

ÍNDICE

CAPÍTULO 1

INTRODUCCIÓN AL LENGUAJE PYTHON	1
1 Introducción.....	2
1.1 Evolución de las versiones de Python.....	2
1.2 Características principales de los lenguajes de programación ...	4
1.2.1 Generaciones y niveles de abstracción de los lenguajes de programación	4
1.2.2 Paradigmas	5
1.2.3 Clasificación de lenguajes según su tipado.....	7
1.2.4 Características de Python.....	7
1.3 Ámbitos de uso de Python.....	10
1.3.1 Programación a nivel de sistema operativo	10
1.3.2 Aplicaciones con interfaz de usuario.....	10
1.3.3 Aplicaciones web e interacción con servicios web.....	11
1.3.4 Interacción con servicios de Internet.....	12
1.3.5 Gestión de contenido.....	13
1.3.6 Aplicaciones científicas y manejo de datos.....	13
1.3.7 Inteligencia artificial y Python	15
1.4 Python Enhancement Proposals (PEP)	16
1.4.1 Proceso de creación de un PEP	17
1.5 PEP-8: Guía de estilos	18
1.5.1 Indentación	19
1.5.2 Longitud de líneas.....	19
1.5.3 Espacios, saltos de línea y líneas en blanco	20
1.5.4 Otros consejos generales	20
1.5.5 Comentarios y documentación de código.....	21
1.5.6 Convención de nombres.....	22
1.5.7 Herramientas para cumplir con la PEP-8	24
1.6 PEP-20: Zen de Python	24

1.7	Compañías que usan productos creados en Python.....	26
1.8	Posición de Python entre los lenguajes de programación	28
1.9	Python 2 vs Python 3	30
1.9.1	str, bytes y Unicode.....	30
1.9.2	Comparaciones de tipos no ordenables.....	31
1.9.3	Operaciones numéricas diferentes.....	31
1.9.4	Iteradores por defecto.....	32
1.9.5	Función print.....	32
1.9.6	Migrar de Python 2 a Python 3.....	33
1.10	Instalación de Python en diferentes sistemas operativos	35
1.10.1	Instalación en Linux.....	35
1.10.2	Instalación en Windows	37
1.10.3	Instalación en Mac OS X.....	38
1.11	Distribuciones de Python.....	39
1.11.1	Anaconda.....	40
1.11.2	WinPython	41
1.11.3	Enthought Canopy	42
1.11.4	ActivePython.....	43
1.12	Instalación de librerías y módulos en Python.....	43
1.13	Manejo de entornos virtuales.....	46
1.14	Intérpretes interactivos (REPL).....	47
1.14.1	Python.....	48
1.14.2	IPython	48
1.14.3	bpython.....	49
1.14.4	ptpython	49
1.14.5	Intérpretes online.....	50
1.15	El intérprete de Python	50
1.15.1	Estructura del intérprete de Python	51
1.16	Implementaciones de Python	52
1.16.1	CPython	53
1.16.2	Jython.....	53
1.16.3	PyPy.....	53
1.16.4	IronPython	54
1.17	Desarrollar programas en Python (IDE).....	54

1.17.1	Entornos de desarrollo open source.....	55
1.17.2	Entornos de desarrollo de código cerrado	64
1.17.3	Comentario general sobre IDE.....	67
1.18	Primeros programas en Python.....	67
1.18.1	Ejecutar programas en el REPL de Python	67
1.18.2	Usar Jupyter Notebook.....	68
1.18.3	Primeros programas ejecutados desde ficheros.....	71

CAPÍTULO 2

	VARIABLES Y TIPOS DE DATOS	75
1	Introducción a los tipos de datos	75
2	Literales, variables y datos en Python	77
2.1	Literales.....	78
2.2	Variables e identificadores	79
2.3	Gestión de memoria en Python	83
2.4	Mutabilidad de variables.....	85
3	Tipos booleanos.....	86
3.1	Operaciones con booleanos	87
3.2	Cortocircuito lógico	87
4	Comparaciones	88
5	Tipos numéricos	89
5.1	Operaciones numéricas básicas	89
5.2	Enteros	90
5.3	Operaciones a nivel de bits con enteros	95
5.4	Números de coma flotante	95
5.5	Números complejos	99
5.6	Utilizar distintos tipos numéricos	101
6	Secuencias	101
6.1	Listas.....	102
6.2	Tuplas.....	107
6.3	Rangos	109
6.4	Selección de subsecuencias basadas en índices (slices)	113
6.5	Operaciones predefinidas para secuencias	114
7	Secuencias de caracteres	115

