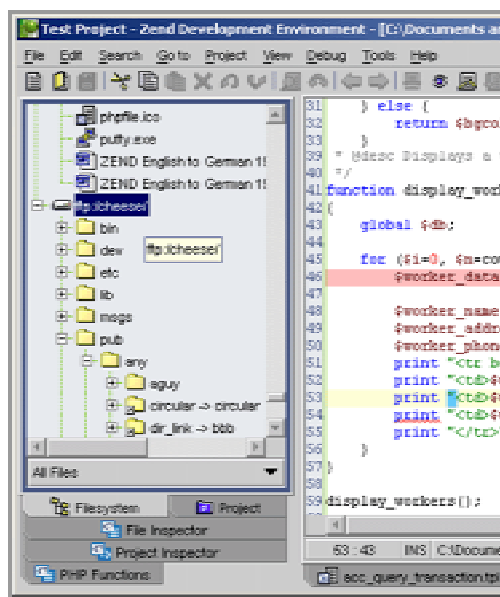| Tab	Customize
Desktop	Desktop icons, fonts, background colors, language and more.
Editing	Settings and options, which effect editing tools and appearance.
Code Completion	Settings for content and control of completion windows.
Colors	Color assignments for Syntax Highlighting.
Debug	Settings and customization features for the debugging process.
Keymap	Settings for customizable shortcuts.
CVS	**File types and associated file extensions.**
File Types	Enables CVS integration for version control management
Dialogs	Settings for optional dialog prompts.

You can restore all default settings by right clicking in the Customization Window and selecting *Restore all* from the short cut menu.

6. *File/Project Inspector*

The Project Inspector provides the ability to view the constants, classes and members declared throughout the Project files.

Each displayed class and member is displayed in a hierarchical tree. This tree can be expanded and collapsed to help in viewing information. Additionally Inspector icons are displayed next to each item to indicate constants, class, member variables or member functions.

By clicking on a constant, class of member you can go to file and declaration statement.

When placing the cursor on an item, you can use the right mouse menu and jump directly to the source in the editing workspace or recursively jump to the Parent Class in the File/Project Inspectors. In order to view only the constants, classes and members for the current file, view the File Inspector. The following image shows constants, classes and member variables and functions displayed in the Project Inspector window
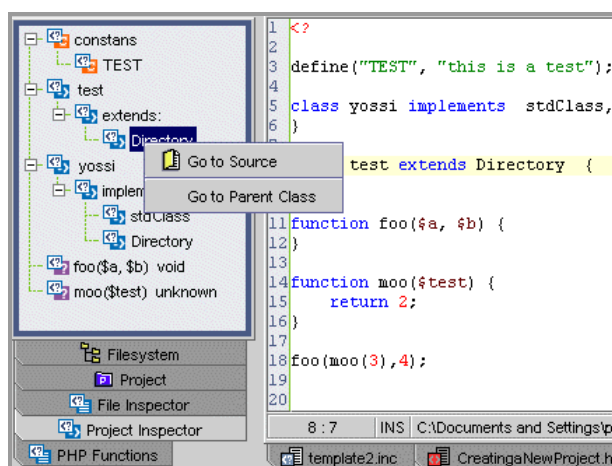
7. *Go To Matching Brackets*

This feature allows you to jump back and forth between the paired enclosing characters.

Enclosing characters include: (){}[]' ' " " <>

From the Main menu, select Go To Go To Matching Bracket.

The cursor jumps to the matching bracket.

When selecting a closing bracket and the matching bracket does not fit in the Editing Window, you can enable the display a tooltip containing the matching bracket line. To do so, simply enable the Matching Brackets Line Tooltip in the Editing tab in the Customization.

8. *Encoding Integration*

You can integrate the Zend Studio Development Environment with the Zend Encoder/SafeGuard Suite, if installed. This package is available during the Zend Studio Installation. Such integration enables you to encode the active project files from within the Development Environment workspace. If you have the Zend Encoder installed, you will need to set the Encoder Path Setting. This setting is the path to the location where the encoder is installed.

To Change the Encoder Path Setting

a.   From the main menu, select *Tools > Customization.* The Customization Window appears.

b.   Select the *Editing* tab. Enter the Zend Encoder Path.

The Zend Encoder does not come standard with the Zend Development Studio. If you would like to find out more information about the Zend Encoder, refer to the Zend Website.

9. *Line/Block Comments*

You can select a location or block of text and tag it as comments. There are two types of comment code, which can be added or removed: Line Comment and Block Comment
The behavior of this feature differs between a PHP section of code and an HTML section.
For PHP Sections, Line Comments option inserts the double forward-slash characters (//) to the line or lines selected as comments

Block Comment option inserts the beginning (/*) and ending (*/) coding in order to mark the selected line or lines as comments.
For Html Sections: Line Comments and Block Comment options behave the same, inserting beginning (<!--) and ending (-->) tags to mark the selected line or lines as comments.

