

# DEVELOPING RICH INTERNET APPLICATIONS WITH ZEND STUDIO FOR ECLIPSE

By Roy Ganor

OR...  
PHP AND JAVASCRIPT IN  
A SINGLE FRAME

# Why ?

- Many tutorials and blogs written about “PHP & AJAX”, “ZF & Dojo” and “Rich Internet Application” but take no notice of a suitable development environment
- Many Technologies → Tedious work → Request for PHP and JavaScript IDE



# Outline

- Zend Studio for Eclipse
- AJAX Tools plug-in
- Source editing and Debugging
- Embedded Mozilla Browser
  
- What we will *not* cover
  - What is “Rich Internet Application”
  - How to create a ZF MVC application

You should already know the basics of  
Zend Framework and AJAX features

# Web Development Environment

**AJAX Tools  
& Mozilla**

**Zend Studio for Eclipse**

- **Creating AJAX applications**
- **Browser Tooling**
- **JavaScript Editor**
- **JavaScript Debugger**
- **Embedded Mozilla Browser**
- **Personality Builder**
  
- **Project Management**
- **PHP Editor**
- **PHP Debugger**
- **Database Tools**
- **Web Services**
- **Source Control**
- **PHP Unit Support**

# Setup

## 1. Install Zend Studio For Eclipse

## 2. Install atf-wtp plugin

- plugins → [Installation Path]/plugins
- features → [Installation Path]/features

## 3. Launch Zend Studio

- with `-clean` parameter

# Focus on

- **Improving**
  - Look and Feel
  - Client-Server Balance
  - Client-Server Communication
  
- **Technologies / Mechanisms**
  - CSS
  - DOM manipulation
  - PHP and JavaScript Scripting
  - Asynchronous communication
  - Database management



