Índice

1 Introducción **1**
- 1.1 Lenguajes de programación . . . 4
  - 1.1.1 Lenguajes compilados e interpretados . . . 5
  - 1.1.2 El proceso de traducción y ejecución en Python . . . 7
- 1.2 Características más destacadas de Python . . . 8
- 1.3 Entornos de desarrollo integrados . . . 12

2 Tomando contacto **15**
- 2.1 Modo intérprete interactivo . . . 16
- 2.2 Uso del editor: creación de *scripts* . . . 20
- 2.3 Uso de comentarios . . . 22
- 2.4 Datos y tipos de dato: objetos y clases . . . 24
- 2.5 Nombres y objetos . . . 25
  - 2.5.1 La sentencia de asignación . . . 26
  - 2.5.2 Creación de objetos y referencias . . . 27
  - 2.5.3 Asignación múltiple y encadenamiento de asignaciones . . . 28
  - 2.5.4 Persistencia de los objetos . . . 30
  - 2.5.5 Reglas para la creación de nombres . . . 32
- 2.6 Inspección de valores en consola . . . 33
- 2.7 Errores . . . 35

3 Las clases numéricas y la clase `bool` **39**
- 3.1 Números: las clases `int`, `float` y `complex` . . . 40
  - 3.1.1 Números enteros y reales . . . 40
  - 3.1.2 Operadores aritméticos . . . 43
  - 3.1.3 Asignación aumentada . . . 46
  - 3.1.4 Errores en la representación interna de números reales . . . 47
  - 3.1.5 Funciones numéricas nativas . . . 47
  - 3.1.6 Números complejos . . . 51
- 3.2 Valores lógicos: la clase `bool` . . . 53
  - 3.2.1 Operadores lógicos . . . 53
  - 3.2.2 Operadores relacionales . . . 54
- 3.3 Construcción y evaluación de expresiones . . . 56
  - 3.3.1 Precedencia de operadores . . . 56
  - 3.3.2 Asignación y control del tipo de los datos . . . 57

4 Secuencias **61**
4.1 Clases secuencia en Python . . . 63
4.1.1 Cadenas de caracteres: la clase `str` . . . 63
4.1.2 Tuplas: la clase `tuple` . . . 67
4.1.3 Listas: la clase `list` . . . 72
4.1.4 Rangos: la clase `range` . . . 75
4.2 Operaciones comunes a las clases secuenciales . . . 77
4.2.1 Longitud de una secuencia: la función `len()` . . . 78
4.2.2 Comprobación de pertenencia: operadores `in` y `not in` . . . 78
4.2.3 Comparación lexicográfica: operadores `<`, `>`, `<=`, `>=`, `==` y `!=` . . . 80
4.2.4 Concatenación y replicación: operadores `+` y `*` . . . 82
4.2.5 Indexación y secciones: el operador `[]` . . . 85
4.2.6 Búsqueda en una secuencia: los métodos `.index()` y `.count()` . 90
4.2.7 Ordenación de una secuencia: la función `sorted()` . . . 92
4.2.8 Máximo y mínimo de una secuencia: funciones `max()` y `min()` . . 93
4.3 Anidamiento en listas y tuplas . . . 94
4.4 Secuencias inmutables versus secuencias mutables . . . 95
4.4.1 Las listas y las tuplas contienen referencias . . . 96
4.5 Particularidades de las cadenas de caracteres . . . 98
4.5.1 Concatenación de literales *string* . . . 98
4.5.2 Secuencias de escape . . . 99
4.5.3 Introducción de texto por parte del usuario: la función `input()` . 100
4.5.4 Literales *string* con formato . . . 102
4.5.5 Manipulación de *strings* mediante métodos de la clase `str` . . . 107
4.6 Manipulación de listas . . . 112

5 Diccionarios y conjuntos **125**
5.1 Diccionarios . . . 126
5.1.1 Creación de diccionarios . . . 127
5.1.2 Consulta de diccionarios . . . 130
5.1.3 Modificación de diccionarios . . . 131
5.1.4 Anidamiento de diccionarios . . . 132
5.2 Conjuntos . . . 134
5.2.1 Creación de conjuntos . . . 134
5.2.2 Consulta de pertenencia . . . 136
5.2.3 Operaciones sobre conjuntos . . . 136
5.2.4 Conjuntos inmutables . . . 141
5.3 Otros datos estructurados en Python . . . 143