10. *Using Bookmarks*

Use Bookmarks (Add/Remove by using F2) to navigate throughout your script. Furthermore, you can attach a description to each Bookmark, which can be later seen in a tooltip next to the Bookmark. Press *Ctrl+Shift+M* to access the Bookmark Manager, add a Bookmark description and navigate between existing Bookmarks.

Press *Ctrl+F2* to move the cursor to the next Bookmark in the file.

Press *Alt+F2* to move the cursor to the next Bookmark in the entire project.

Bookmarks are highlighted and when placing the cursor on a Bookmark, its description appears in a tooltip.

11. *Finding Text in File(s)*

Search for text or regex in a file, several files or in an entire project by using *Ctrl+Alt+F*. Search Results are displayed in the Messages Window.

12. *Clone View*

When you need to refer to another section of the script that you are editing or to another file altogether, use the Clone View feature.

The Clone View enables you to open additional windows for viewing parts of the same script file or open different files in separate window.

To clone View a file, in the open file right click and select *Clone View*.

You may open as many windows as you need, either of the same file or different files.

13. *Managing Projects*

A Project is a subset group of files from your Filesystem. You can specify which files belong to the project and add or remove files as needed. You can create a New Project if you wish to define a working environment with unique characteristics such as debug configurations, Bookmarks and Watches storage, etc

14. *Remote/Local Debugging*

Remote and local debugging allows you to debug PHP applications in their native environment by connecting to the server they are deployed on. Use Zend Studio's virtual Filesystem capabilities to test the code changes and fix on the remote server, even before the application is deployed.
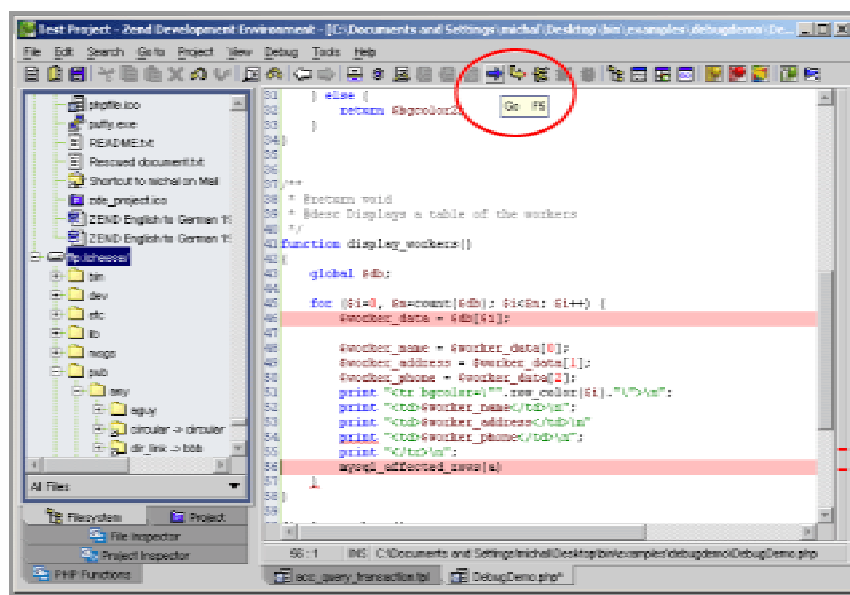
Prior to running a debug session, you need to add Breakpoints and Watches. During debugging you can use these tools to stop the debugging, gather and review information.

Watching variables can become a process of scrolling through numerous variables to locate and pinpoint variable problems. The Zend Studio Development Environment provides Watches as a feature to help you narrow the scope. In the Watches area you can place only the items you want to monitor.

Breakpoints are markers that instruct the Zend Debug Server to stop the PHP/Zend Engine at the beginning of the line of code specified.

Use breakpoints to enable you to review information such as Variables, Watches, Stack, and Debugging Messages.

Breakpoints have two states: enabled and disabled. A disabled breakpoint remains in the project and displays in the Edit Window and Breakpoint tab, however during debugging a disabled breakpoint will not cause a break in execution.

You can condition breakpoints by adding any valid Boolean expression, thus condition the stop. (e.g. "$var>5" causes the engine to stop only if the value of $var is greater than 5).