# Demonstration # 1

# IDE Basics

Menu

Project Management

Outline View

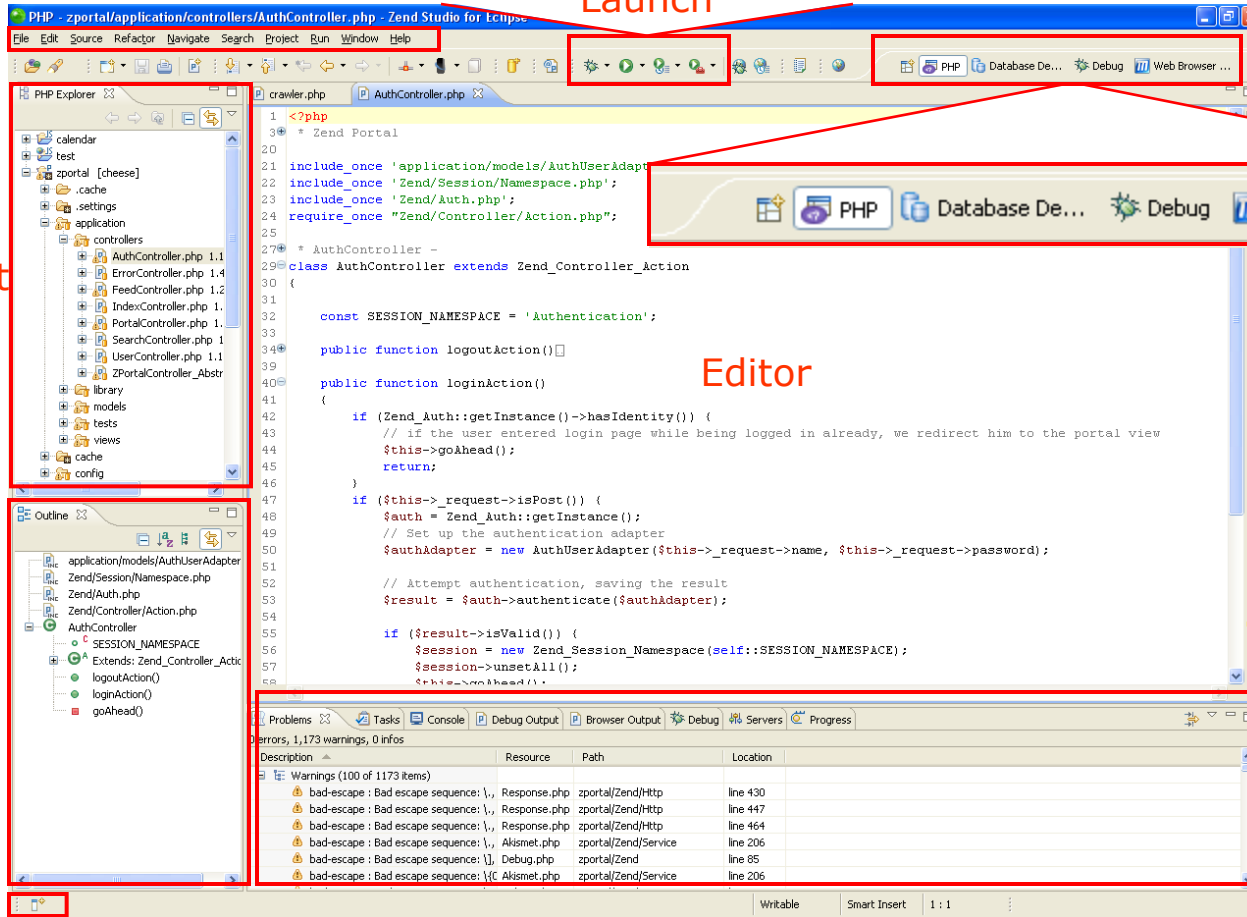
Fast View

Launch

Perspectives

Editor

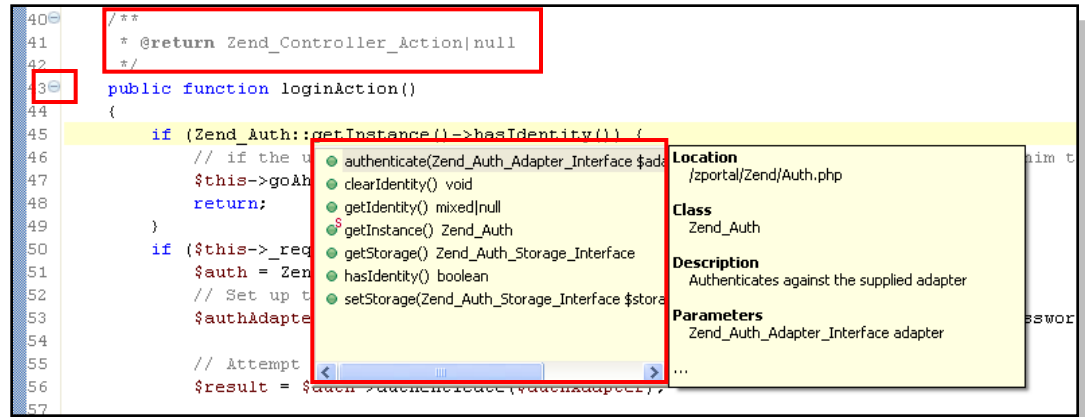
Problems, Tasks, Console



# PHP Development Environment

- **PHP Source Editor**

- Syntax coloring
- Code folding
- Go to Declarations
- Code Assist
- PHP Templates
- PHP Manual links



```
40  /**
41   * @return Zend_Controller_Action|null
42   */
43  public function loginAction()
44  {
45      if (Zend_Auth::getInstance()->hasIdentity()) {
46          // if the u
47          $this->goAb
48          return;
49      }
50      if ($this->_req
51          $auth = Zen
52          // Set up t
53          $authAdapte
54
55          // Attempt
56          $result = $auth->authenticate($adapter);
57
```

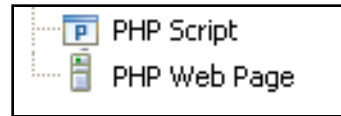
The screenshot shows a code editor with syntax coloring. A tooltip is displayed over the `Zend_Auth` class name, showing its location (`/zportal/Zend/Auth.php`), class name (`Zend_Auth`), description (`Authenticates against the supplied adapter`), and parameters (`Zend_Auth_Adapter_Interface adapter`).

