

## SENSIOLABS

92-98 Boulevard Victor Hugo  
92115 CLICHY Cedex  
Tel: +33 1 40 99 82 05  
Fax: +33 1 40 99 82 30  
Web: <http://training.sensiolabs.com>

**Sensio**Labs



## GETTING STARTED WITH SYMFONY2

### Goals:

In two days only, discover the basics of the Symfony framework with the best experts. You'll learn to master the Symfony architecture, the MVC design pattern, the translation best practices and the techniques to build and validate forms.

**Duration:** 2 days - 14 hours

**Price:** €2,100.00 / person

**Code:** SF2C1

**Public:** PHP Web Developers

**Requirements:** Mastering OOP in PHP

### OBJECT ORIENTED PROGRAMMING (ON DEMAND)

- Classes and objects
- Properties and methods
- Constructor and destructor methods
- Class inheritance
- Attributes and methods scopes (visibility)
- Error handling with exceptions
- Namespaces

### GENERAL INTRODUCTION TO THE SYMFONY FRAMEWORK AND ITS TOOLKIT

- Introducing the Symfony framework and its ecosystem
- Installing the Symfony « Standard Edition Distribution »
- Discovering the Symfony project architecture (application, bundles, libraries, bundles etc.)
- Understanding the HTTP request processing workflow
- Understanding the concept of environment (development, production and testing)
- Introducing the coding and files naming conventions
- Discovering the concept of a « Bundle »

SensioLabs

92 / 98 bld Victor Hugo, 92115 Clichy Cedex, France  
+33 (0)140 99 82 05 - <http://sensiolabs.com>

## DEBUGGING THE CODE

- Getting help from the « Web Debug Toolbar »
- Introspecting the recorded logs
- Profiling HTTP requests thanks to the « Web Profiler » application
- Spotting problems thanks to explicit errors and exceptions stack traces

## CONFIGURING THE APPLICATION

- Comparing supported built-in configuration: XML, YAML and PHP
- Configuring the application thanks to global settings
- Configuration application's routes and URLs
- Overriding the configuration defined by a built-in or third-party « bundle »

## GENERATING PAGES FOR THE END USER

- Introducing the Twig templates rendering engine
- Comparing Twig and raw PHP templating strategies
- Accessing global variables from Twig templates (request, session, user)
- Using tags, filters and function to generate and format contents in a Twig template
- Making templates truly extensible thanks to the template inheritance feature
- Splitting templates into small templates chunks to avoid code duplications
- Rendering a Symfony controller from within a Twig template
- Generating URLs based on the application's routes configuration
- Rendering Symfony forms with custom Twig view helpers

## DESIGNING CONTROLLERS AND ANALYZING THE REQUEST

- Designing a controller class with its action methods
- Mapping URL patterns to Symfony controllers thanks to annotations configuration
- Generating a raw response or a response based on a Twig template
- Accessing client's information from the Request object
- Reading and writing session data
- Reading and writing cookies' data
- Triggering a 404 error page when the resource is not found
- Triggering an internal redirect to another controller to avoid code duplication
- Redirecting the user to another page

## INTERACTING WITH THE END USER THANKS TO FORMS

- Discovering the « Form » component
- Designing and processing simple web forms
- Adding validation constraints on submitted form fields' data
- Prototyping the form rendering thanks to Twig view helpers
- Gathering the form submitted data and processing them
- Sending the form validated and filtered data by e-mail to some recipient

## INTERNATIONALIZING AND LOCALIZING THE USER INTERFACE

- Discovering the « Translation » component
- Changing the default user's locale
- Defining custom abstract translation keys in Twig templates
- Generating XLIFF translation catalogues skeletons
- Translating abstract translation keys in XLIFF catalogues
- Dealing with dynamic translations including pluralized sentences