Place breakpoint, add watches and then run the debugger either by pressing *Go* button or *Go to Cursor* button.
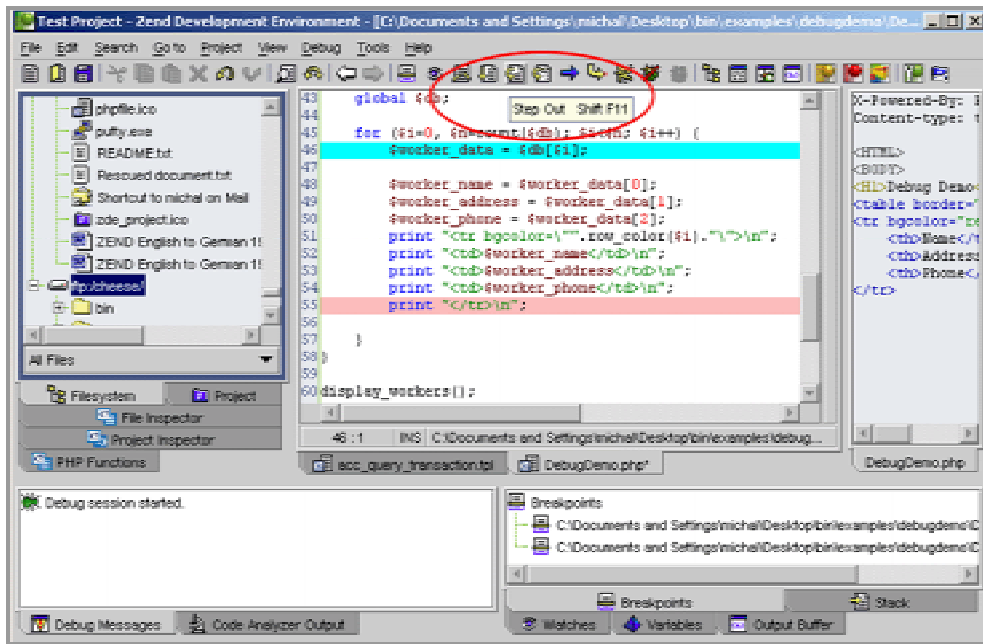
Start debugging and skip to the next bug by using the *Step into, Step over, Step out* buttons.

15. *One-click Debug Toolbar*

Revolutionize the concept of debugging and profiling by allowing you to easily debug and profile directly from your browser. Automatically start a debug session that open all accessed URLs in debug mode. Effortlessly debug frame based applications, quickly debug and profile all forms on a website or repeatedly debug the same form in real time.
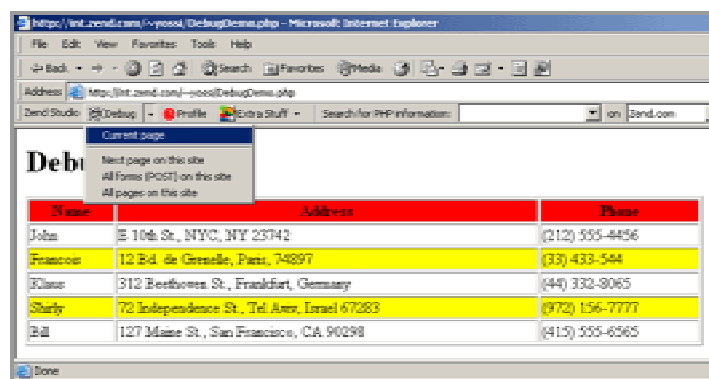

The browser debug toolbar allows you to debug (and profile) the page viewed in the browser. Click Debug and the Zend Studio opens with that page in Debug mode.

16. *Code Profiling*

Optimize the overall performance of your applications by using the Code Profiler. Use the browser debug toolbar to profile the page viewed in the browser window. Click on Profile and the Zend Studio opens with the page in Profile mode.
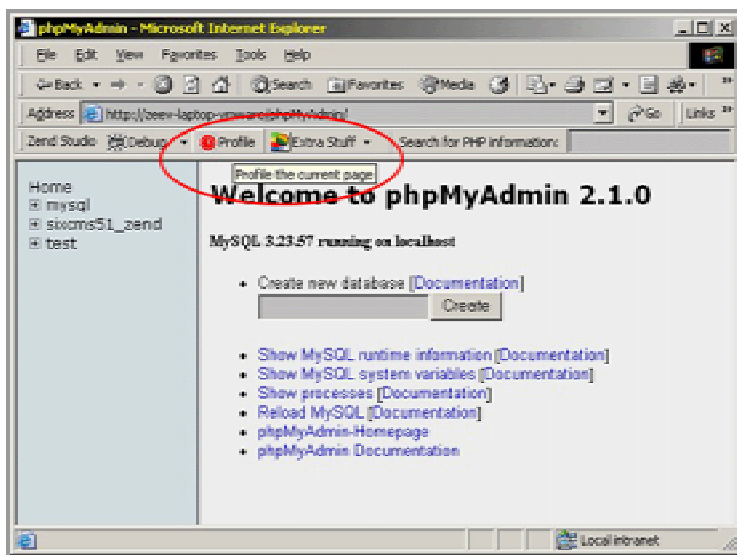
Zend Profiler detects bottlenecks in scripts by locating sections of code, which consume most loading time and provides you with detailed reports that you need in order to optimize the overall performance of your application.

