
Contents

Foreword	ix
Preface	xi
Chapter 1. Namespaces: Model and Operations	1
1.1. Reusing, dividing, confining	2
1.1.1. Analysis of some developer needs	2
1.1.2. Meeting developer needs	3
1.2. Namespaces	5
1.2.1. Namespaces definition	6
1.2.2. Extending notions of environment and binding	8
1.3. <i>kit</i> development	12
1.3.1. Namespace of a <i>kit</i>	14
1.3.2. Order of fields introduction	15
1.3.3. Typing <i>kits</i>	15
1.3.4. <i>kit</i> values	18
1.3.5. <i>kit</i> export, confining fields	22
1.3.6. <i>kits</i> import	28
1.3.7. Stages of <i>kit</i> development	32
1.4. Incomplete <i>kits</i>	33
1.4.1. Type and value of an incomplete <i>kit</i>	34
1.4.2. Completion of an incomplete <i>kit</i>	34
1.4.3. Confining an incomplete <i>kit</i>	36
1.5. Parameterized <i>kits</i>	36
1.5.1. <i>kits</i> parameterized by a type	36
1.5.2. <i>kits</i> parameterized by types and values	41
1.5.3. Confinement, parametrization, incomplete <i>kits</i> and export	45
1.6. Functors of <i>kits</i>	46
1.7. <i>kit</i> extension	49

1.7.1. Presentation of extension	49
1.7.2. Confinement and extension	55
1.8. Conclusion	59
Chapter 2. Modules	61
2.1. Modules in Ada	62
2.1.1. Developing modules	62
2.1.2. Export and confinement	65
2.1.3. Nesting modules	66
2.1.4. Importing a module	66
2.1.5. Flattening an import	67
2.1.6. Generic modules	68
2.1.7. Modules and separate compilation	71
2.2. Modules in OCaml	71
2.2.1. Module definition	72
2.2.2. Export and confinement	73
2.2.3. Confinement of type definitions	78
2.2.4. Functors	82
2.3. Modularity, namespaces and <i>W-kit</i>	85
2.3.1. Declaration interfaces	85
2.3.2. <i>W-kits</i>	86
2.3.3. Modularity and header files in C	86
Chapter 3. Class and Object Features	93
3.1. Object-oriented features	93
3.1.1. Objects	94
3.1.2. Classes	95
3.2. <i>kits</i> and object features	104
3.2.1. Modeling classes	104
3.2.2. Modeling objects	107
3.2.3. Inheritance, redefinition and late binding	109
3.2.4. Incomplete <i>C-kits</i> , parameterized <i>C-kits</i>	113
3.2.5. Subclassing, subtyping	115
3.2.6. Type languages, classes and objects	119
Chapter 4. Classes in Selected Languages	123
4.1. Classes in Java	123
4.1.1. General presentation	123
4.1.2. Modules and packages	124
4.1.3. Classes	125
4.1.4. Marks	131
4.1.5. Developing classes	134

4.2. Classes in C++	144
4.2.1. Header files, namespaces, confinement	145
4.2.2. Classes	148
4.2.3. Inheritance and confinement	155
4.2.4. Overloading in C++	165
4.2.5. Parameterized classes	168
4.3. Classes in OCaml	174
4.3.1. Presentation	174
4.3.2. An overview of classes	174
4.3.3. Marks, incomplete classes, parametrization	181
4.3.4. Objects	186
4.3.5. Class signatures: confinement and inheritance	192
4.3.6. Multiple inheritance	195
4.3.7. Other features	197
4.4. Presentation of Python	197
4.4.1. Getting started	198
4.4.2. An overview of classes, modules and types	199
4.4.3. Names and assignment	208
4.4.4. Assignment and typechecking	211
4.4.5. Overloading	213
4.4.6. Modules and packages	217
4.4.7. Confinement	218
4.4.8. Inheritance	219
4.4.9. Incomplete <i>C-kits</i> and abstract classes	221
4.4.10. Other features	222
Appendix: Questions to Guide Learning	225
List of Notations	229
References	231
Index	233