Errata for *Intermediate Perl*

Submit your own errata for this product.

Print
Print

CONFIRMED ERRATA UNCONFIRMED ERRATA

The errata list is a list of errors and their corrections that were found after the product was released. If the error was corrected in a later version or reprint the date of the correction will be displayed in the column titled "Date Corrected".

The following errata were submitted by our customers and approved as valid errors by the author or editor.

Color Key: Serious Technical Mistake | Minor Technical Mistake | Language or formatting error | Typo | Question | Note | Update

Version	Location	Description	Submitted By	Date Submitted	Date Corrected
Printed	Page 2 5th paragraph	https://pause.perl.org/pause/authenquery?ACTION=request_id should be https://pause.perl.org/pause/query?ACTION=request_id Without the above change, authentication is required to access the web page (which is not possible for first time users). **Note from the Author or Editor:** Change as noted	aphilipp	Sep 03, 2012	Jun 07, 2013
Printed	Page 8 1st paragraph, 1st line	Change "that" to "what". **Note from the Author or Editor:** Change to "which"	aphilipp	Sep 04, 2012	Jun 07, 2013
Printed	Page 8 1st paragraph, 2nd line	"with which version of perl first included them" should maybe be something like "with which version of perl they were first included" **Note from the Author or Editor:** change to "which version of perl first included them", striking the "with"	aphilipp	Sep 04, 2012	Jun 07, 2013
Printed	Page 9 code section at bottom	The module documentation for File::Basename that is shown here is outdated as it was taken from the 1st edition of this book. Perhaps it could be updated to reflect the latest documentation. **Note from the Author or Editor:** We can do that. :)	aphilipp	Sep 04, 2012	Jun 07, 2013
PDF, Other Digital Version	Page 10 First paragraph, first sentence	The sentence starts "We�ve included on the top portion..." The word "on" should be "only". **Note from the Author or Editor:** Changed as noted in r546856.	Matt Perry	Aug 18, 2012	Jun 07, 2013
PDF, Other Digital Version	Page 11 First paragraph, first sentence	The sentence is a statement, not an inquiry, and should end with a period instead of a question mark. **Note from the Author or Editor:**	Matt Perry	Aug 30, 2012	Jun 07, 2013

		This is the sentence "However, suppose we already had a dirname subroutine?" Change that to a full stop at the end.			
Printed	Page 14 6th paragraph, 2nd line	Change "feature" to "features".	aphilipp	Sep 04, 2012	Jun 07, 2013
Printed	Page 14 8th paragraph	"Some distributions, such as Strawberry Perl (...) or ActivePerl (...), add extra modules to its distribution." should be "Some distributions, such as Strawberry Perl (...) and ActivePerl (...), add extra modules to their distributions." or the following (to avoid repeating the word "distributions") "Some distributions, such as Strawberry Perl (...) and ActivePerl (...), include extra modules." **Note from the Author or Editor:** Change as noted	aphilipp	Sep 04, 2012	Jun 07, 2013
PDF, Other Digital Version	Page 15 Sixth paragraph	The sentence that starts "If we have the a recent version of Perl, should also...". Remove "the" from the sentence and add "we" before "should". **Note from the Author or Editor:** That should be "If we have a recent version of Perl, we should"	Matt Perry	Aug 30, 2012	Jun 07, 2013
Printed	Page 15 2nd code example	Module::CoreList->first_release('Module::Build'); # 5.009004 should be print Module::CoreList->first_release('Module::Build'); # 5.009004 **Note from the Author or Editor:** Change as noted	aphilipp	Sep 09, 2012	Jun 07, 2013
Printed	Page 15 4th paragraph 4th line	the discription is different between the context and code: in context: Module::CoreLIst has been part of the Standard Library since v5.8.9 in code: #5.009004. I thought the context should be change to this: in context: Module::CoreLIst has been part of the Standard Library since v5.9.4 **Note from the Author or Editor:** Change next to last code example with % corelist Module::Build Module::CoreList was first released with perl v5.9.4 Change last code example to: % corelist Module::CoreList Module::CoreList was first released with perl v5.8.9	hanlei	Apr 12, 2013	Jun 07, 2013
Printed	Page 18 3rd paragraph	First a module is installed like this: % perl Makefile.PL INSTALL_BASE=/Users/home/Ginger Then it is explained how you can 'use lib qw(/Users/home/Ginger); to load the modules; This does not work, INSTALL_BASE creates a directory /Users/home/Ginger/lib' which is not searched by the 'use lib' statement. Obviously, the nicest way to do this is to use local::lib, which is also mentioned in the book. **Note from the Author or Editor:** Change as noted	Michiel Beijen	Sep 14, 2012	Jun 07, 2013
PDF, Other Digital Version	Page 20 First paragraph, first sentence	The first sentence says "to find and update when we need to move the file needs to a new..." The word "needs" should be removed. **Note from the Author or Editor:** Change to "This makes it easier to find and to update it when we need to move the file to a new system"	Matt Perry	Aug 30, 2012	Jun 07, 2013
Printed	Page 20 4th paragraph, 3rd line	Change "/home/gilligan/lib/" to "/Users/gilligan/lib/". **Note from the Author or Editor:** Change as noted	aphilipp	Sep 10, 2012	Jun 07, 2013

Printed	Page 23 5th code section, 5th line	Change "foo" to "perlstuff". **Note from the Author or Editor:** This is the line export PERL5LIB="/Users/Ginger/foo/lib/perl5/darwin&#8722;2level:/Users/Ginger/per Change that to export PERL5LIB="/Users/Ginger/perlstuff/lib/perl5/darwin&#8722;2level:/Users/Ginge	aphilipp	Sep 11, 2012	Jun 07, 2013
Printed	Page 23 2nd code example	Change "Set::Crossproduct" to "Set::CrossProduct".	aphilipp	Sep 12, 2012	Jun 07, 2013
Printed	Page 23 between 1st and 2nd paragraph	Change "the Perl" to "Perl". **Note from the Author or Editor:** In the note on that page, change as noted.	aphilipp	Sep 15, 2012	Jun 07, 2013
Printed	Page 24 Exercise 2	This is the line: "Write a program that reports the name and first release date for all the modules in Perl v5.14.2." Instead of reporting the first release date, the solution presented on page 313 reports the perl version when each module first appeared in core as ordered by the perl version number. **Note from the Author or Editor:** In the exercise, I need to convert the version to a date through the %released hash: $Module::CoreList::released{ Module::CoreList->first_release($module) };	aphilipp	Sep 17, 2012	Jun 07, 2013
Printed	Page 25, 106, 141, 143, 144 code examples	This is one of the lines: my @castaways = sort qw(Gilligan Skipper Ginger Professor Mary Ann); Because the qw shortcut treats each word that is separated by whitespace as a separate element in the array, we have the 2 elements "Mary" and "Ann" instead of the single element "Mary Ann". Perhaps, we can add a hyphen or underscore between "Mary" and "Ann" so that it is treated as a single element in the array. Another option is to use a list literal instead of the qw shortcut. **Note from the Author or Editor:** In the book source, there are dashes there but they didn't make it past the production tools. Odd.	aphilipp	Sep 13, 2012	Jun 07, 2013
Printed	Page 31 1st paragraph, 3rd line	Change "$SIG{_ _WARN_ _}" to "$SIG{__WARN__}". **Note from the Author or Editor:** The __ together looks as unclear in print unless we separate them by a hair space so they look like two characters. Either way there's a typography problem.	aphilipp	Sep 20, 2012	Jun 07, 2013
Printed	Page 35 4th paragraph, 1st and 2nd line	Change "Skipper" to "the Skipper". **Note from the Author or Editor:** Change as noted.	aphilipp	Sep 13, 2012	Jun 07, 2013
Printed	Page 36 3rd paragraph	Last sentence of the paragraph: "Thus, the grep checks each required item against the list." Is "grep" correct in this sentence?. I don't see the "grep" function used anywhere on this page. **Note from the Author or Editor:** Remove the sentence "Thus, the grep checks each required item against the list." This is left over from the example in the previous edition.	Neil Hainer	Aug 29, 2012	Jun 07, 2013
Printed	Page 42 2nd code example	This code example is supposed to be a rewrite of the code example on page 41, with the difference being that the braces were removed around @$items. But the code here also performs a grep instead of checking the %whos_items hash to determine the missing items. **Note from the Author or Editor:** The code in "Getting our braces off" is held over from the previous edition. It should now be:	aphilipp	Sep 22, 2012	Jun 07, 2013

