

1. Record Nr.	UNINA9910954896203321
Autore	Stefanov Stoyan
Titolo	Object-oriented JavaScript : create scalable, reusable high-quality JavaScript applications and libraries // Stoyan Stefanov
Pubbl/distr/stampa	Birmingham, UK, : Packt Publishing, c2008
ISBN	9786611731748 9781281731746 1281731749 9781847194152 184719415X
Edizione	[1st edition]
Descrizione fisica	1 online resource (356 p.)
Disciplina	005.133 006.76
Soggetti	Computer software - Reusability JavaScript (Computer program language) Object-oriented programming (Computer science) Software patterns
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Note generali	Includes index.
Nota di contenuto	Cover; Table of Contents; Preface; Chapter 1: Introduction; A Bit of History; The Winds of Change; The Present; The Future; Object-Oriented Programming; Objects; Classes; Encapsulation; Aggregation; Inheritance; Polymorphism; OOP Summary; Setting up Your Training Environment; Getting the Tools You Need; Using the Firebug Console; Summary; Chapter 2: Primitive Data Types, Arrays, Loops, and Conditions; Variables; Variables are Case Sensitive; Operators; Primitive Data Types; Finding out the Value Type -the typeof Operator; Numbers; Octal and Hexadecimal Numbers; Exponent Literals; Infinity NaNStrings; String Conversions; Special Strings; Booleans; Logical Operators; Operator Precedence; Lazy Evaluation; Comparison; Undefined and null; Primitive Data Types Recap; Arrays; Adding/Updating Array Elements; Deleting Elements; Arrays of arrays; Conditions and Loops; Code Blocks; if Conditions; Checking if a Variable Exists; Alternative if Syntax; Switch; Loops; While Loops; Do-

while loops; For Loops; For-in Loops; Comments; Summary; Exercises;
Chapter 3: Functions; What is a Function?; Calling a Function;
Parameters; Pre-defined Functions; parseInt(); parseFloat(); isNaN();
isFinite()
Encode/Decode URIs; eval(); A Bonus-the alert() Function; Scope of
Variables; Functions are Data; Anonymous Functions; Callback
Functions; Callback Examples; Self-invoking Functions; Inner (Private)
Functions; Functions that Return Functions; Function, Rewrite Thyself!;
Closures; Scope Chain; Lexical Scope; Breaking the Chain with a
Closure; Closure #1; Closure #2; A Definition and Closure #3; Closures
in a Loop; Getter/Setter; Iterator; Summary; Exercises; Chapter 4:
Objects; From Arrays to Objects; Elements, Properties, Methods;
Hashes, Associative Arrays; Accessing Object's Properties
Calling an Object's Methods; Altering Properties/Methods; Using this
Value; Constructor Functions; The Global Object; constructor Property;
instanceof Operator; Functions that Return Objects; Passing Objects;
Comparing Objects; Objects in the Firebug Console; Built-in Objects;
Object; Array; Interesting Array Methods; Function; Properties of the
Function Objects; Methods of the Function Objects; The arguments
Object Revisited; Boolean; Number; String; Interesting Methods of the
String Objects; Math; Date; Methods to Work with Date Objects;
RegExp; Properties of the RegExp Objects
Methods of the RegExp Objects
String Methods that Accept Regular
Expressions as Parameters; search() and match(); replace(); Replace
callbacks; split(); Passing a String When a regexp is Expected; Summary;
Exercises; Chapter 5: Prototype; The prototype Property; Adding
Methods and Properties Using the Prototype; Using the Prototype's
Methods and Properties; Own Properties versus prototype Properties;
Overwriting Prototype's Property with Own Property; Enumerating
Properties; isPrototypeOf(); The Secret __proto__ Link; Augmenting
Built-in Objects; Augmenting Built-in Objects-Discussion
Some Prototype gotchas

Sommario/riassunto

Create scalable, reusable high-quality JavaScript applications and
libraries
