

Table of contents

- Preface
- Part 1 Introduction to object-oriented problem solving and the Smalltalk language**
 - Chapter 1 - Object-oriented problem solving - essential concepts
 - Chapter 2 - Finding objects
 - Chapter 3 - Principles of Smalltalk
- Part 3 Essential classes, user interface components, and application development**
 - Chapter 4 - True and False objects, blocks, selection, iteration
 - Chapter 5 - Numbers
 - Chapter 6 - Design of applications with user interfaces, Action Buttons
 - Chapter 7 - Introduction to collections, Table widgets
 - Chapter 8 - More sequenceable collections, List and menu widgets
 - Chapter 9 - Unordered collections - sets, bags, and dictionaries
 - Chapter 10 - Streams, files, BOSS
- Part 3 Advanced topics**
 - Chapter 11 - Stacks, queues, linked lists, and trees
 - Chapter 12 - More about user interfaces
 - Chapter 13 - Processes and their coordination
- Appendices**
 - Appendix 1 - Check Boxes, Radio Buttons, Input Fields, and their applications
 - Appendix 2 - Dataset, Notebook, Subcanvas, Dialog Window, Menus
 - Appendix 3 - Chess board – a view holder application
 - Appendix 4 - Classes and Metaclasses
 - Appendix 5 - Style recommendations
 - Appendix 6 - Projects
 - Appendix 7 - Smalltalk syntax
 - Appendix 8 - Smalltalk tidbits
 - Appendix 9 - Selected Smalltalk products
- Glossary
- References
- Index

Detailed Table of Contents

	Preface
Part 1	Object-oriented problem solving essential concepts
	Chapter 1 - Object-oriented problem solving - essential concepts
	1.1 Introduction
	1.2 What is object-oriented problem solving?
	1.3 Examples of objects in computer applications
	1.4 How does an object-oriented computer application work?
	1.5 Classes and their instances
	1.6 A first look at Smalltalk classes
	1.7 Object properties
	1.8 Using System Browser to find out about objects
	1.9 Class, subclass, superclass, abstract class, inheritance, class hierarchy
	1.10 Smalltalk's class hierarchy
	1.11 Polymorphism
	Conclusion
	Chapter 2 - Finding objects
	2.1 Examples of object-based solutions
	2.2 Finding objects
	2.3 Example 1 –A Rental Property Management Program
	2.4 Example 2 – The Farm Program
	Conclusion
	Chapter 3 - Principles of Smalltalk
	3.1 Basic rules
	3.2 Maintaining access to objects - variables
	3.3 Writing and executing programs
	3.4 More about variables
	3.5 Smalltalk messages
	3.6 Nesting of expressions
	3.7 Order of evaluation of messages
	3.8 Tracing message evaluation with the Debugger
	3.9 Cascading
	3.10 Global variables, class instance variables, and pool dictionaries
	Conclusion
Part 2	Essential classes, user interface components, and application development
	Chapter 4 - True and False objects, blocks, selection, and iteration
	4.1 Why we need true and false objects
	4.2 Boolean messages for deciding whether to take an action or not
	4.3 The definition of ifTrue:
	4.4 Selecting one of two alternative actions
	4.5 Use ifTrue: and ifFalse: only when necessary
	4.6 Creating a new class and a method
	4.7 Logic operations
	4.8 Exclusive or, equality, equivalence
	4.9 Use of Booleans to repeat a block of statements
	4.10 Other forms of iteration
	Conclusion
	Chapter 5 - Numbers
	5.1 Numbers
	5.2 Operations on numbers
	5.3 Implementation of binary arithmetic messages - double dispatching and primitives
	5.4 Using numbers for iteration - 'repeat n times'