7.1	Construir cadenas de caracteres	120
7.2	Convertir caracteres a números y viceversa	121
7.3	Operaciones de búsqueda de caracteres y conteo	122
7.4	Operaciones relacionadas con el tamaño de letra	123
7.5	Operaciones de identificación de cadenas	125
7.6	Operaciones relacionadas con la codificación.....	131
7.7	Traducciones, reemplazos y mapeos.....	133
7.8	Funciones de manipulación de cadenas: limpiado, división y unión de cadenas	135
7.9	Funciones de justificado y alineación de cadenas de caracteres	139
7.10	Formatear cadenas de caracteres	141
7.11	Diferentes subtipos de cadenas.....	144
7.12	Cadenas f-string en profundidad	146
7.13	Introducción al minilenguaje de formateado de strings.....	149
7.14	Función para imprimir caracteres (<code>print</code>)	153
8	Secuencias binarias	155
8.1	Qué son los datos binarios.....	155
8.2	Tipos <code>bytes</code> y <code>bytearray</code>	156
8.3	Operaciones con <code>bytes</code> y <code>bytearray</code>	159
8.4	Cadenas de literales de <code>bytes</code> (<code>byte literals</code>)	161
9	Conjuntos (<code>set</code> y <code>frozenset</code>).....	163
9.1	Funciones de actualización de conjuntos	164
9.2	Funciones para operar conjuntos.....	166
9.3	Operaciones condicionales para conjuntos	167
10	Mapas (diccionarios).....	168
10.1	Explorar valores de diccionarios	172
10.2	Actualizar valores en diccionarios	174
10.3	Objetos de tipo vista en diccionarios (<code>view objects</code>)	177
11	Iterables e iteradores.....	179
11.1	Operadores para trabajar con iterables.....	182
12	Expresiones generadoras	188
12.1	Inicializar tipos por comprensión	190
12.2	Inicializar objetos con expresiones generadoras.....	194

CAPÍTULO 3

FUNDAMENTOS DEL LENGUAJE	195
1 Asignaciones simples y múltiples	195
1.1 Asignaciones simples	196
1.2 Asignaciones de múltiples variables.....	196
1.3 Asignaciones múltiples	198
1.4 Asignaciones slicing con *	199
2 Control de flujo de ejecución.....	199
2.1 Control de flujo condicional con las sentencias if, elif y else.....	200
2.2 Implementaciones de switch case en Python.....	201
2.3 Sentencia if ternaria	202
3 Flujo de ejecución con bucles	202
3.1 Analizando bucles while.....	202
3.2 Usar bucles con sentencia for	204
3.3 Control de flujo dentro de bucles: break y continue.....	207
4 Operador walrus para asignar	208
5 Funciones en Python.....	209
5.1 Ejemplo de modularización de código.....	212
5.2 Parámetros y argumentos en funciones.....	213
5.3 Uso de args y kwargs.....	217
5.4 Anotaciones y tipado en las funciones.....	219
5.5 Funciones recursivas	221
5.6 Funciones anónimas: expresiones lambda	226
5.7 Funciones de orden superior	229
5.8 Funciones dentro de funciones.....	230
5.9 Espacio de nombres y contextos	232
5.10 Memoización	235
5.11 Decoradores	236
5.12 Documentación de funciones.....	240
5.13 Funciones generadoras.....	241
5.14 Corrutinas	242
5.15 Funciones asíncronas	248
6 Excepciones.....	251
6.1 Controlar el flujo de ejecución con excepciones	252

6.2	Utilizar las trazas de error	253
6.3	Excepciones conocidas	255
6.4	Elevar excepciones de forma manual	267
6.5	Definición de excepciones propias	268

CAPÍTULO 4

PROGRAMACIÓN ORIENTADA A OBJETOS	271	
1	Definición de clase	272
2	Atributos	272
2.1	Inicializar clases	273
2.2	Operar con los atributos	274
2.3	Atributos de clases	275
3	Nombres y privacidad en clases	278
4	Construcción de clases personalizadas	280
5	Propiedades en clases	281
6	Métodos	287
6.1	Métodos de instancia	287
6.2	Métodos de clase	288
6.3	Métodos estáticos	290
7	Métodos mágicos	291
7.1	Métodos para usar operaciones matemáticas	295
7.2	Emular contenedores	300
7.3	Personalizar el acceso a los atributos	305
7.4	Información sobre funciones definidas por el usuario	308
8	Controlar el espacio de atributos con <code>slots</code>	311
9	Duck typing o polimorfismo	312
10	<code>namedtuple</code>	314
11	<code>dataclasses</code>	317
12	Herencia	321
12.1	Herencia simple y el uso de <code>super</code>	324
12.2	Herencia múltiple	327
12.3	Clases Mixin	333

13	Metaclasses y <code>type</code>	336
13.1	Creación de metaclasses propias	339
14	Entonces, ¿cuándo se deben usar clases?.....	341

CAPÍTULO 5

ESTRUCTURA DE CÓDIGO EN PYTHON	343
1 Diferentes componentes de un módulo.....	344
2 Estructura básica de los paquetes.....	345
2.1 Importación de código Python.....	346
2.2 Potenciales problemas de usar <code>import *</code>	349
2.3 Evitar problemas al importar con importaciones cíclicas.....	351
2.4 Importar contenido en <code>__init__.py</code>	355
3 Repositorios de paquetes.....	357
3.1 Estructura de un paquete de Python.....	357
4 Paquetes disponibles en la librería estándar.....	358