6 Estructuras de control **145**
6.1 Estructura de control condicional . . . 147
6.1.1 La sentencia `if` . . . 149
6.1.2 Importancia del sangrado y otras cuestiones sintácticas . . . 154
6.2 Bucles controlados por condición: la sentencia `while` . . . 157
6.2.1 Anidamiento de bucles `while` . . . 159
6.3 Condiciones: qué es verdadero y qué es falso . . . 161
6.4 Bucles iteradores: la sentencia `for` . . . 163

6.4.1 Objetos iterables . . . 163
6.4.2 La sentencia `for` . . . 164
6.4.3 Iteración sobre objetos colección . . . 165
6.4.4 Bucles `for` controlados por contador . . . 170
6.4.5 Anidamiento de bucles `for` . . . 173
6.4.6 Modificación iterativa de objetos colección mutables . . . 176
6.5 Otras construcciones relacionadas . . . 180
6.5.1 Expresiones condicionales . . . 180
6.5.2 Comprensión de listas, diccionarios y conjuntos . . . 182
6.5.3 Expresiones generadoras . . . 189
6.6 Alteración del comportamiento de un bucle . . . 192
6.7 Correspondencia de patrones: la sentencia `match` . . . 199
6.8 Gestión de excepciones: la sentencia `try` . . . 206

7 Modularidad 219
7.1 Invocación: argumentos y devolución de valores . . . 221
7.1.1 Devolución de valores . . . 223
7.1.2 Tipos de argumento en las llamadas a función . . . 224
7.2 Definición de funciones . . . 229
7.2.1 Dónde definir nuestras funciones . . . 231
7.2.2 Reglas de ámbito: espacios de nombres . . . 232
7.2.3 Devolución de objetos y finalización de la ejecución con `return` . 235
7.2.4 Documentación de funciones . . . 237
7.2.5 Generación de excepciones: la sentencia `raise` . . . 239
7.2.6 Correspondencia entre argumentos y parámetros . . . 242
7.2.7 Especificación de parámetros . . . 244
7.3 Efectos de la mutabilidad de los argumentos . . . 256
7.3.1 Paso de objetos inmutables como argumentos . . . 256
7.3.2 Paso de objetos mutables como argumentos . . . 257
7.4 Programación estructurada y funciones . . . 259
7.5 Estructura modular del código . . . 261
7.5.1 Creación e importación de módulos: la sentencia `import` . . . 262
7.5.2 Un paso más en la organización modular: creación de paquetes . 266

8 Archivos 271
8.1 Gestión de rutas: el módulo `pathlib` . . . 272
8.1.1 Creación de objetos `Path` . . . 273
8.1.2 Rutas relativas y absolutas . . . 275
8.1.3 Acceso a los componentes de una ruta . . . 276
8.1.4 Comprobación de la existencia de archivos y carpetas . . . 277
8.2 Gestión de archivos . . . 278
8.2.1 Apertura de archivos . . . 278
8.2.2 Cierre de archivos . . . 283
8.2.3 Gestión de errores al operar con archivos . . . 284
8.3 Lectura y escritura en modo texto . . . 286
8.4 Gestión de archivos CSV . . . 301
8.4.1 El módulo CSV . . . 303

8.4.2 Lectura y escritura de archivos CSV . . . 304
8.4.3 Gestión de archivos CSV con cabecera . . . 306
8.5 Lectura y escritura en modo binario . . . 310

9 Orientación a objetos 317
9.1 Clases y objetos . . . 319
9.2 Definición de clases e instanciación de objetos . . . 320
9.2.1 Inicialización del estado: atributos de instancia . . . 321
9.2.2 Definición del comportamiento: métodos de instancia . . . 324
9.3 Objetos clase. Atributos de clase . . . 327
9.4 Atributos privados y alteración de nombres . . . 334
9.5 Atributos especiales . . . 335
9.6 Atributos gestionados: la función `property()` . . . 337
9.6.1 Uso de la función `property()` . . . 339
9.6.2 Uso del decorador `@property` . . . 342
9.7 Métodos de clase y métodos estáticos . . . 349
9.8 Métodos especiales . . . 352
9.9 Herencia . . . 369
9.9.1 Herencia simple . . . 370
9.9.2 La superclase `object` . . . 379
9.9.3 Clases base abstractas . . . 381
9.9.4 Herencia múltiple . . . 383
9.9.5 Composición, agregación y asociación . . . 384
9.10 Polimorfismo . . . 385