

# Laravel/Symfony Console

# Preparation

Install Composer: <https://getcomposer.org/download/>

Composer Mirror: `composer config -g repo.packagist composer https://packagist.phpcomposer.com`

Principle: <https://pkg.phpcomposer.com/>

``composer install``

php-cs-fixer: <https://packagist.org/packages/friendsofphp/php-cs-fixer>

# PHP CLI

- CGI/CLI
- Usage: Developing Shell Applications
- CLI Parameters: `$argv` *and* `$argc`
- `#!/usr/bin/php` *or* `php script.php`

# Review

- <http://www.php.net/cli>

# Composer

- `/vendor/autoload.php`
- `spl_autoload_register([ClassLoader, 'loadClass'])`
- Auto Class Loader
- `composer.json`
- Not `include/require/include_once/require_once`
- `packgist.org/packgist.com/toran proxy`

# Review

- <https://getcomposer.org/doc/>
- <http://php.net/manual/zh/language.oop5.autoload.php>
- <https://packagist.com/>
- <https://toranproxy.com/>

# Application Architecture

- CLI: Argv Input -> Console Kernel -> Terminal Output
- CGI: Http Request -> Http Kernel -> Response

# Request-Response Workflow

The web in action

The **User** asks for a **Resource** in a **Browser**

The **Browser** sends a **Request** to the **Server**

**Symfony** gives the **Developer** a **Request Object**

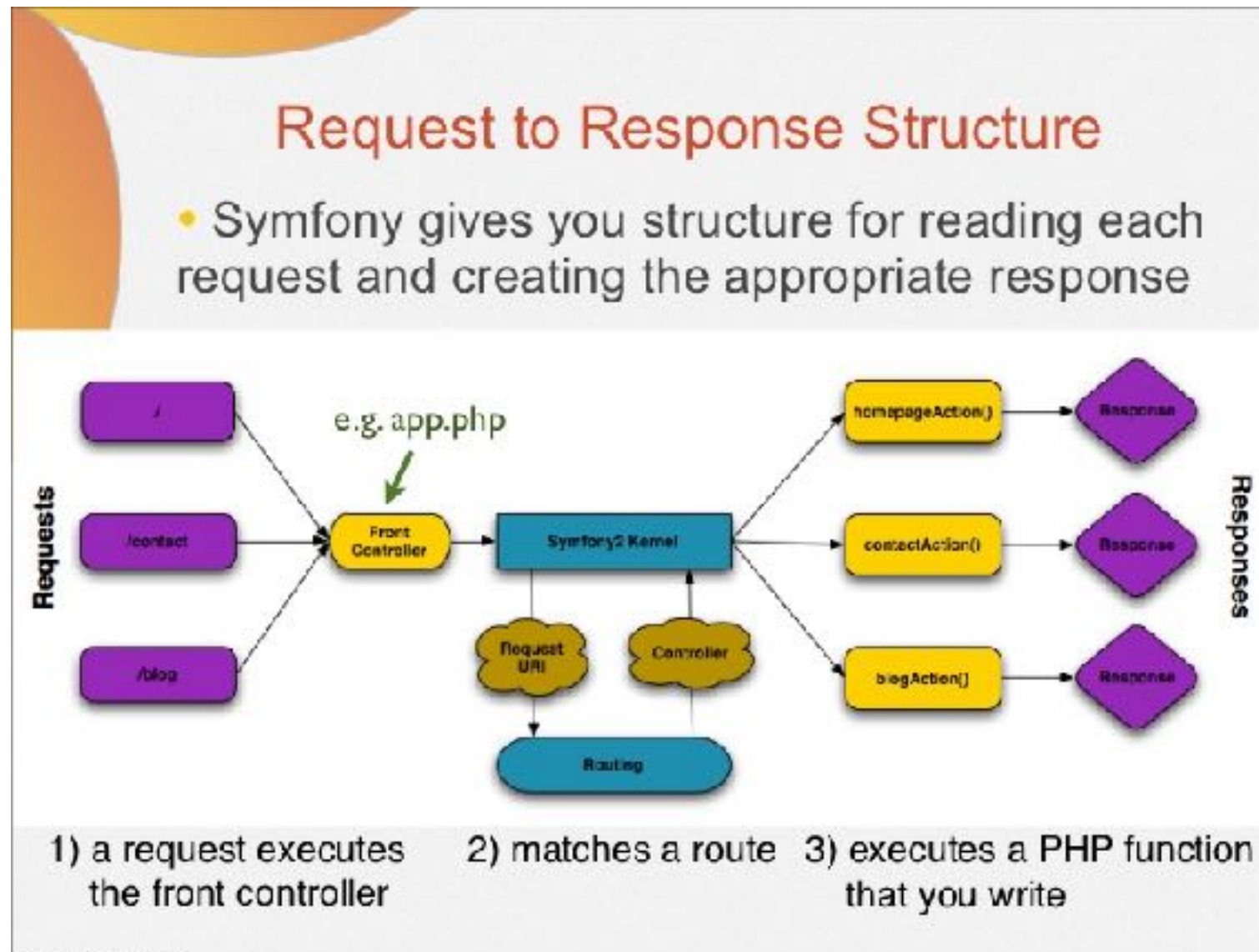
The **Developer** “converts” the **Request Object** to a **Response Object**