		sub check_required_items {
		  my($who, $items) = @_;
		  my %whos_items = map {$_, 1} @$items;
		
		  my @required = qw(preserver sunscreen water_bottle jacket);
		  for my $item (@required) {
		    unless ($whos_items{$item}) { # not found in list?
		      print "$who is missing $item.\n";
		    }
		  }
		}			

Printed	Page 47 Figure 4-9	This is the line: "Figure 4-9. The PeGS structure for the %gilligan_info hash" It seems redundant to include the word "structure" since PeGS (i.e. "Perl Graphical Structure") already contains that word. In fact, the way PeGS is referred to throughout the book is inconsistent. Sometimes we see the text "PeGS diagram", other times "PeGS structure", and then just "PeGS". Perhaps its use can be standardized to one of these. **Note from the Author or Editor:** Change "PeGS structure" to "PeGS"	aphilipp	Jan 30, 2013	Jun 07, 2013
Printed	Page 48 Figure 4-10	Change "PeGs" to "PeGS".	aphilipp	Jan 29, 2013	Jun 07, 2013

Printed	Page 50, 51
	code examples
		foreach my $key (%$hash_ref) {
		
		should be
		
		foreach my $key (keys %$hash_ref) {
		
		Note from the Author or Editor:
		Change as noted
			aphilipp	Sep 24, 2012	Jun 07, 2013

Printed	Page 51 3rd code example	croak "I expected a hash reference!" unless eval { keys %$ref_type; 1 } should be croak "I expected a hash reference!" unless eval { keys %$hash_ref; 1 }; **Note from the Author or Editor:** Change as noted	aphilipp	Sep 24, 2012	Jun 07, 2013
Printed	Page 59 4th code example	This is the code section: my @temporary_name = (qw(blue_shirt hat jacket preserver sunscreen)); Is there a reason for the outer parentheses in this code? They seem unnecessary. **Note from the Author or Editor:** There's nothing extra that the outer parens do, and they probably shouldn't be there.	aphilipp	Sep 25, 2012	Jun 07, 2013
Printed	Page 67 Figure 5.2	In the PeGS diagram, change "gillign.crew.hut" to "gilligan.crew.hut". **Note from the Author or Editor:** Change as noted	aphilipp	Sep 29, 2012	Jun 07, 2013
Printed	Page 79 2nd paragraph, 1st line	Change "we can the string form of eval" to "we can use the string form of eval". **Note from the Author or Editor:** Change as noted	aphilipp	Oct 01, 2012	Jun 07, 2013
Printed	Page 85 3rd code section, 1st line	Change "--- #YAML:1.0" to "---". **Note from the Author or Editor:** Change as noted. The module's output is now different.	aphilipp	Oct 02, 2012	Jun 07, 2013
Printed	Page 100 1st paragraph, 2nd line	Change "referenced by subroutine" to "referenced by the subroutine". **Note from the Author or Editor:** Change as noted	aphilipp	Oct 19, 2012	Jun 07, 2013
Printed	Page 101 4th paragraph, 3rd line	Change "create_find_callback_that_counts" to "create_find_callbacks_that_sum_the_size". **Note from the Author or Editor:**	aphilipp	Oct 12, 2012	Jun 07, 2013

		This is at the end of the paragraph that starts "Distinguishing ..."			
Printed	Page 103 2nd code example	When I run this code example, I get the following error: "Undefined subroutine &main::find_by_name called at ./program.pl line 13." The example works though if the function name is fully qualified (i.e. File::Find::Closures::find_by_name). **Note from the Author or Editor:** I'd change this to import that sub: use File::Find::Closures qw(find_by_name);	aphilipp	Oct 13, 2012	Jun 07, 2013

Printed, PDF	Page 106
	last code example on the page
		Last code example on page 106, which explains using state to initialize array/hash refs. The only way I got this to work was to use anonymous array and hash constructors and rename %tab to $tab; see below. Was this a typo or am I not understanding the example?
		
		Thanks
		
		use v5.10;
		sub add_to_tab {
		my $castaway = shift;
		state $castaways = [qw(Ginger Mary Ann Gilligan)]; # works!
		state $tab = { map { $_, 0 } @$castaways }; # works!
		$tab&#8722;>{$castaway}++;
		}
		
		Note from the Author or Editor:
		The line in the last code example is wrong. It's unchanged from the previous example, and should now be a hash and array references as you note. Wrap them in the appropriate reference constructors:
		
		sub add_to_tab {
		my $castaway = shift;
		state $castaways = [qw(Ginger Mary&#8722;Ann Gilligan)]; # works!
		state $tab = { map { $_, 0 } @$castaways }; # works!
		$tab&#8722;>{$castaway}++;
		}
			Anonymous	Sep 08, 2012	Jun 07, 2013

Printed	Page 106 3rd code example	Assuming that we are sorting lines of text, change "<=>" to "cmp". **Note from the Author or Editor:** Change as noted	aphilipp	Oct 20, 2012	Jun 07, 2013
Printed	Page 107 4th paragraph, 2nd line	If possible, the amount of space should be reduced between the underscores in "_ _SUB_ _". **Note from the Author or Editor:** There's not really a space there, but we tried to insert a hair space to show that there are two underscores to type. It didn't come out that well though.	aphilipp	Oct 20, 2012	Jun 07, 2013
Printed	Page 109 1st code example, 12th line	The closing curly brace is not properly aligned. It should be indented at the same level as the line that follows. **Note from the Author or Editor:** Add a single space before that brace	aphilipp	Oct 21, 2012	Jun 07, 2013

Printed	Page 109-112
	output of code examples
		The code returned by coderef2text is not consistent between the code examples. For instance, on pages 109 and 110, the outputs contain:
		
		use warnings;
		use strict;
		no feature;
		use feature ':5.16';
		
		But on pages 110 to 112, the outputs contain:
		
		use warnings;
		use strict 'refs';
		BEGIN {
		$^H{'feature_unicode'} = q(1);
		$^H{'feature_say'} = q(1);
		$^H{'feature_state'} = q(1);
		$^H{'feature_switch'} = q(1);
		}
		
		I was able to duplicate both these outputs depending on the version of my Perl installation. The book should probably choose one of them for all the examples.
		
