



Python ライブラリ集

python はコンパイルが不要なスクリプト言語です。逐次翻訳しながら実行するインタープリタ言語に属しますが、一度実行したコードは中間的なコードでキャッシュされる為、“そこそこ速い”です。また文法がシンプルで理解しやすく、ネット上に情報も多いです。それと一番重要なポイントはライブラリが充実している事です。

データの統計処理やデータ解析、さらには機械学習の活用が進み、pip3 コマンドによりライブラリのインストールも簡単なので、気軽に試す事ができます。まずはどのようなライブラリがあるかを知る事で、素晴らしい応用ソフトウェアを作成する事ができます。

自分で作成した処理がライブラリを用いる事で一行で実現できる場合もあります。品質を高めるだけでなく、ソースコードの可読性も高まります。

目次

1. pip3 コマンド.....	4
2. 標準ライブラリ.....	6
2.1. テキスト処理サービス.....	6
2.2. バイナリデータ処理.....	6
2.3. データ型.....	7
2.4. 数値と数学モジュール.....	8
2.5. 関数型プログラミング用モジュール.....	8
2.6. ファイルとディレクトリへのアクセス.....	9
2.7. データの永続化.....	9
2.8. データ圧縮とアーカイブ.....	10
2.9. ファイルフォーマット.....	10
2.10. 暗号関連のサービス.....	11
2.11. 汎用オペレーティングシステムサービス.....	11
2.12. 並行実行.....	12
2.13. ネットワーク通信とプロセス間通信.....	12
2.14. インターネット上のデータの操作.....	13
2.15. 構造化マークアップツール.....	13
2.16. インターネットプロトコルとサポート.....	14
2.17. マルチメディアサービス.....	15
2.18. 国際化.....	15
2.19. プログラムのフレームワーク.....	16
2.20. Tk を用いたグラフィカルユーザインターフェイス.....	16
2.21. 開発ツール.....	16
2.22. デバッグとプロファイル.....	17
2.23. ソフトウェア・パッケージと配布.....	17
2.24. Python ランタイムサービス.....	18
2.25. カスタム Python インタプリタ.....	18
2.26. モジュールのインポート.....	19
2.27. Python 言語サービス.....	19
2.28. 各種サービス.....	20

2.29. MS Windows 固有のサービス.....	20
2.30. Unix 固有のサービス.....	20
3. 外部ライブラリ.....	21
3.1. データ解析に関するライブラリ.....	21
3.2. 機械学習と認識.....	22
3.3. 自然言語処理.....	23
3.4. 音声、画像、動画などのマルチメディア.....	24
3.5. ネットワークと通信.....	25
3.6. Web 開発/便利ライブラリ.....	26
3.7. 画面操作.....	27
3.8. データベース.....	28
3.9. その他、便利なライブラリ.....	29

1. pip3 コマンド

python には標準ライブラリの他に豊富なサードパーティーライブラリが公開されています。python では公開元を以下のサイトで一元的に公開しています。公開されているライブラリは pip3 コマンドでインストールすることができます。

<https://pypi.org/>

ここでは Debian11 で python を使用する場合とします。Debian のデフォルトインストールに pip3 は付属されません。まずは pip3 コマンドをインストールしましょう。最初にシステムを更新しておきます。更新する前にルート権限ユーザとしてログインする必要があります。

```
# apt update  
# apt upgrade
```

以下のコマンドを実行して Python 3 の Pip for Python 3 を Debian にインストールします。

```
# apt install python3-venv python3-pip
```

pip3 を用いてインストールした場合、システムの全てのユーザが使えるか、それとも自分だけ使えるようにするか、あらかじめ設定しておきます。通常はバージョンをそのままに留めて置きたいユーザも居るかも知れませんので、多人数で使う場合は自分のローカルに、自分専用のシステムは全てのユーザに設定します。

全てのユーザにインストールする場合

```
# pip3 install <ライブラリ名>
```

自分だけのローカルにインストールする場合

```
$ pip3 install --user <ライブラリ名>
```

インストール済パッケージの一覧

```
$ pip3 list
```

ライブラリの詳細を表示します

```
$ pip3 show <ライブラリ名>
```

ライブラリを更新する

```
# pip3 install -U <ライブラリ名>
```

新しいバージョンがリリース済みのライブラリ表示

```
# pip3 install -U <ライブラリ名>
```

ライブラリを削除する

```
# pip3 uninstall <ライブラリ名>
```

python のライブラリは他の言語に比べても充実しています。しかも取得が大変に楽なもの、人気言語になりうる理由なのかもしれません。

2. 標準ライブラリ

python の標準ライブラリには以下があります。これらは基本ですので、どのようなライブラリがあるか確認しましょう。標準ライブラリは pip3 によるインストールは不要で import 文ですぐに利用できます。