CAPÍTULO 6

PERSISTENCIA DE DATOS EN FICHEROS	379
1 Ficheros de texto plano.....	379
1.1 Trabajar con ficheros sin estructura	380
1.2 Trabajar con ficheros de anchura definida.....	388
1.3 Trabajar con ficheros en formato CSV y TSV.....	392
1.4 XML.....	400
1.5 HTML.....	405
1.6 JSON.....	415
1.7 YAML	422
1.8 Librerías con todo incluido - <code>tablib</code> y <code>pandas</code>	424
2 Ficheros binarios	427
2.1 Serialización de objetos Python - <code>pickle</code>	428
2.2 Persistiendo diccionarios - <code>shelve</code>	431
3 Compresión y archivación de ficheros	433
3.1 Comprimir ficheros en archivos ZIP - <code>zipfile</code>	434
3.2 Archivación y compresión de ficheros - <code>tarfile</code>	438

CAPÍTULO 7

- PERSISTENCIA EN BASES DE DATOS 443**
- 1 Interfaz para trabajar con bases de datos DB-API 444
 - 1.1 Funciones básicas para todos los conectores 444
 - 1.2 Capa de abstracción de bases de datos en Python
 - pydal 446
 - 1.3 Librería para consultas SQL en crudo - records..... 449
- 2 Bases de datos relacionales o SQL 451
 - 2.1 Conceptos básicos de las bases de datos relacionales 453
 - 2.2 Sentencias básicas en SQL 455
 - 2.3 SQLite y Python..... 458
 - 2.4 Diferentes bases de datos profesionales 464
- 3 Mapeo de objetos relacionales - ORM..... 470
 - 3.1 SQLAlchemy ORM 471
 - 3.2 Peewee ORM..... 473
 - 3.3 Pony ORM 475
 - 3.4 Django models..... 475
 - 3.5 ORM asíncrono GINO 477
- 4 Bases de datos NoSQL..... 477
 - 4.1 Bases de datos clave-valor..... 478
 - 4.2 Bases de datos orientadas a documentos..... 478
 - 4.3 Bases de datos en tiempo real 480
 - 4.4 Bases de datos para series temporales 480
 - 4.5 Bases de datos orientadas a grafos de datos 481
 - 4.6 Bases de datos columnares 483
 - 4.7 Bases de datos para búsquedas de texto - Full-text databases..... 484
- 5 Sistemas de clave-valor y cachés en Python - dbm, Memcached y Redis..... 485
 - 5.1 Usar dbm en Python..... 486
 - 5.2 Memcached en Python..... 488
 - 5.3 Usar Redis como base de datos y caché en Python 490
- 6 Bases de datos en la nube..... 494
 - 6.1 Servicios de bases de datos en la nube..... 494

CAPÍTULO 8

PARALELISMO Y CONCURRENCIA	499
1 Procesos en Python	503
1.1 Lanzar procesos externos - subprocess	503
1.2 Múltiples procesos en Python - multiprocessing	507
2 Hilos en Python	515
2.1 GIL en CPython	516
2.2 Hilos en Python - threading	517
3 Ejecuciones de hilos y de procesos - concurrent futures	524
4 Asincronía de entrada/salida - Asynchronous I/O	526
4.1 AsyncIO en Python	527

CAPÍTULO 9

INTERFACES DE USUARIO	537
1 Interfaz con consola de comandos en Python - CUI	538
1.1 input y print - E/S estándar	538
1.2 CLI estándar - argparse	539
1.3 Interfaces de usuario basadas en texto	542
2 Interfaces gráficas de usuario - GUI	545
3 Tkinter a fondo	547
3.1 Componentes principales de una aplicación tkinter	548
3.2 Componentes disponibles	549
3.3 Disposición y propiedades de elementos en tkinter	550
3.4 Manejo de eventos en tkinter	551
3.5 Organización de la aplicación	552
4 Ejemplos de aplicaciones creadas con tkinter	556
4.1 Calculadora de porcentajes creada con tkinter	556
4.2 Analizador de ficheros de texto creado con tkinter	561

CAPÍTULO 10

SERVICIOS DE RED Y APLICACIONES WEB	569
1 Protocolos de Internet y Python	569
1.1 Transferencia de hipertexto - HTTP y HTTPS	571
1.2 Transmisión de ficheros - FTP	574

1.3	Conexiones entre servidores - Telnet	578
1.4	Conexiones seguras entre servidores - SSH y SFTP	581
1.5	Correo electrónico - SMTP	585
2	Desarrollo web	591
2.1	Participantes en la web	591
2.2	Trabajando con protocolo HTTP (request-response)	593
2.3	Estructura de las aplicaciones web en Python	597
2.4	Servidor simple utilizando <code>http.server</code>	600
2.5	Frameworks web en Python	601
3	Desarrollando un blog con Django	605
3.1	Planteamiento de la aplicación a realizar	605
3.2	Primeros pasos con Django	607
3.3	Primera aplicación con Django	609
3.4	Modelos con Django	611
3.5	Panel de administración de Django	616
3.6	Añadiendo contenido por defecto	620
3.7	Desarrollando la lógica de la aplicación	623
3.8	Renderizado basado en plantillas	624

ANEXO A

TESTEO Y COBERTURA DE APLICACIONES EN PYTHON	627
---	------------