		Note from the Author or Editor:
		It looks like I ran those examples with different versions of Perl. I've made then the first version.
			aphilipp	Oct 21, 2012	Jun 07, 2013

Printed	Page 110 2nd code example, 13th line	This line of code should be indented at the same level as the previous line. In other words:	aphilipp	Oct 21, 2012	Jun 07, 2013

		require Data::Dumper;
		local $Data::Dumper::Terse = 1;
		should be
		require Data::Dumper;
		local $Data::Dumper::Terse = 1;
		Note from the Author or Editor:
		undent that line as noted.			

Printed	Page 120 5th paragraph	"We declare the $fh variable with state ..." should be "We declare the $fhs variable with state ..." **Note from the Author or Editor:** Change as noted, adding an 's' to the variable name.	Anonymous	May 26, 2013	Jun 07, 2013
Printed	Page 121 4th line in the note	Change "preface" to "the introduction". **Note from the Author or Editor:** Change as noted	aphilipp	Oct 24, 2012	Jun 07, 2013
Printed	Page 123 1st code example	This code example would be more appropriate if $string_log contained some lines of text. **Note from the Author or Editor:** Change to my $string_log = '...'; # assume many lines	aphilipp	Oct 27, 2012	Jun 07, 2013
Printed	Page 124 4th paragraph, 2nd line	Change "hands" to "handles". **Note from the Author or Editor:** Change as noted	aphilipp	Oct 27, 2012	Jun 07, 2013
Printed	Page 124 3rd paragraph, 2nd line	Change "read from the output from a command" to "read the output of a command". **Note from the Author or Editor:** Change as noted	aphilipp	Oct 27, 2012	Jun 07, 2013
Printed	Page 127 Exercise 2	This is the sentence: "Use a hash keyed by castaway name and whose values are IO::File objects for each output file." Instead of using IO::File, the solution presented on page 325 uses open. **Note from the Author or Editor:** Change to Hint: Use a hash keyed by castaway name and whose values are filehandle references for each output file.	aphilipp	Oct 31, 2012	Jun 07, 2013
ePub	Page 129 code examples under "Regexes as Scalars"	1. Why do you have the "(?: ...)?" part of the regexes in @patterns/%patterns, when they add no demonstrated value (except as an example)? They are redundant, are they not (extra work for no gain)? 2. Also in that code block, the period in the "Mrs?." part should probably be escaped thus: "Mrs?\."? This latter oversight is repeated throughout "Assembling Regular Expressions" around page 134. confess("As a user of en-GB, it's rare to see a period after Mr/Mrs, these days, so perhaps make the period optional, too?"); **Note from the Author or Editor:** The non-capturing parentheses allow you to apply quantifiers to part of the pattern. For instance, we can match 'Willie Gilligan' or 'Gilligan' with no 'Willie'. Although an optional portion at the beginning seems like it does nothing, if you put it together with other sub-expressions, it can matter. We didn't say anything about that in that section though. I fixed the Mrs\.., even though I'm used to seeing Mrs with no period as well.	Geraint	Oct 18, 2012	Jun 07, 2013
Printed	Page 130 2nd code section, 6th line	Change "Fred" to "Gilligan". **Note from the Author or Editor:** Change as noted	aphilipp	Oct 31, 2012	Jun 07, 2013
Printed	Page 130 4th code example	This is the line:	aphilipp	Nov 01, 2012	Jun 07, 2013

		return defined $@ ? 0 : 1; If there is no error after the eval operation, $@ will be empty (and not undefined). Therefore this statement always returns 0 as $@ will always be defined. **Note from the Author or Editor:** Change to return length $@ ? 0 : 1; I think I meant to do that based on the preceding paragraph.			
Printed	Page 131 6th code example	This is the line: my $regex = qr'$var'; # four characters, not two We do not know the value of $var as it is not assigned in this example. So why does the comment say "not two"? **Note from the Author or Editor:** Good point. We should fix that comment.	aphilipp	Nov 01, 2012	Jun 07, 2013
Printed	Page 131 8th paragraph	This paragraph shows "end of string" written both with and without hyphens. For consistency, the text should use one form throughout. **Note from the Author or Editor:** Use hyphens in both cases since they are adjectives.	aphilipp	Nov 01, 2012	Jun 07, 2013
Printed	Page 132 3rd paragraph, 2nd line	"flags that affect the pattern and flags that affected the operator" should probably be "flags that affect the pattern and flags that affect the operator" **Note from the Author or Editor:** Change as noted	aphilipp	Nov 02, 2012	Jun 07, 2013
Printed	Page 132 5th paragraph, 2nd line	Change "allow" to "allows". **Note from the Author or Editor:** Change as noted	aphilipp	Nov 02, 2012	Jun 07, 2013
Printed	Page 135 1st code example, 9th line	The return statement should be indented. **Note from the Author or Editor:** Change as noted	aphilipp	Nov 02, 2012	Jun 07, 2013
Printed	Page 137-138 code examples	Within the while loops, the default variable $_ contains each line of text terminated by a newline. Therefore change "say" to "print" in the while loops. **Note from the Author or Editor:** Change as noted	aphilipp	Nov 03, 2012	Jun 07, 2013
Printed	Page 138 1st code example, 4th line	This is the line: print if /$RE{URL}{HTTP}/; Change to: print if /$RE{URI}{HTTP}/; **Note from the Author or Editor:** Change as noted	aphilipp	Nov 03, 2012	Jun 07, 2013

Printed	Page 139
	2nd code example, 3rd line
		This is the line:
		while(<DATA>) {
		To be consistent with the previous examples, change to:
		while(<>) {
		Note from the Author or Editor:
		Change as noted
			aphilipp	Nov 03, 2012	Jun 07, 2013

Printed	Page 141 1st footnote, 2nd line	Change "Somes" to "Some". **Note from the Author or Editor:** Change as noted	aphilipp	Nov 05, 2012	Jun 07, 2013
Printed	Page 152 3nd code example,	This is the line:	aphilipp	Nov 06, 2012	Jun 07, 2013

	9th line	dump_data_for_path("$path/$_", $directory{$_});
		Change to:
		dump_data_for_path("$path/$_", $data->{$_});
		Note from the Author or Editor:
		Change as noted			

