1.	Record Nr.	UNINA9910154828703321
	Autore	Zandstra Matt
	Titolo	PHP Objects, Patterns, and Practice / / by MATT ZANDSTRA
	Pubbl/distr/stampa	Berkeley, CA : , : Apress : , : Imprint : Apress, , 2016
	ISBN	9781484219966
		1484219961
	Edizione	[5th ed. 2016.]
	Descrizione fisica	1 online resource (XXVII, 576 p. 99 illus., 27 illus. in color.)
	Disciplina	004
	Soggetti	Computer programming
		Programming languages (Electronic computers)
		Web Development
		Programming Languages, Compilers, Interpreters
	Lingua di pubblicazione	Inglese
	Formato	Materiale a stampa
	Livello bibliografico	Monografia
	Nota di bibliografia	Includes bibliographical references at the end of each chapters and index.
	Nota di contenuto	Part I. Objects -- 1. PHP: Design and Management -- 2. PHP and Objects -- 3. Object Basics -- 4. Advanced Features -- 5. Object Tools -- 6. Objects and Design -- Part II. Patterns -- 7. What Are Design Patterns? Why Use Them? -- 8. Some Pattern Principles -- 9. Generating Objects -- 10. Patterns for Flexible Object Programming -- 11. Performing and Representing Tasks -- 12. Enterprise Patterns -- 13. Database Patterns -- Part III. Practice -- 14. Good (and Bad) Practice -- 15. PHP Standards -- 16. PHP Using and Creating Components with Composer -- 17. Version Control with Git -- 18. Testing -- 19. Automated Build with Phing -- 20. Vagrant -- 21. Continuous Integration -- 22. Objects, Patterns, and Practice -- 23. App A: Bibliography -- 24. App B: A Simple Parser.
	Sommario/riassunto	The 5th edition of this popular book has been fully updated for PHP 7, including replacing the PEAR package manager with Composer, and new material on Vagrant and PHP standards. It provides a solid grounding in PHP's support for objects, it builds on this foundation to instill core principles of software design and then covers the tools and practices needed to develop, test and deploy robust code. PHP Objects, Patterns, and Practice begins by covering PHP's object-oriented

features. It introduces key topics including class declaration, inheritance, reflection and much more. The next section is devoted to design patterns. It explains the principles that make patterns powerful. The book covers many of the classic design patterns and includes chapters on enterprise and database patterns. The last segment of the book covers the tools and practices that can help turn great code into a successful project. The section shows how to manage multiple developers and releases with git, how to manage builds and dependencies with Composer. It also explores strategies for automated testing and continuous integration. Taken together, these three elements: object fundamentals, design principles, and best practice, will help the reader develop elegant and rock solid systems.