

# 目次

訳者まえがき	v
まえがき	vii
<b>1章 テキスト</b>	<b>1</b>
1.0 イントロダクション	1
1.1 文字列を1文字ずつ処理する	7
1.2 キャラクタとその文字コード値を相互変換	8
1.3 オブジェクトが文字列のようなものかテストする	9
1.4 文字列の整形	11
1.5 文字列両端のスペースを刈り込む	11
1.6 文字列の連結	12
1.7 文字列を単語ごとに反転する	15
1.8 文字列にキャラクタセットの文字が含まれるか調べる	16
1.9 文字列クラスのtranslateメソッドを簡便に使う	19
1.10 文字列をキャラクタセットでフィルタリング	21
1.11 文字列がテキストかバイナリ調べる	24
1.12 大文字小文字	26
1.13 文字列の一部にアクセスする	27
1.14 インデントの変更	31
1.15 タブとスペースの変換	32
1.16 文字列中の変数に手を加える	35
1.17 文字列中の変数に手を加える (Python2.4以降)	36
1.18 一度に複数のパターンの置換をする	38
1.19 文字列末尾が各種パターンと一致するか調べる	41
1.20 Unicodeにより多言語テキストを扱う	43
1.21 Unicodeと通常の文字列を相互変換	45
1.22 標準出力にUnicodeキャラクタを出力	47

1.23	XMLやHTML向けにUnicodeデータをエンコード	48
1.24	一部の文字列のみ大文字小文字無関係にする	51
1.25	Unixターミナル上でHTMLをテキストに変換	55

## 2章 ファイル ..... 57

2.0	イントロダクション	57
2.1	ファイルを読む	62
2.2	ファイルに書き込む	65
2.3	ファイル中のテキストの検索置換	67
2.4	ファイル中の特定の行を読み込む	68
2.5	ファイルの行数を数える	69
2.6	ファイル中の全てのワードを処理する	72
2.7	入出力をランダムアクセスで	74
2.8	ランダムアクセスファイルの更新	75
2.9	zipファイルのデータを読む	77
2.10	文字列の中のzipファイルの扱い	79
2.11	ファイルツリーをtarファイルにして圧縮	80
2.12	バイナリデータをWindowsの標準出力に送る	82
2.13	C++的なiostreamシンタックスを使う	83
2.14	不可逆ストリーム入力のファイルを巻き戻す	85
2.15	「ファイル的」オブジェクトを普通のファイルオブジェクトに	88
2.16	ディレクトリツリーの探索	89
2.17	ディレクトリツリーに散在するファイルの拡張子を変更	90
2.18	サーチパスにあるファイルを見つける	92
2.19	サーチパスとパターンマッチでファイルを見つける	93
2.20	Pythonサーチパスにあるファイルを見つける	94
2.21	Pythonサーチパスの動的変更	95
2.22	相対パスを導き出す	97
2.23	バッファされない文字をプラットフォームに依存せずに読む	99
2.24	Mac OS XでPDFのページ数を数える	100
2.25	Windowsでファイル属性を変更する	101
2.26	OpenOffice.org文書からテキストを抽出	102
2.27	Microsoft Word文書からテキストを抽出	103
2.28	プラットフォームに依存せずAPIでファイルロック	105
2.29	ファイル名にバージョンを	107
2.30	CRC-64の計算	109

## 3章 時間とお金 ..... 113

3.0	イントロダクション	113
3.1	昨日と明日の計算	119
3.2	先週の金曜日	121
3.3	2つの日付の間の期間は	123
3.4	合計演奏時間	124
3.5	2つの日付の間のウィークデーの数	125
3.6	休日の自動計算	127
3.7	ファジーな日付解釈	131
3.8	夏時間が使用されているか調べる	132
3.9	タイムゾーン間の変換	133
3.10	コマンドの反復実行	135
3.11	コマンドの予定実行	136
3.12	Decimal(十進)による計算	138
3.13	Decimalを通貨として整形する	140
3.14	Pythonを簡単な加算器に	144
3.15	クレジットカードのチェックサム	147
3.16	為替レートの監視	148