Printed	Page 154 1st code example	This code example generates the following warning: "Parentheses missing around "my" list at ./test.pl line 27." The line of code responsible for the warning is: opendir my $dh, $path; One workaround is to put parentheses around the first lexical variable: opendir my ($dh), $path; **Note from the Author or Editor:** Change as noted to avoid the warning	aphilipp	Nov 24, 2012	Jun 07, 2013
Printed	Page 155 2nd code example	This code example also generates the warning mentioned in the errata for page 154: "Parentheses missing around "my" list at ./test.pl line 27." One workaround is to put parentheses around the first lexical variable in the opendir statement: opendir my ($dh), $path; **Note from the Author or Editor:** Change as noted	aphilipp	Nov 24, 2012	Jun 07, 2013
Printed	Page 157 Exercise 5	The sample code in the exercise description shows slightly different arguments being passed to the data_for_path subroutine than what is shown in the solutions on pages 332 and 333. The type argument in the sample code is either "depth-first" or "breadth-first", while the solutions test for equality with just "depth". data_for_path in the solution does not expect a threshold parameter. Yet, data_for_path still calculates and keeps track of the level (even though it is otherwise not being used). **Note from the Author or Editor:** One pages 332 and 333, there are three instances of $type eq 'depth' Change each to $type eq 'depth-first'	aphilipp	Dec 08, 2012	Jun 07, 2013
Printed	Page 163 1st line in the note	Change "tracked" to "track".	aphilipp	Dec 11, 2012	Jun 07, 2013
Printed	Page 168 3rd paragraph, 2nd sentence	This is the sentence: "Most programs leave the package at the default main package." Change "at" to "as".	aphilipp	Dec 11, 2012	Jun 07, 2013
Printed	Page 177 3rd and 4th code examples	The example line commands that run module-starter need to specify the module name (i.e. --module=Animal). **Note from the Author or Editor:** For each command line on this page, it should start with % module-starter --module=Animal The "--module=Animal" is missing from the bottom two.	aphilipp	Dec 17, 2012	Jun 07, 2013
Printed	Page 178 2nd paragraph, 2nd line	Change "distribution" to "distributions".	aphilipp	Dec 13, 2012	Jun 07, 2013
Printed	Page 179 1st paragraph, 4th line	Change "the MANIFEST files helps" to "the MANIFEST file helps".	aphilipp	Dec 18, 2012	Jun 07, 2013
Printed	Page 179 5th paragraph, 4th line	Change "all of the test" to "all of the tests".	aphilipp	Dec 13, 2012	Jun 07, 2013
Printed	Page 180 3rd paragraph, 2nd	Change "files that contains" to "files that contain".	aphilipp	Dec 13, 2012	Jun 07, 2013

	line				
Printed	Page 181 4th paragraph, 2nd line	Change "The Module::Starter::AddModule can do the job" to "Module::Starter::AddModule can do the job".	aphilipp	Dec 13, 2012	Jun 07, 2013
Printed	Page 181 4th paragraph, 1st line	Change "created" to "create".	aphilipp	Dec 13, 2012	Jun 07, 2013
Printed	Page 181 4th paragraph, 2nd line	Remove "which" from "but which we have to install it".	aphilipp	Dec 13, 2012	Jun 07, 2013
Printed	Page 185 1st paragraph, 2nd sentence	This is the sentence: "The =headn directories specify a heading." The word "directories" does not seem to fit in this sentence. I think the intended word was "directives". **Note from the Author or Editor:** Change "directories" to "directives"	aphilipp	Dec 13, 2012	Jun 07, 2013
Printed	Page 185 1st paragraph, 2nd line	Change "header" to "heading".	aphilipp	Dec 13, 2012	Jun 07, 2013
Printed	Page 188 2nd code example	This is the line: % module-starter --mb --name="Animal" Change "--name" to "--module". Author and email should also be specified. **Note from the Author or Editor:** Change as noted	aphilipp	Dec 20, 2012	Jun 07, 2013
Printed	Page 189 5th code example	This is the line: % module-starter --builder="ExtUtils::Makemaker" --name="Animal" Change "--name" to "--module". Author and email should also be specified. **Note from the Author or Editor:** Change to % module-starter --builder="ExtUtils::Makemaker" --module="Animal"	aphilipp	Dec 20, 2012	Jun 07, 2013
Printed	Page 192 1st code example	This is the line: % module-starter --module=Cow,Horse,Sheep Since we are adding modules to an existing distribution, the --dist argument should be specified as follows: % module-starter --module=Cow,Horse,Sheep --dist=. **Note from the Author or Editor:** Change as noted	aphilipp	Dec 22, 2012	Jun 07, 2013
Printed	Page 198 8th paragraph, 3rd sentence	Change "We override a the method" to "We override the method".	aphilipp	Dec 22, 2012	Jun 07, 2013

Printed	Page 202
	2nd code example, 2nd line
		This is the line:
		use parent qw(Cow);
		Change to:
		use parent qw(Animal);
		Note from the Author or Editor:
		Change as noted
			aphilipp	Dec 23, 2012	Jun 07, 2013

Printed	Page 202 5th code example, 6th line	This is the line: $class->Animal::speak(@_); # tell it where to start Since this code example is supposed to represent the best practices so far, change to: $class->SUPER::speak(@_); # tell it where to start **Note from the Author or Editor:** Change as noted	aphilipp	Dec 22, 2012	Jun 07, 2013
Printed	Page 202 2nd code example	This is the code example:	aphilipp	Dec 22, 2012	Jun 07, 2013

		package Cow;
		use parent qw(Cow);
		sub sound { 'neigh' }
		1;
		
		To be consistent with previous examples in the chapter, change 'neigh' to 'moooo'.
		
		Note from the Author or Editor:
		Change as noted			

Printed	Page 210 4th paragraph	The following sentence should be rewritten: "We try it with works with its default variable, $_:" **Note from the Author or Editor:** Change to "We try it with its default variable, $_:", removing the "with works"	aphilipp	Jan 13, 2013	Jun 07, 2013
Printed	Page 212 4th paragraph, 1st line	Change "procedes" to "proceeds".	aphilipp	Jan 07, 2013	Jun 07, 2013

Printed	Page 213
	2nd code example, 4th line
		When I run the test program, I get the following output:
		
		$ perl -Iblib/lib -T t/00-load.t
		You said to run 0 tests at t/00-load.t line 8.
		BEGIN failed--compilation aborted at t/00-load.t line 8.
		
