Programowanie obiektowe w PHP 5

Hasin Hayder

Drogi Czytelniku! Poniżej zamieszczona jest errata do książki:
"Programowanie obiektowe w PHP 5"

Jest to lista błędów znalezionych po opublikowaniu książki, zgłoszonych i zaakceptowanych przez naszą redakcję. Pragniemy, aby nasze publikacje były wiarygodne i spełniały Twoje oczekiwania. Zapoznaj się z poniższą listą. Jeśli masz dodatkowe zastrzeżenia, możesz je zgłosić pod adresem https://helion.pl/user/erraty

Strona	Linia	Jest	Powinno
0	dedykacja	dedukuję	dedykuję
20	1	liczbę kodu	ilość kodu
23	9	asesorów	akcesorów
25	9 od dołu	tą strukturę	tę strukturę
32	4 od dołu	zmiennych i właściwości	właściwości i metod

33		Rozpoczynamy od otwarcia pliku class.emailer.php(czyli klasy Emailer) i zmieniamy deklarację zmiennej $sender. Po zmianie definicja zmiennej powinna być następują:
		
		protected $sender
		
		Następnie tworzymy kolejny plik o nazwie class.extendedemailer.php, w którym powinien znajdować się poniższy fragment kodu:
		
		<?
		class ExtendedEmailer extends Emailer
		
		function __construct()
		public function setSender($sender)
		
		$this->sender = $sender;
		
		?>
			Rozpoczynamy od otwarcia pliku class.emailer.php (czyli klasy Emailer) i zmieniamy deklarację zmiennej $sender oraz $recipients. Po zmianie definicja zmiennej powinna być następują:
			
			protected $sender;
			protected $recipients;
			
			Następnie tworzymy kolejny plik o nazwie class.extendedemailer.php, w którym powinien znajdować się poniższy fragment kodu:
			
			<?
			class ExtendedEmailer extends Emailer
			{
			function __construct()
			{
			$this->recipients = array();
			public function setSender($sender)
			{
			$this->sender = $sender;
			}
			}
			}
			?>

33	20	protected, to dostępdo niej można uzyskać tylko z poziomu podklasy	protected, to dostęp do niej można uzyskać z klasy, w której została zdefiniowana oraz we wszystkich jej podklasach
34	1 od dołu	class factorial	class Factorial
36	22	Programista może wyraźnie utworzyć destruktora	Programista może jawnie utworzyć destruktor
39	11	$headers .= ´From: $this->sender´ . "\r\n";	$headers .= "From: $this->sender" . "\r\n";
39	24	<? include_once("class.htmlemailer.php"); $hm = new HtmlEmailer(); 	<? include_once("class.emailer.php" include_once("class.htmlemailer.ph $hm = new HtmlEmailer("hasin@somewherein.ne
42	5	musi zawierać deklaracje	musi zawierać definicje
55	2	pacy	pracy
55	2 od dołu	Kiedy zachodzi potrzeba sprawdzenia, czy dana funkcja istnieje...	Kiedy zachodzi potrzeba sprawdzenia, czy dana klasa istnieje...
65	2 od dołu	użyteczne	użyteczna
65	36	użyteczne	użyteczna
86	6 od dołu	insertId()	insert_id()
126	16	ilość tych części	liczbę tych części

215	Rozdział9		Adres strony (http://code.google.com/p/orchidframew na której Autor umieszcza na bieżąco pliki do omówionej i stale rozbudowywanej struktury MVC.

Poniżej znajduje się lista błędów znalezionych przez czytelników, ale jeszcze nie potwierdzonych przez Redakcję:

Strona	Linia	Jest	Powinno

33	na dole strony	Rozpoczynamy od otwarcia pliku class.emailer.php (czyli klasy Emailer) i zmieniamy deklarację zmiennej$sender. Pozmianie definicjazmiennej powinnabyć następują: protected$sender Następnie tworzymy kolejny plik o nazwie class.extendedemailer.php, w którympowinien znajdować się poniższyfragment kodu: <? class ExtendedEmailer extends Emailer function __construct() public function setSender($sender) $this->sender = $sender;	Rozpoczynamy od otwarcia pliku class.emailer.php (czyli klasy Emailer) i zmieniamy deklarację zmiennej $sender oraz $recipients. Po zmianie definicja zmiennej powinna być następują: protected $sender; protected $recipients; Następnie tworzymy kolejny plik o nazwie class.extendedemailer.php, w którym powinien znajdować się poniższy fragment kodu: <? class ExtendedEmailer extends Emailer function __construct() $this->recipients = array(); public function setSender($sender) $this->sender = $sender;

34	5 od dołu	...to nazwa metody konstruktora będzie Emailer()	...to nazwa metody konstruktora będzie emailer()
38	1 i 2 od dołu	... rozszerzono przedstawioną wcześniej klasę Emailer oraz nadpisano funkcję sendEmail, która obecnie ma możliwość wysyłania wiadomości e-mail w formacie HTML	... rozszerzono przedstawioną wcześniej klasę Emailer o dodanie metody sendHTMLEmail, która ma możliwość wysyłania wiadomości e-mail w formacie HTML
39	11	$headers .= 'From: $this->sender' . "\r\n";	$headers .= "From: $this->sender" . "\r\n";
39	24–	<? include_once("class.htmlemailer.php"); $hm = new HtmlEmailer();	<? include_once("class.emailer.php"); include_once("class.htmlemailer.php" $hm = new HtmlEmailer("hasin@somewherein.net
102	22	//Przetwarzane kodu BBCode.	//Przetwarzane kodu emotikonów.
106	21	Fascade	Facade
129	3	$this->assertTrue($result);	$this->assertFalse($result);
138	8	Message field is not present	W formularzu brakuje pola o nazwie message.

226	**1**	**pomaga wczytywanie plików**	**pomaga we wczytywaniu plików**