<https://docs.python.org/ja/3/library/index.html>

2.1. テキスト処理サービス

ライブラリ名	機能概要
string	一般的な文字列操作
re	正規表現操作
difflib	文字列の差分比較を行う。比較結果を html 形式などで出力する事もできます。
textwrap	テキストの折り返しと詰め込み
unicodedata	Unicode データベース
stringprep	文字列が ASCII かや正規表現に可能かなど確認します
readline	GNU readline のインタフェース
rlcompleter	GNU readline 向け補完関数

2.2. バイナリデータ処理

ライブラリ名	機能概要
struct	バイナリデータは Python で bytes 型で表現されます。struct モジュールを使うと細かくフォーマットを指定してバイナリデータを作成したり、バイナリデータから数値等に変換を行うことができます。
codecs	日本語などのマルチバイトで書かれたテキストファイルを文字コード指定で読み込む事などができます。

2.3. データ型

ライブラリ名	機能概要
datetime	基本的な日付型および時間型
zoneinfo	UTC/JST の変換など、タイムゾーンの取扱い
calendar	一般的なカレンダーに関する関数群
collections collections.abc コレクションの抽象基底クラス	地味に便利なデータ構造ライブラリ。配下に以下のデータ取扱いライブラリがある。 namedtuple() 名前付きフィールドを持つタプルのサブクラスを作成するファクトリ関数 deque 両端における append や pop を高速に行えるリスト風のコンテナ ChainMap 複数のマッピングの一つのビューを作成する辞書風のクラス Counter ハッシュ可能なオブジェクトを数え上げる辞書のサブクラス OrderedDict 項目が追加された順序を記憶する辞書のサブクラス defaultdict ファクトリ関数を呼び出して存在しない値を供給する辞書のサブクラス UserDict 辞書のサブクラス化を簡単にする辞書オブジェクトのラッパ UserList リストのサブクラス化を簡単にするリストオブジェクトのラッパ UserString 文字列のサブクラス化を簡単にする文字列オブジェクトのラッパ
heapq	ヒープキューアルゴリズムの実装を提供しているモジュールです。ヒープキューアルゴリズムは、優先度キューアルゴリズムでもあります。キューとは複数の要素が並んだものです。優先度キューとは、ある優先度に従い要素を取り出す仕組みを持ったキューです。ヒープキューとは優先度キューのアルゴリズムの1つで、全ての親の値が、その全ての子の値以下であるようなツリー構造になっており、その構造を生かして効率的に要素を取り出すことができます
bisect	bisectとは、「bisect」は「二分探索」を利用することで効率よく、リストをソートされた状態に保つことをサポートするためのライブラリで、二分探索 (binary search)はソート済みの配列に対する探索アルゴリズムです
array	型の付いたリスト構造です
weakref	ガベージコレクタですぐに解放できる参照(弱参照)
types	オブジェクトのデータ型を取得
copy	オブジェクトのコピーを作ることができます。ただし、コピーには shallowcopy(浅いコピー)と deepcopy(深いコピー)という概念があるので注意
pprint	リスト(list型)や辞書(dict型)などのオブジェクトをきれいに整形して出力・表示したり、文字列(str型オブジェクト)に変換したりすることができます

reprlib	大きなコンテナや、深くネストしたコンテナを省略して、文字列のサイズに制限を設けて、オブジェクト表現を生成する手段を提供します。Python デバッガで使用されています
enum	列挙型と呼ばれ、予め決められた複数の選択肢から値を指定する際に使用される型を指します
graphlib	前後順序のある有向エッジのトポロジカルソート (例えば、前工程のあるタスクの順序の解決等)を行えるライブラリ