		This is line:
		
		use Test::More tests => scalar @classes;
		
		The program runs fine though if the value is hardcoded:
		
		use Test::More tests => 5;
		
		Note from the Author or Editor:
		Change that code to
		
		#!perl &#8722;T
		
		BEGIN {
		my @classes = qw(Animal Cow Sheep Horse Mouse);
		use Test::More;
		plan tests => scalar @classes;
		
		foreach my $class (@classes) {
		use_ok($class) or print "Bail out! $class did not load!\n"
		}
		}
			aphilipp	Jan 07, 2013	Jun 07, 2013

Printed	Page 213 3rd code example	This is the code example: % ./Build % perl -Iblib/lib -T t/00-load.t 1..4 ok 1 - use Animal; ok 2 - use Cow; ok 3 - use Sheep; ok 4 - use Horse; Change to: % ./Build % perl -Iblib/lib -T t/00-load.t 1..5 ok 1 - use Animal; ok 2 - use Cow; ok 3 - use Sheep; ok 4 - use Horse; ok 5 - use Mouse; **Note from the Author or Editor:** Change as noted, adding that last line	aphilipp	Jan 07, 2013	Jun 07, 2013

Printed	Page 214
	3rd code example, 5th line
		This is the line:
		
		use Test::More tests => 6;
		
		Change to:
		
		use Test::More tests => 7;
			aphilipp	Jan 14, 2013	Jun 07, 2013

Printed	Page 215 3rd code section, 3rd line	Change "1..6" to "1..7".	aphilipp	Jan 08, 2013	Jun 07, 2013
Printed	Page 215 3rd code section, 2nd line	This is the line: % perl -Iblib/lib t/boilerplate.t	aphilipp	Jan 08, 2013	Jun 07, 2013

		Change to: % perl -Iblib/lib -T t/boilerplate.t			
Printed	Page 215 3rd paragraph, 1st line	Change "the tests appears" to "the tests appear".	aphilipp	Mar 10, 2013	Jun 07, 2013
Printed	Page 216 2nd code section, 21st line	This is the line: t/boilerplate.t (Wstat: 0 Tests: 6 Failed: 0) Change to: t/boilerplate.t (Wstat: 0 Tests: 7 Failed: 0) **Note from the Author or Editor:** Change as noted	aphilipp	Jan 15, 2013	Jun 07, 2013
Printed	Page 216 1st code section, 3rd line	This is the line: t/boilerplate.t (Wstat: 0 Tests: 6 Failed: 0) Change to: t/boilerplate.t (Wstat: 0 Tests: 7 Failed: 0)	aphilipp	Jan 08, 2013	Jun 07, 2013
Printed	Page 218 3rd code example, 8th line	This is the line: like($at, qr/You must/, 'sound() dies with a message'); Change to: like($at, qr/You have/, 'sound() dies with a message'); The above change should also be applied to the examples on pages 219, 220, 342, and 343. **Note from the Author or Editor:** Change as noted	aphilipp	Jan 15, 2013	Jun 07, 2013

Printed	Page 219
	1st code example
		This section of code gives a warning as it declares the same "my" variable twice in the same scope:
		  # check that sound() dies
		  eval { Animal->sound() } or my $at = $@;
		  like($at, qr/You must/, 'sound() dies with a message');
		  # check that speak() dies too
		  eval { Animal->speak() } or my $at = $@;
		  like($at, qr/You must/, 'speak() dies with a message');
		One solution is to use a different variable in the second part of this example. Any change to the above code should also be applied to the examples on pages 220, 342, and 343.
		Note from the Author or Editor:
		Add braces around each couplet in both pages 220 and 342
		{
		# check that sound() dies
		eval { Animal->sound() } or my $at = $@;
		like($at, qr/You must/, 'sound() dies with a message');
		}
		{
		# check that speak() dies too
		eval { Animal->speak() } or my $at = $@;
		like($at, qr/You must/, 'speak() dies with a message');
		}
			aphilipp	Jan 15, 2013	Jun 07, 2013

Printed	Page 219
	test whether 'An Animal subclass does the right thing'
		use strict;
		use warnings;
		use Test::More tests => 6;
		# same as before
		...
		{
		package Foofle;
		use parent qw(Animal);
		sub sound { 'foof' }
		is(
		Foofle&#8722;>speak,
		"A Foofle goes foof!\n",
		'An Animal subclass does the right thing'
		);
		}
			galeb	Jan 14, 2013	Jun 07, 2013

		This tests for a method that prints to screen, so Foofle->speak gives '1' and not "A Foofle goes foof!\n" in the is() test. Please clarify how to test a method that prints to screen, as I'm not getting it to work as written. Also, since wrapped in the bare block, shouldn't is() be Test::More::is() ?
		Thanks
		
		Note from the Author or Editor:
		Change the code example to:
		
		use strict;
		use warnings;
		use Test::More tests => 6;
		
		# same as before
		
		...
		
		{
		package Foofle;
		use parent qw(Animal);
		sub sound { 'foof' }
		}
		
		ok(Foofle->speak, 'An Animal subclass does the right thing');			

Printed	Page 221
	2nd code section
		This is the code section:
		
		if( ) { ... }
		elsif() { ... }
		else { ... }
		
		unless()
		elsif() { ... }
		else { ... }
		
		give(...) {
		  when { ... }
		  when { ... }
		  when { ... }
		}
		
		Change to:
		
		if() { ... }
		elsif() { ... }
		else { ... }
		
		unless() { ... }
		elsif() { ... }
		else { ... }
		
		given() {
		  when() { ... }
		  when() { ... }
		  when() { ... }
		  default { ... }
		}
		
		Note from the Author or Editor:
		Add "n" to make "give" "given", and add "default {...}" in that block.
			aphilipp	Jan 17, 2013	Jun 07, 2013

Printed	Page 225
	2nd code example
		This is the code example:
		
		package Horse;
		@ISA = qw(Animal);
		sub sound { 'neigh' }
		
		Change to:
		
		package Horse;
		use Animal;
		our @ISA = qw(Animal);
		sub sound { 'neigh' }
		
		Or this:
		
		package Horse;
		use parent qw(Animal);
		sub sound { 'neigh' }
		
		Note from the Author or Editor:
		Change second example on 225 to
		
		package Horse;
		use parent qw(Animal);
		sub sound { 'neigh' }
			aphilipp	Jan 30, 2013	Jun 07, 2013

Printed	Page 226 Figure 15-1	According to the title description, Figure 15-1 should show the PeGS for a generic object. Yet the diagram in Figure 15-1 shows the PeGS for $tv_horse, identical to that of Figure 15-2. Also, the text that references Figure 15-1 expects a PeGS diagram showing the structure for $tv_horse. **Note from the Author or Editor:**	aphilipp	Jan 29, 2013	Jun 07, 2013

		Figure 15-1 should just be the top half of that image, and Figure 15-2 should be the bottom half.			
Printed	Page 226 Figure 15-1	Change "PeGs" to "PeGS".	aphilipp	Jan 29, 2013	Jun 07, 2013
Printed	Page 227 Figure 15-2	Change "PeGs" to "PeGS".	aphilipp	Jan 29, 2013	Jun 07, 2013

Printed	Page 228
	4th code example
		This is the code example:
		
		  package Horse;
		    use parent qw(Animal);
		    sub sound { 'neigh' }
		    sub name {
		      my $self = shift;
		      $$self;
		    }
		    sub named {
		      my $class = shift;
		      my $name = shift;
		      bless \$name, $class;
		    }
		
		The indentation for all lines after the first line should be reduced by 2 spaces.
		