- **PHP Doc assist**

- Using type hints for code assist

- **PHP Debugging**

- Local script
- Web page



# JavaScript Development Environment

- **Add JS Support**

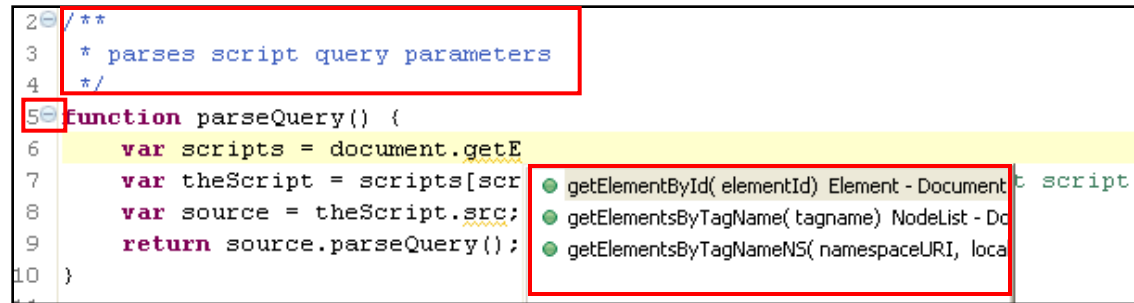
- Right click on your project, then →



- **JavaScript Source Editor**

- Syntax coloring
- Code folding
- Go to Declarations
- Code Assist
- JS Templates
- Mark occurrences

```
20 /**
21  * parses script query parameters
22  */
23
24 function parseQuery() {
25     var scripts = document.getE
26     var theScript = scripts[scr
27     var source = theScript.src;
28     return source.parseQuery();
29 }
30
```

A screenshot of a code editor showing JavaScript code. The code is syntax-highlighted: comments are in blue, keywords like 'function' and 'return' are in red, and variables are in black. A red box highlights the comment block on lines 20-22. Another red box highlights the function call 'return source.parseQuery();' on line 28. A tooltip is visible on the right side of the editor, showing the signature for 'parseQuery()' and its return type.