The Zend Studio Profiler performs the following:

- Monitors the calls to functions

- Monitors the number of times that a section of code is executed

- Calculates the total time spent on execution

- Generates reports that reflect the time spent on execution

- Graphically displays information of time division

- Enables comparison statistics between functions

- Enables viewing the file from the server just by clicking on any function

- Shows the hierarchical structure of the functions involved in the script execution

The Profiler user interface contains 3 tabs:

- **Profiler Information** - General information on the profiling duration and date, number of files constructing the requested URL and more. In addition, it displays a Time Division Pie Chart for the files in the URL.
- **Function Statistics** - Provides you with the list of files constructing the URL and detailed information on function in the files.

- **Call Trace** - An hierarchical display of functions according to process order, enabling you to jump to the function, view the function call, function declaration, details and more.
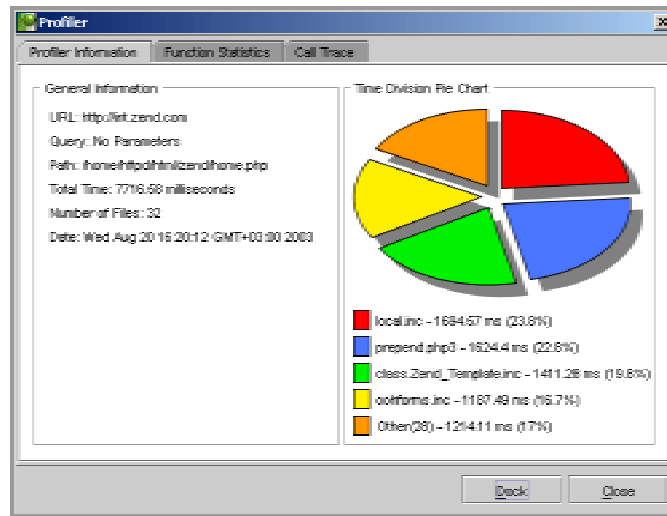

Zend Profiler displays a graphical representation of performance issues within the application.
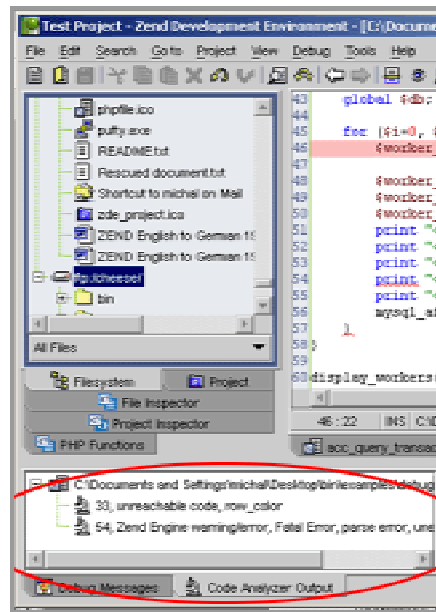
17. *Code Analyzer*

Create robust, secure applications using the Code Analyzer, which analyzes static source code and attempts to reconcile problematic code. Locate unreachable code, code that has been defined but not used or empty variables.

The Code Analyzer supplies you with a detailed error log while focusing on the error location in the file that is open in the Editing Window. In addition, it supplies you with practical suggestions for improving the code.
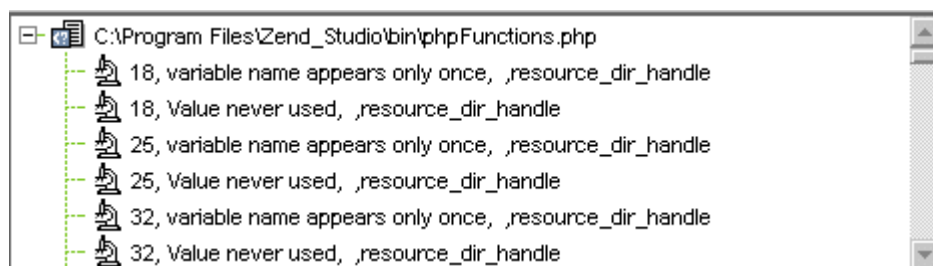
To run the Code Analyzer, select the Project tab and from the right mouse menu, select Analyze Code.

Double-click the code error in the log to move the cursor to the exact location in the file.

From the Tools menu, select *Analyze* Code. The *Code Analyzer Output* window opens with the analysis.

We thank you very much for evaluating the Zend Studio 3.0 application. If you require additional information or if you have any questions, please visit the Zend Support Site.

For information on Zend's other products, visit the Zend Store.