		Note from the Author or Editor:
		Change as noted
			aphilipp	Jan 29, 2013	Jun 07, 2013

Printed	Page 228 1st paragraph, 2nd sentence	This is the sentence: "neigh is the return value, which ends up as the earlier $noise variable." Change "as" to "in".	aphilipp	Jan 30, 2013	Jun 07, 2013

Printed	Page 230
	4th code example
		This is the code example:
		
		  package Animal;
		    sub named {
		      my($class, $name) = @_;
		      bless \$name, $class;
		    }
		    sub name {
		      my $either = shift;
		      ref $either
		        ? $$either # it's an instance, return name
		        : "an unnamed $either"; # it's a class, return generic
		    }
		    sub speak {
		      my $either = shift;
		      print $either->name, ' goes ', $either->sound, "\n";
		    }
		    sub eat {
		      my $either = shift;
		      my $food = shift;
		      print $either->name, " eats $food.\n";
		    }
		
		The indentation for all lines after the first line should be reduced by 2 spaces.
		
		Note from the Author or Editor:
		Change as noted
			aphilipp	Jan 29, 2013	Jun 07, 2013

ePub	Page 232 section "custom import routines" first 2 'sub import...' examples	*{"main::$name"}=\&$_; should be *{"main::$name"}=\&$name; and *{$package."::$name"}=\&$_; should be *{$package."::$name"}=\&$name; **Note from the Author or Editor:** Change as noted	Gedge	Oct 20, 2012	Jun 07, 2013
Printed	Page 237 1st paragraph, 7th line	Change "where they called wrong method" to "where they called the wrong method".	aphilipp	Jan 29, 2013	Jun 07, 2013
Printed	Page 247 Exercise 1	This is the sentence: "Write a script that uses your module and prints the values for the date, month, and year." In order to match the beginning of the exercise description, change "date" to "day". This change should also be reflected in the solutions on pages 346 and 347. **Note from the Author or Editor:**	aphilipp	Feb 02, 2013	Jun 07, 2013

		One page 247, exercise 1, change "named day" to "named date". Ignore the issues on the other pages.			
Printed	Page 249 1st paragraph, 2nd sentence	This is the sentence: "Now we're going to show how to get our own modules do that." Change to: "Now we're going to show how to get our own modules to do that." **Note from the Author or Editor:** Add a "to" in front of "do"	aphilipp	Feb 05, 2013	Jun 07, 2013
Printed	Page 250 3rd paragraph, 3rd sentence	This is the part of the sentence: "(e.g., File::Basename to the importing namespace (e.g., main)." Change to: "(e.g., File::Basename) to the importing namespace (e.g., main)."	aphilipp	Feb 05, 2013	Jun 07, 2013
Printed	Page 253 2nd paragraph, 2nd line	Change "theExporter" to "the Exporter".	aphilipp	Feb 05, 2013	Jun 07, 2013
Printed	Page 258 2nd code example, 1st line	This is the line: ## include animal classes from previous chapter... Change "previous chapter" to "Chapter 15".	aphilipp	Feb 11, 2013	Jun 07, 2013
Printed	Page 276 3rd paragraph, 2nd sentence	This is the sentence: "When we do that, we don't need a default speak subroutine because we can use the role features to denote that all classes using this role need to define their own sound:" Shouldn't "speak subroutine" really be "sound subroutine"? **Note from the Author or Editor:** Change "speak subroutine" to "sound subroutine"	aphilipp	Mar 14, 2013	Jun 07, 2013
Printed	Page 276 2nd code example	The following line should be added: use namespace::autoclean; **Note from the Author or Editor:** After "use Moose::Role;", add the noted line.	aphilipp	Mar 15, 2013	Jun 07, 2013

Printed	Page 276
	3rd code example
		The following lines should be added:
		use namespace::autoclean;
		__PACKAGE__->meta->make_immutable;
		Note from the Author or Editor:
		The complete code should be:
		package Horse;
		use Moose;
		use namespace::autoclean;
		with 'Animal';
		sub sound { 'neigh' }
		__PACKAGE__->meta->make_immutable;
		1;
			aphilipp	Mar 15, 2013	Jun 07, 2013

Printed	Page 285 5th paragraph, 1st sentence	This is the part of the sentence: "it outputs special ok messages so keep the test numbering right" Change to: "it outputs special ok messages to keep the test numbering right" **Note from the Author or Editor:** change "so" to "to"	aphilipp	Apr 07, 2013	Jun 07, 2013
Printed	Page 286 3rd code example, 8th & 9th lines	These are the lines: is($trigger->name, 'Trigger', 'Trigger's name is correct'); is($tv_horse->name, 'Mr. Ed', 'Mr. Ed's name is correct');	aphilipp	Apr 07, 2013	Jun 07, 2013

		Change to: is($trigger->name, 'Trigger', 'Trigger\'s name is correct'); is($tv_horse->name, 'Mr. Ed', 'Mr. Ed\'s name is correct'); or is($trigger->name, 'Trigger', "Trigger's name is correct"); is($tv_horse->name, 'Mr. Ed', "Mr. Ed's name is correct"); **Note from the Author or Editor:** Use the last example, the one with the "			
Printed	Page 288 1st code example	These are the lines: like($at, qr/You must/, 'sound() dies with a message'); like($at, qr/You must/, 'speak() dies with a message'); Change to: like($at, qr/You have/, 'sound() dies with a message'); like($at, qr/You have/, 'speak() dies with a message'); **Note from the Author or Editor:** These are two lines in different places in the first code block. Change "must" to "have"	aphilipp	Mar 16, 2013	Jun 07, 2013

Printed	Page 291
	2nd code example, 5th line
		This is the line:
		
		stdout_is(\&print_hello, "Welcome Aboard\n");
		
		Change to:
		
		stdout_is(\&print_hello, "Welcome Aboard!\n");
		
		Note from the Author or Editor:
		Adds exclamation point
			aphilipp	Apr 07, 2013	Jun 07, 2013

Printed	Page 291 2nd paragraph, 1st sentence	This is the sentence: "If we create temporary files like this, we need to remember that our current directory is the same as the test script (even if we're running make test from the parent directory)." When running the example from the preceding page, I am finding that the temporary file is being created in the directory from which ./Build test is being run. Therefore the current directory is the same as the directory from which ./Build test is being run, and not the same as the test script. **Note from the Author or Editor:** Change the first sentence of the second paragraph on page 291 to: "If we create temporary files like this, we need to remember that our current directory is the same as the build script, not the location of the test program."	aphilipp	Apr 07, 2013	Jun 07, 2013

Printed	Page 292
	3rd code example
		This is the code example:
		
		  use Test::More;
		  use Test::Warn;
		  sub add_letters { "Skipper" + "Gilligan" }
		  warning_like { add_letters() }, qr/nonnumeric/;
		  done_testing();
		