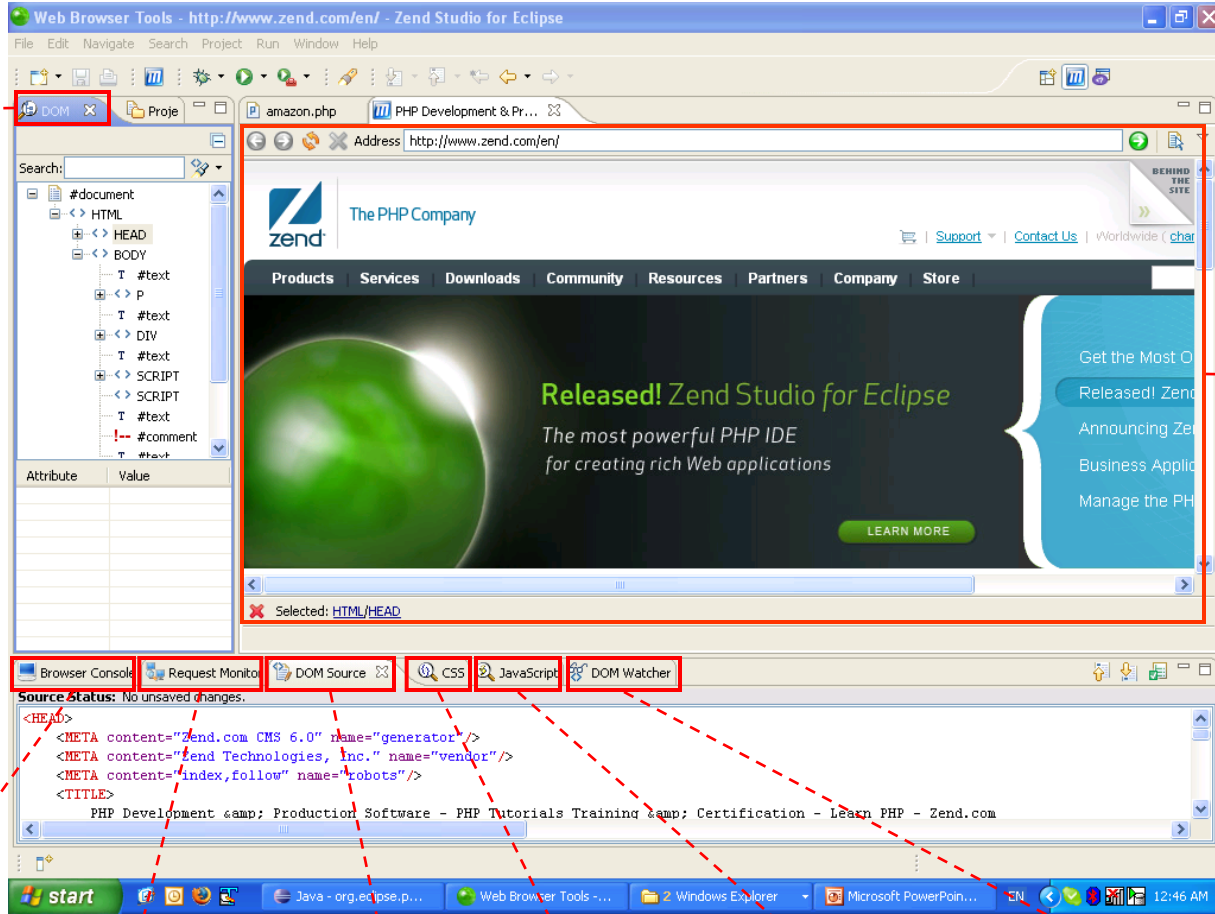
- **JavaScript Debugging**

- Local script
- Web page



# Embedded Mozilla Browsing

**DOM  
Inspector**



**Embedded  
Browser**

**Browser  
console**

**Request  
Monitor**

**DOM  
Source**

**CSS**

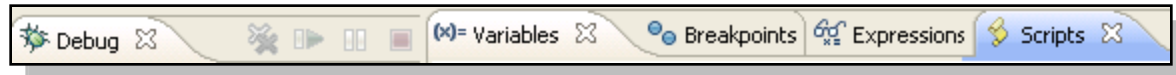
**JavaScript  
Evaluator**

**DOM  
Watcher**

# Debugger

- **Debug Perspective**

- **Breakpoints**



- Ability to set breakpoints
- Breakpoints can be enable/disable

- **Debug Views**

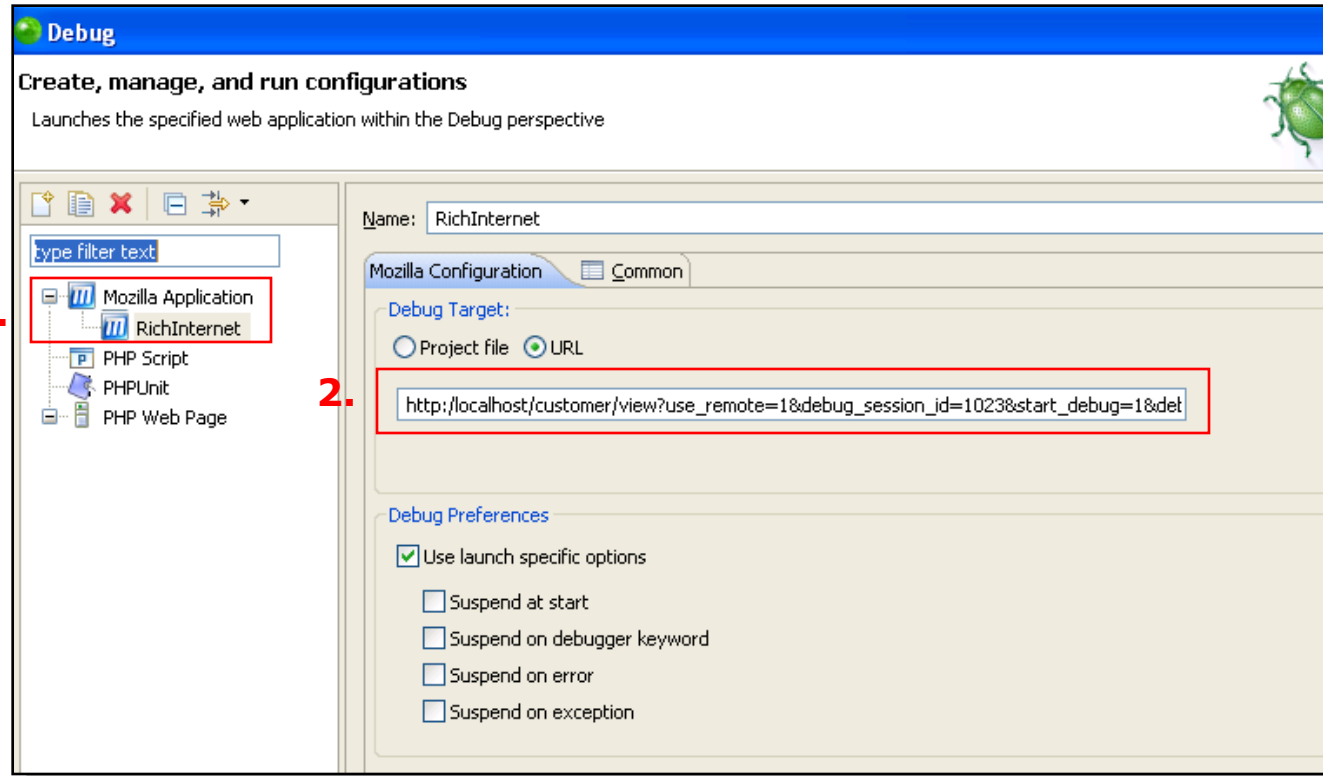
- Call Stack
- Variables
- Breakpoints
- Scripts View (for JavaScript)

- **Expression Support**

- **Debug Preference options**

# Debugger (PHP & Javascript)

- To debug both your server and browser during the same execution – copy the URL of the PHP debug to the URL field in the Mozilla launch





# Demonstration #2