## 4章 ショートカット ..... 151

4.0	イントロダクション	151
4.1	オブジェクトのコピー	153
4.2	リスト内包でリストを作る	156
4.3	指定した要素がリスト中に存在すれば返す	158
4.4	シーケンスのアイテムとそのインデックスをループ処理	159
4.5	参照を共有せずにリストのリストを作る	160
4.6	入れ子になったシーケンスの平滑化	162
4.7	列から成るリストのデータを行単位で削除/並べ替え	164
4.8	2次元配列の転置	166
4.9	ディクショナリから値を得る	168
4.10	ディクショナリにエントリを追加	169
4.11	引用符におぼれずディクショナリ構築	171
4.12	キーと値が交互に入ったリストからdictを構築	172
4.13	ディクショナリの部分集合	174
4.14	ディクショナリの反転	176
4.15	ディクショナリの各キーに複数の値を割当てる	177
4.16	メソッドや関数の発行にディクショナリを使う	179
4.17	複数のディクショナリ集合から、和(∪)や交差(∩)を探索	180
4.18	名前のついたアイテム群を集めておく	183

4.19	単一の文で代入と検査を同時に行う	185
4.20	Pythonでprintf	187
4.21	アイテムごとに確率を定めたランダム抽出	188
4.22	式中での例外処理	189
4.23	指定のモジュール内で名前が定義されていることを保証する	191

## 5章 サーチとソート ..... 193

5.0	イントロダクション	193
5.1	ディクショナリのソート	198
5.2	文字列リストを大文字小文字無関係にソート	199
5.3	オブジェクトのリストをオブジェクトの属性でソート	200
5.4	キーやインデックスを、その値によりソート	203
5.5	含まれる数値で文字列をソート	206
5.6	リストのアイテム全てをランダムな順序で処理	207
5.7	アイテムが加えられてもソート済みに保つ	209
5.8	シーケンスのアイテムを小さい方から取る	211
5.9	ソート済みシーケンスからアイテムを探索	213
5.10	n番目に小さい要素は	214
5.11	クイックソートを3行で示す	217
5.12	シーケンスにアイテムの存在チェックをかける	220
5.13	サブシーケンスを見つける	223
5.14	ディクショナリにレーティング機能を	224
5.15	名前をソートしてイニシャルで分ける	229

## 6章 オブジェクト指向プログラミング ..... 231

6.0	イントロダクション	231
6.1	温度スケールの変換	237
6.2	定数を定義する	240
6.3	属性の設定を制限	242
6.4	ディレクトリ探索のチェーン化	244
6.5	継承ではできないことをする(自動委譲)	246
6.6	proxyで特殊メソッドを委譲	250
6.7	名前付きアイテムのタプル	253
6.8	プロパティにアクセスするコードをキレイに	255
6.9	オブジェクトのコピーを素早く	256
6.10	ガベージコレクションを邪魔することなく結合メソッドへのリファレンスを保持	259
6.11	リングバッファを実装する	262
6.12	インスタンスに変化があったかチェックする	265

6.13	オブジェクトが必要な属性を備えているかチェックする	269
6.14	Stateデザインパターンを実装する	272
6.15	Singletonデザインパターンを実装する	274
6.16	Singletonデザインパターンの使用を避けてBorgの方法を利用する	276
6.17	Null Objectデザインパターンを実装する	281
6.18	__init__の引数でインスタンス変数を自動設定	284
6.19	スーパークラスの__init__が存在すれば実行する	286
6.20	スーパークラスのメソッドの共同呼び出しを簡潔かつ安全に	289