- 5.5 Repeating a block for all numbers between a start and a stop value
- 5.5 Repeating a block with a specified step
- 5.7 Measuring the speed of arithmetic and other operations
- 5.8 Declaring a new class: Currency
- 5.9 Another implementation of Currency
- 5.10 Generalized rectangles
- Conclusion
- Chapter 6 - Design of Applications with graphical user interfaces**
 - 6.1 Example of application development: An application selector
 - 6.2 Implementing the user interface - the window
 - 6.3 Painting widgets and defining their properties
 - 6.4 Defining *Action* and *Aspect* properties
 - 6.5 The remaining *Action* methods
 - 6.6 Text Editor widget
 - 6.7 Value holders, models, and dependents
 - 6.8 Opening an application - hook methods
 - 6.9 MVC – the Model – View – Controller triad
 - 6.10 IDs make widgets accessible at run time - a Tic-Tac-Toe game
 - Conclusion
- Chapter 7 - Introduction to Collections**
 - 7.1 Introduction
 - 7.2 Essential collections
 - 7.3 Properties shared by all collections
 - 7.4 Arrays
 - 7.5 Examples of uses of arrays
 - 7.6 Two-dimensional arrays - tables and matrices
 - 7.7 Implementing an n-dimensional array
 - 7.8 Use of TwoDList in the Table widget
 - Conclusion
- Chapter 8 - More sequenceable collections, List widgets**
 - 8.1 Class OrderedCollection
 - 8.2 Several examples with ordered collections
 - 8.3 Class SortedCollection
 - 8.4 Ordered collections as the basis of dependence
 - 8.5 Tennis – another example of dependency
 - 8.6 The List collection
 - 8.7 String, Text, and Symbol - an introduction
 - 8.8 Text - its nature and use
 - 8.9 List widgets
 - Conclusion
- Chapter 9 - Sets, bags, and dictionaries**
 - 9.1 Sets
 - 9.2 Copying objects
 - 9.3 Bags
 - 9.4 Associations and dictionaries
 - 9.5 Dictionary with multiple values
 - 9.6 Example - a two-way dictionary
 - 9.7 A Finite State Automaton
 - Conclusion
- Chapter 10 - Streams, files, and BOSS**
 - 10.1 Introduction to streams
 - 10.2 Internal streams
 - 10.3 Examples of operations on internal streams
 - 10.4 Example: A text filter
 - 10.5 Example: Circular Buffer

- 10.6 Introduction to files and external streams
- 10.7 Class Filename
- 10.8 Examples of file operations that don't require external streams
- 10.9 External streams
- 10.10 Storing objects with BOSS
- 10.11 Other ways of storing objects
- Conclusion

Part 3 Advanced topics

Chapter 11 - Stacks, queues, linked lists, trees, and graphs

- 11.1 Stack - an access-at-top-only collection
- 11.2 Context Stack and Exceptions
- 11.3 More about exceptions
- 11.4 Queues
- 11.5 Text Filter – a new implementation
- 11.6 Linked Lists
- 11.7 Trees
- 11.8 Use of trees in compilation
- 11.9 Graphs
- Conclusion

Chapter 12 - Developing user interfaces

- 12.1 Principles of user interfaces – display surfaces, graphics contexts, and visual parts
- 12.2 An example of the use of windows – a virtual desktop
- 12.3 Principles of displaying – graphics contexts, geometric objects, and other concepts
- 12.4 Images, pixmaps, masks, and paint
- 12.5 Models, views, and controllers revisited
- 12.6 Creating UI components with the view holder widget
- 12.7 Controllers

Chapter 13 - Processes and their coordination, additional UI topics

- 13.1 A stopwatch and the concept of a Process
- 13.2 Alarm tool
- 13.3 Coordinating mutually dependent processes – train simulation
- 13.4 Making train simulation layout customizable
- Conclusion

Appendices

Appendix 1 - Check Boxes, Radio Buttons, Input Fields, and their applications 32 pages

- A.1.1 Check Boxes and Radio Buttons - an introduction
- A.1.2 Check Boxes
- A.1.3 Radio Buttons
- A.1.4 Input Fields
- A.1.5 A computerized restaurant menu
- A.1.6 Other implementations of restaurant menu
- A.1.7 Validation of user input
- A.1.8 A course evaluation program
- A.1.9 A (very) simple computerized Tax Form
- Conclusion

Appendix 2 - Dataset, Subcanvas, Notebook, Dialog Window, Menus

- A.2.1 Dataset widgets
- A.2.2 Subcanvas
- A.2.3 Diary - Using a subcanvas to reuse a complete application
- A.2.4 The Notebook widget
- A.2.5 Dialog windows
- A.2.6 Menus in general and Pop up Menus in particular
- A.2.7 Menu Bars

Conclusion

Appendix 3 - Chess board – a custom user interface

- A.3.1 Chess - specification
- A.3.2 Preliminary design
- A.3.3 Design refinement
- A.3.4 Implementation

Conclusion

Appendix 4 - Classes, Metaclasses, and Metaprogramming

- A.4.1 Classes and Metaclasses
- A.4.2 What is the complete class hierarchy?
- A.4.3 What are the main properties of metaclasses?
- A.4.4 Class Behavior
- A.4.5 Class ClassDescription
- A.4.6 Class Class
- A.4.7 Is this magic useful?
- A.4.8 Another example of metaprogramming: Enhanced Workspace
- A.4.9 Another example: Wrapping objects to intercept messages

Conclusion

Appendix 5 - Style recommendations

- A.5.1 Introduction to Smalltalk style guidelines
- A.5.2 Naming
- A.5.3 Comments
- A.5.4 Names of common protocols
- A.5.5 Introduction to idioms and patterns
- A.5.6 General patterns
- A.5.7 Methods
- A.5.8 Behaviors
- A.5.9 Variables

Glossary
References
Index