		There should not be a comma in the warning_like statement between BLOCK and REGEXP. warning_like tests that a block gives exactly one warning and that it can be matched by the given regular expression. Because we have 2 warnings (one for each argument in the addition), we should use warnings_like instead. Also, the string "nonnumeric" does not appear in the warnings. Taking all this into account, the above code example could be rewritten as:
		
		  use Test::More;
		  use Test::Warn;
		  sub add_letters { "Skipper" + "Gilligan" }
		  warnings_like { add_letters() } [qr/numeric/, qr/numeric/];
		  done_testing();
		
		Note from the Author or Editor:
		Remove the comma in the warnings_like line as noted
			aphilipp	Apr 07, 2013	Jun 07, 2013

Printed	Page 293 4th paragraph, 3rd line	Change "retreive" to "retrieve".	aphilipp	Mar 15, 2013	Jun 07, 2013

Printed	Page 294
	2nd code example
		This is the code example:
		
		  sub check_required_items {
		    my $who = shift;
		    my %whos_items = map { $_, 1 } @_; # the rest are the person's items
		    my @required = qw(preserver sunscreen water_bottle jacket);
		    for my $item (@required) {
		      unless ($whos_items{$item}) { # not found in list?
			aphilipp	Apr 07, 2013	Jun 07, 2013

		print "$who is missing $item.\n";
		      }
		    }
		  }
		
		In order to match the latest version of the subroutine found on page 43 in Chapter 4, change to:
		
		  sub check_required_items {
		    my $who = shift;
		    my $items = shift;
		    my %whose_items = map { $_, 1 } @$items;
		    my @required = qw(preserver sunscreen water_bottle jacket);
		    my @missing = ( );
		    for my $item (@required) {
		      unless ($whose_items{$item}) { # not found in list?
		        print "$who is missing $item.\n";
		        push @missing, $item;
		      }
		    }
		    if (@missing) {
		      print "Adding @missing to @$items for $who.\n";
		      push @$items, @missing;
		    }
		  }
		
		Note from the Author or Editor:
		Change as noted			

Printed	Page 295 2nd paragraph, 2nd sentence	This is the part of the sentence: "add required_items_ok to @EXPORT" Change to: "add check_required_items_ok to @EXPORT"	aphilipp	Apr 07, 2013	Jun 07, 2013

Printed	Page 295
	2nd code example
		This is the code example:
		
		  sub check_required_items {
		    my $who = shift;
		    my $items = shift;
		    my @required = qw(preserver sunscreen water_bottle jacket);
		    my @missing = ( );
		    for my $item (@required) {
		      unless (grep $item eq $_, @$items) { # not found in list?
		        push @missing, $item;
		      }
		    }
		    if (@missing) {
		    ...
		    }
		    else {
		    ...
		    }
		  }
		
		Following the code from the errata submission for page 294, change to:
		
		  sub check_required_items {
		    my $who = shift;
		    my $items = shift;
		    my %whose_items = map { $_, 1 } @$items;
		    my @required = qw(preserver sunscreen water_bottle jacket);
		    my @missing = ( );
		    for my $item (@required) {
		      unless ($whose_items{$item}) { # not found in list?
		        push @missing, $item;
		      }
		    }
		    if (@missing) {
		    ...
		    }
		    else {
		    ...
		    }
		  }
		
		Note from the Author or Editor:
		Change as noted
			aphilipp	Apr 07, 2013	Jun 07, 2013

Printed	Page 296
	2nd code example
		This is the code example:
		
		  sub check_required_items_ok {
		    my $who = shift;
		    my $items = shift;
		    my @required = qw(preserver sunscreen water_bottle jacket);
		    my @missing = ( );
		    for my $item (@required) {
		      unless (grep $item eq $_, @$items) { # not found in list?
		        push @missing, $item;
		      }
		    }
		    if (@missing) {
			aphilipp	Apr 07, 2013	Jun 07, 2013

		$Test->diag("$who needs @missing.\n");
		$Test->ok(0);
		}
		else {
		$Test->ok(1);
		}
		}
		
		Following the code from the errata submission for page 295, change to:
		
		sub check_required_items_ok {
		my $who = shift;
		my $items = shift;
		my %whose_items = map { $_, 1 } @$items;
		my @required = qw(preserver sunscreen water_bottle jacket);
		my @missing = ( );
		for my $item (@required) {
		unless ($whose_items{$item}) { # not found in list?
		push @missing, $item;
		}
		}
		if (@missing) {
		$Test->diag("$who needs @missing.\n");
		$Test->ok(0);
		}
		else {
		$Test->ok(1);
		}
		}
		
		Note from the Author or Editor:
		Change as noted			

Printed	Page 312 2nd paragraph, 3rd line	Change "$_" to "$file".	aphilipp	Sep 11, 2012	Jun 07, 2013
Printed	Page 314 Chap 2 Ex 3	This is the line: print "Country code: " . $isbn->country_code . "\n"; In order to match the exercise description on page 24, change to the following: print "Group code: " . $isbn->group_code . "\n"; **Note from the Author or Editor:** The interface for Business::ISBN changed.	aphilipp	Sep 19, 2012	Jun 07, 2013
Printed	Page 314 Chap 3 Ex 1	grep { -s < 1000 } @ARGV; should be grep { (-s) < 1000 } @ARGV; or grep { -s $_ < 1000 } @ARGV; **Note from the Author or Editor:** There's the small problem of precedence, and the parens fixes this.	aphilipp	Sep 20, 2012	Jun 07, 2013
Printed	Page 316 Figure A-2	In the PeGS diagram, change "$ginger" to "@ginger". **Note from the Author or Editor:** Change as noted	aphilipp	Sep 25, 2012	Jun 07, 2013
Printed	Page 320 Chap 5 Ex 3	The solution should skip comments by including this line at the beginning of the while loop: next if /^#/; **Note from the Author or Editor:** I changed most of those uses and omissions to /\A\s*#/ to skip leading whitespace too.	aphilipp	Oct 03, 2012	Jun 07, 2013
Printed	Page 322 Chap 6 Ex 2	The exercise description on page 90 asks to modify the program from Exercise 1 (which is based on Exercise 2 in Chapter 5). But the solution on page 322 is instead based on Exercise 3 in Chapter 5. **Note from the Author or Editor:** Change the problem in Exercise 6.1 on page 90 to read "The program from Exercise 3 in ..."	aphilipp	Oct 03, 2012	Jun 07, 2013
Printed	Page 327 Chap 9 Ex 2	The following line should be added before the first while loop: my @patterns; **Note from the Author or Editor:** Change as noted	aphilipp	Nov 05, 2012	Jun 07, 2013

<table>
<tr><th></th><th></th><th></th><th></th><th></th><th></th></tr>
<tr><td>Printed</td><td>Page 328
Chap 9 Ex 2</td><td>This is the section of code:

if(/$pattern/) {
print "Match at line $. | $_" if /$pattern/;
next LINE;
}

Change to:

if(/$pattern/) {
print "Match at line $. | $_";
next LINE;
}

<b>Note from the Author or Editor:</b>
Fix as noted</td><td>aphilipp</td><td>Nov 05, 2012</td><td>Jun 07, 2013</td></tr>
<tr><td>Printed</td><td>Page 331
Chap 10 Ex 4</td><td>This is the line:

my $path = shift || 0;