The **Server** sends back a **Response** to the **Browser**

The **Browser** displays the **Resource** to the **User**



# Request-Response Workflow



# Review

- [https://symfony.com/doc/current/components/http\\_kernel.html](https://symfony.com/doc/current/components/http_kernel.html) (very very good article)

# Console Kernel

- Boot Kernel.
- `handle($input, $output)`.
- `$kernel->getConsoleApplication()->run($argv_input, $output);`
- ‘*convert*’ input to output by kernel.
- Terminate Kernel.

# Console Application

- Add Multiple Commands: `$app->add($command)`
- Run a Command: `$app->run($input, $output)`
  - (1) resolve a command object by `$input`
  - (2) run a command
  - (3) output message by `$output`

# Review

- [https://symfony.com/doc/current/components/console/single\\_command\\_tool.html](https://symfony.com/doc/current/components/console/single_command_tool.html)

# Container

- \Illuminate\Foundation\Application::class
  - bind()
  - make()
- Dependency Injection
- Auto Build a *Console Kernel Object*

# Review

- [https://symfony.com/doc/current/service\\_container.html](https://symfony.com/doc/current/service_container.html)
- <https://laravel.com/docs/5.4/container>

# CLI Description Language

- <http://docopt.org/>



# Command Signature

- {name} {arguments} {options}
- Pass dynamic information by {arguments} or {options} to the {name} command
- Ordered, Required, Optional

# Console Input

- Arguments
- Options

# Arguments

- `\Symfony\Component\Console\Input\InputArgument::class`

# Options

- `\Symfony\Component\Console\Input\InputOption::class`

# Review

- <https://symfony.com/doc/current/console/input.html>
- [https://symfony.com/doc/current/components/console/console\\_arguments.html](https://symfony.com/doc/current/components/console/console_arguments.html)

# Console Output

- Color and Style Output

# Helpers

- Progress Bar Helper
- Table Helper
- Formatter Helper

# Review

- <https://symfony.com/doc/current/components/console/helpers/index.html>
- <https://symfony.com/doc/current/components/console/helpers/progressbar.html>
- <https://symfony.com/doc/current/components/console/helpers/table.html>
- <https://symfony.com/doc/current/components/console/helpers/formatterhelper.html>



# Principle of Command Execution

- `ConsoleApplication::class`
  - `getCommandName($input)`: get command name
  - `find($name)`: find command object
  - `run($input, $output)`: run specified command(if exists events)

# Principle of Command Execution

- Command::class
  - configure()
  - command lifecycle
    - initialize()
    - interact()
  - execute(\$input, \$output)

# Console Events

- Symfony
  - 'console.command'
  - 'console.error'
  - 'console.terminate'
- Laravel
  - \Illuminate\Console\Events\ArtisanStart::class

# Review

- <https://symfony.com/doc/current/components/console/events.html>

# Symfony/Laravel Command

- Symfony Command
- Laravel Command(dependency injection)

# Key Points

- CLI Application Architecture

# Talk

- Financial Software in America

# Q & A