## 7章 永続性、データベース ..... 293

7.0	イントロダクション	293
7.1	marshalモジュールを用いてデータをシリアライズする	296
7.2	pickle/cPickleモジュールを利用してシリアライズする	298
7.3	シリアライズしたデータを圧縮する	300
7.4	クラス、インスタンスにcPickleを使用する	301
7.5	結合メソッドをPickle化できる状態で保存する	304
7.6	コードオブジェクトをPickle化する	306
7.7	shelveを使ったオブジェクトに変更を加える	309
7.8	Berkeley DBデータベースを使う	311
7.9	MySQLデータベースへのアクセス	314
7.10	BLOBをMySQLに保存する	315
7.11	BLOBをPostgreSQLに保存する	317
7.12	BLOBをSQLiteに保存する	318
7.13	フィールド名と列番号の対応表のディクショナリを生成する	320
7.14	dtupleでクエリ結果にフレキシブルにアクセスする	322
7.15	データベースカーソルの内容をきれいに出力したい	324
7.16	DB APIモジュールに渡すパラメータのスタイルを統一する	327
7.17	ADOを経由してMicrosoft Jetを使用する	329
7.18	JythonのサープレットからJDBCを使ってデータベースにアクセスしたい	331
7.19	JythonでExcelのデータを扱うためにODBCを利用する	334

## 8章 デバッグとテスト ..... 337

8.0	イントロダクション	337
8.1	ループや条件式の実行を止める	338
8.2	Linux上でもメモリ使用量を測定する	339
8.3	ガベージコレクションの過程をデバッグ	341
8.4	例外のトラップと記録	342
8.5	デバッグモードで式やコメントを追う	344

8.6	トレースバックからより多くの情報を	347
8.7	トラップできない例外で自動的にデバグを起動	350
8.8	最も簡単なユニットテスト	351
8.9	ユニットテストを自動的に	353
8.10	doctestをunittestで使う(Python2.4)	354
8.11	ユニットテストで範囲値をチェック	357
<b>9章</b>	<b>プロセスとスレッド、その同期</b>	<b>359</b>
9.0	イントロダクション	359
9.1	オブジェクト中の全メソッドを同期	362
9.2	スレッドの終了	365
9.3	Queue.Queueを順位付きキューとして使う	367
9.4	スレッドプール	369
9.5	引数のセットをいくつも使って関数を並列実行	373
9.6	簡単なメッセージパッシングによるスレッド協調	375
9.7	スレッドごとの情報を保存	378
9.8	スレッドを使わずに協調的マルチタスクを	381
9.9	Windowsで、同じスクリプトが既に走っているか確かめる	383
9.10	MsgWaitForMultipleObjectsでWindowsメッセージの処理	385
9.11	popenで外部プロセスを駆動	388
9.12	Unixシェルコマンドの標準出力、エラー出力をキャプチャ	389
9.13	デーモンのフォーク(Unix)	392
<b>10章</b>	<b>システム管理</b>	<b>395</b>
10.0	イントロダクション	395
10.1	ランダムなパスワードの生成	396
10.2	簡単に覚えられて、ある意味ランダムなパスワードの生成	397
10.3	POPサーバを使ったユーザ認証	400
10.4	IPアドレスごとのヒット数集計(Apache)	402
10.5	クライアントキャッシュのヒット率計算(Apache)	404
10.6	スクリプトからエディタを起動	405
10.7	ファイルのバックアップ	407
10.8	Mailboxファイルを選択コピー	409
10.9	Mailboxからアドレスのホワイトリストを作成	410
10.10	重複メールのブロック	412
10.11	Windowsのサウンドシステムをチェック	414
10.12	Windows DLLの登録・削除	415
10.13	Windowsのログイン時の自動実行タスクをチェック、修正	416

10.14	「共有」の作成(Windows)	418
10.15	実行中のInternet Explorerにつなげる	418
10.16	Microsoft Outlookの「連絡先」を読む	420
10.17	詳しいシステム情報を得る(Mac OS X)	422

## 11章 ユーザインタフェース ..... 427

11.0	イントロダクション	427
11.1	コンソールでプログレスインジケータを表示する	429
11.2	lambdaを使わずコールバック関数を書く	431
11.3	tkSimpleDialogの関数にデフォルト値や範囲制限を使う	432
11.4	Tkinterのリストボックスをドラッグアンドドラッグで並び替えるようにする	433
11.5	Tkinterでアクセント記号を入力	435
11.6	Tkinterで使うGIFをコードに埋め込む	437
11.7	画像のフォーマットを変換する	439
11.8	Tkinterによるストップウォッチ	442
11.9	スレッドでGUIと非同期I/Oを結合	444
11.10	TkinterでIDLEのツリーウィジェットを利用する	449
11.11	複数の値を入れられる行(row)を持ったTkinterのリストボックス	451
11.12	Tkinterウィジェット間でジオメトリメソッドやオプションをコピーする	455
11.13	Tkinterでタブを	457
11.14	wxPythonでタブを	460
11.15	JythonでImageJプラグインを実装	461
11.16	SwingとJythonを使ってURLのイメージを表示	462
11.17	Mac OSでユーザに入力させるには	463
11.18	Python Cocoa GUIの「プログラマティックな」構築	465
11.19	IronPythonでフェードインするウィンドウを作る	467