Change to:

my $level = shift || 0;</td><td>aphilipp</td><td>Nov 27, 2012</td><td>Jun 07, 2013</td></tr>
<tr><td>Printed</td><td>Page 331
Chap 10 Ex 4</td><td>This is the line:

dump_data_for_path($_, $data->{$_}, $indent + 1);

Change to:

dump_data_for_path($_, $data->{$_}, $level + 1);

<b>Note from the Author or Editor:</b>
Change $indent to $level</td><td>aphilipp</td><td>Nov 27, 2012</td><td>Jun 07, 2013</td></tr>
<tr><td>Printed</td><td>Page 334
Chap 12 Ex 1</td><td>This is the line:

% module-starter --module=Animal --name=Gilligan --email="gilligan@example.net"

Change "--name" to "--author". Since we are creating a distribution with Build.PL, the "--mb" switch should also be specified.

<b>Note from the Author or Editor:</b>
Change line to

% module-starter --module=Animal --author=Gilligan --email="gilligan@example.net" --mb</td><td>aphilipp</td><td>Dec 20, 2012</td><td>Jun 07, 2013</td></tr>
<tr><td>Printed</td><td>Page 336
Chap 13 Ex 1</td><td>This is the line:

% module-starter --module=Cow,Horse,Sheep --dist=.

Change to:

% module-starter --module=Cow,Horse,Sheep,Mouse --dist=.

<b>Note from the Author or Editor:</b>
Change as noted</td><td>aphilipp</td><td>Dec 23, 2012</td><td>Jun 07, 2013</td></tr>
<tr><td>Printed</td><td>Page 336
Chap 13 Ex 1</td><td>This is the line:

% module-starter --module=Animal,Cow,Horse,Sheep

Change to:

% module-starter --module=Animal,Cow,Horse,Sheep,Mouse

<b>Note from the Author or Editor:</b>
Change as noted</td><td>aphilipp</td><td>Dec 23, 2012</td><td>Jun 07, 2013</td></tr>
<tr><td>Printed</td><td>Page 340
Chap 13 Ex 3</td><td>This is the line:

Person->speak("Hello, world!");

In order to match the exercise description and the sample output, change to:

Person->speak("Hello, World!");</td><td>aphilipp</td><td>Dec 25, 2012</td><td>Jun 07, 2013</td></tr>
<tr><td>Printed</td><td>Page 343
Chap 14 Ex 2</td><td>This solution has the same problems as reported by galeb for page 219. Foofle->speak is returning the value 1 which is being compared against the string "A Foofle goes foof!\n" in the is() test. Also, the code as written gives an undefined subroutine &Foofle::is.

Any corrections to the code should also be applied to the example on page 220.</td><td>aphilipp</td><td>Jan 24, 2013</td><td>Jun 07, 2013</td></tr>
</table>

		Note from the Author or Editor:
		Change answer in Ex 2 to:
		
		use strict;
		use warnings;
		
		use Test::More tests => 6;
		
		BEGIN {
		use_ok('Animal') || print "Bail out!\n";
		}
		
		diag("Testing Animal $Animal::VERSION, Perl $], $^X");
		
		# they have to be defined in Animal.pm
		ok(defined &Animal::speak, 'Animal::speak is defined');
		ok(defined &Animal::sound, 'Animal::sound is defined');
		
		# check that sound() dies
		eval { Animal&#8722;>sound() } or my $at = $@;
		like($at, qr/You must/, 'sound() dies with a message');
		
		# check that speak() dies too
		eval { Animal&#8722;>speak() } or my $at = $@;
		like($at, qr/You must/, 'speak() dies with a message');
		
		{
		package Foofle;
		use parent qw(Animal);
		sub sound { 'foof' }
		}
		
		ok(Foofle&#8722;>speak, 'An Animal subclass does the right thing');			

Printed	Page 348
	Chap 17 Ex 3
		This is the code section:
		
		is(sum(4, -9, 37, 6), 38, '4-9+37+6 is six');
		
		Change to:
		
		is(sum(4, -9, 37, 6), 38, '4-9+37+6 is 38');
			aphilipp	Feb 05, 2013	Jun 07, 2013

Printed	Page 350 Chap 19 Ex 1	The text mentions that the entire distribution can be obtained online. This distribution needs to be added to the Downloads section at http://www.intermediateperl.com/. **Note from the Author or Editor:** I've fixed this: http://www.intermediateperl.com/downloads_page/ There's nothing for the editors to do.	aphilipp	Mar 15, 2013	
Printed	Page 350 Chap 19 Ex 1	The following line should be added to the Animal class: use namespace::autoclean; **Note from the Author or Editor:** Add this line right after "use Moose"; in the first code example, just like you see it in the Horse code in the second example.	aphilipp	Mar 15, 2013	Jun 07, 2013
Printed	Page 355, 356 Chap 21 Ex 3	This is the part from the code examples: is(Acme::GILLIGAN::Utils::sum(@weird_list), 129, 'The weird sum is 128'); Change to: is(Acme::GILLIGAN::Utils::sum(@weird_list), 129, 'The weird sum is 129'); **Note from the Author or Editor:** Change that 128 to a 129	aphilipp	Apr 14, 2013	Jun 07, 2013
Printed	Page 357 Chap 21 Ex 4	This is the line: % module-starter --module=Tie::Cycle Change to: % module-starter --module=Tie::Cycle --dist=. **Note from the Author or Editor:** Change as noted	aphilipp	Apr 14, 2013	Jun 07, 2013
Printed	Page 357 Chap 21 Ex 4	This is the sentence: "Our module will still go into CPAN, but it won�t be index and PAUSE will send us an email explaining what went wrong." Change "index" to "indexed".	aphilipp	Apr 14, 2013	Jun 07, 2013

Printed	Page 358 Chap 21 Ex 6	This is the sentence: "Looking at CPANdeps, we should also now see a matrix report that shows that we have one version that passes and one that doesn't." Shouldn't "CPANdeps" really be "CPAN Testers Matrix"? **Note from the Author or Editor:** Change as noted	aphilipp	Apr 15, 2013	Jun 07, 2013

About O'Reilly

Academic Solutions
Jobs
Contacts
Corporate Information
Press Room
Privacy Policy
Terms of Service
Work with Us
Editorial Independence

Community

Animals
Authors
Webcasts
Community & Featured Users
Forums
Newsletters
O'Reilly Answers
RSS Feeds
Meetups & User Groups
O'Reilly Atlas (beta)

Partner Sites

makezine.com
makerfaire.com
craftzine.com
Ignite Talks
O'Reilly Insights on Forbes.com

Shop O'Reilly

Customer Service
Contact Us
Shipping Information
Ordering & Payment
Affiliate Program
The O'Reilly Guarantee