## 12章 XMLの処理 ..... 471

12.0	イントロダクション	471
12.1	整形式(well-formed)であるかチェックする	472
12.2	タグを数える	474
12.3	テキストの抽出	475
12.4	XMLエンコーディングの自動認識	476
12.5	XML文書をPythonのオブジェクトツリーに変換	479
12.6	XML DOMノードのサブツリーからスペースのみのテキストノードを削除	481
12.7	Microsoft ExcelのXMLをパース	482
12.8	XML文書の検証(validate)	484
12.9	ある名前空間に属す要素と属性を除く	486

12.10	連続的なテキストイベントをSAXフィルターで結合	488
12.11	XMLやHTMLのパーズにMSHTMLを使う	491
<b>13章</b>	<b>ネットワークプログラミング</b>	<b>493</b>
13.0	イントロダクション	493
13.1	ソケットデータグラムによるメッセージ伝達	495
13.2	Webドキュメントの取得	497
13.3	FTPサイトリストにフィルタをかける	498
13.4	SNTPプロトコルでサーバから時刻を得る	499
13.5	HTMLメールを送る	500
13.6	MIMEによるファイル添付	503
13.7	MIMEマルチパートメッセージのアンパック	505
13.8	添付ファイルの削除	507
13.9	Python 2.4のemail.FeedParserで解釈されたメッセージを修正	509
13.10	POP3メールボックスを対話的に調べる	511
13.11	アクティブでないコンピュータの検知	514
13.12	HTTPベースのネットワークモニタ	519
13.13	ポートのフォワード、リダイレクト	522
13.14	プロキシでSSLトンネリング	524
13.15	ダイナミックIPアドレスのアップデート	528
13.16	IRCに接続してログを記録する	532
13.17	LDAPサーバへのアクセス	534
<b>14章</b>	<b>ウェブプログラミング</b>	<b>537</b>
14.0	イントロダクション	537
14.1	CGIの動作確認	538
14.2	CGIスクリプトでURLを扱う	541
14.3	CGIでファイルのアップロード	543
14.4	ウェブページの存在確認	544
14.5	HTTPにおけるcontent typeの確認	546
14.6	HTTPファイル転送のレジューム	547
14.7	クッキーの処理	549
14.8	認証に必要なプロキシを通じたHTTPSサイト巡回	552
14.9	Jythonでサーブレットを書く	553
14.10	Internet Explorerのクッキーを探す	555
14.11	OPMLファイルの生成	557
14.12	RSSフィードの収集	560
14.13	テンプレートを使ってデータをウェブ化	564

14.14	任意のオブジェクトをNevowでウェブ化	566
-------	----------------------	-----

## 15章 分散プログラミング ..... 571

15.0	イントロダクション	571
15.1	XML-RPCのメソッドコールをする	573
15.2	XML-RPCサーバ	575
15.3	MedusaによるXML-RPC	577
15.4	XML-RPCサーバをリモートから終了可能に	579
15.5	SimpleXMLRPCServerの小技巧いろいろ	580
15.6	wxPythonでXML-RPCサーバにGUIを	582
15.7	TwistedのPerspective Brokerを使う	584
15.8	CORBAサーバとクライアント	587
15.9	telnetlibでリモートログイン	590
15.10	SSHでリモートログイン	593
15.11	SSLクライアントの認証	596

## 付録A Pythonと日本語処理について ..... 599

A.1	Pythonで日本語を使う	599
A.2	ソースコードに日本語が使いたい	601

## 索引 ..... 605