2.4. 数値と数学モジュール

ライブラリ名	機能概要
numbers	数の抽象基底クラス
math	度数法・弧度法の変換や三角関数、指数関数、対数関数、円周率、ネイピア数等、数学処理に必要な機能が一通り揃ったライブラリ。複素数は使えない
cmath	複素数の座標変換、数値の判定を行う関数、指数関数、対数関数、三角関数、双曲線関数などのライブラリ
decimal	十進固定及び浮動小数点数の算術演算
fractions	有理数
random	擬似乱数を生成する
statistics	平均値、中央値、分散、標準偏差など

2.5. 関数型プログラミング用モジュール

ライブラリ名	機能概要
itertools	効率的なループ実行のためのイテレータ生成関数
functools	高階関数と呼び出し可能オブジェクトの操作
operator	関数形式の標準演算子

2.6. ファイルとディレクトリへのアクセス

ライブラリ名	機能概要
pathlib	ファイル・ディレクトリ(フォルダ)のパスをオブジェクトとして操作・処理できる。ファイル名・親ディレクトリの抽出やパス一覧の取得、ファイルの作成・削除など、一通りの処理が可能
os.path	パス文字列からファイル名・フォルダ名・拡張子を取得したり、文字列を結合してパスを生成したりする
fileinput	open 関数では1つずつでしか開けないが、fileinput は複数のファイルを順に処理する事ができる
stat	os.stat()、os.lstat() および os.fstat() (存在すれば) の返す内容を解釈するための定数や関数を定義
filecmp	ファイルおよびディレクトリの比較
tempfile	一時ファイルやディレクトリの作成
glob	Unix 形式のパス名のパターン展開
fnmatch	Unix ファイル名のパターンマッチ
linecache	キャッシュを使って、内部で最適化を図りつつ、ファイルの任意の行を取得するのを可能にします
shutil	shutil モジュールが提供する「高水準なファイル操作」とは、ファイルのコピーやディレクトリツリー全体の削除

2.7. データの永続化

ライブラリ名	機能概要
pickle	複数のオブジェクトを1つのまとまりにファイル保存する事ができます。読み込む事もできます。
copyreg	pickle サポート関数を登録する
shelve	Python の辞書と似た形式でオブジェクトを永続化できる。つまり「shelve オブジェクト[キー] = 値」のようにして「値」に指定したオブジェクトを外部ファイルに永続化したり、逆に「値 = shelve オブジェクト[キー]」として外部ファイルからオブジェクトを取り出したりできる
marshal	内部使用向けの Python オブジェクト整列化
dbm	dbm は key-value 型のデータベース。
sqlite3	SQLite を一言でいうと、オープンソースで軽量の RDBMS(データベース管理システム) です。SQL

と名前が似ているため、データベース操作言語の一種と思う方もいますが、SQLite はデータベースです。データベースには、Oracleをはじめ Microsoft SQL Server などの商用データベースが存在します。また、SQLiteと同じくオープンソースの MySQL や PostgreSQL も有名です。SQLite は名前のとおり、簡易的な(ライトな)データベースであり、サーバーとしてではなく Python アプリケーションに組み込むことでローカルに利用します。

2.8. データ圧縮とアーカイブ

ライブラリ名	機能概要
zlib	gzip 互換の圧縮
gzip	gzip ファイルのサポート
bz2	bzip2 圧縮のサポート
lzma	LZMA アルゴリズムを使用した圧縮
zipfile	ZIP アーカイブの処理
tarfile	tar アーカイブファイルの読み書き

2.9. ファイルフォーマット

ライブラリ名	機能概要
csv	CSV ファイルの読み書き
configparser	Microsoft Windows の INI ファイルに似た構造を持ったベーシックな設定用言語を実装した ConfigParser クラスを提供する。 但し python は json 形式など、より高度な構造のファイルも取り扱えます。
netrc	pip3 などが利用する netrc ファイルの処理
xdrlib	XDR データのエンコードおよびデコード

2.10. 暗号関連のサービス

ライブラリ名	機能概要
hashlib	セキュアハッシュおよびメッセージダイジェスト
hmac	メッセージ認証のための鍵付きハッシュ化
secrets	機密を扱うために安全な乱数を生成する

2.11. 汎用オペレーティングシステムサービス

ライブラリ名	機能概要
os	雑多なオペレーティングシステムインタフェース
io	ストリームを扱うコアツール
time	時刻データへのアクセスと変換
argparse	コマンドラインオプション、引数、サブコマンドのパarser
getopt	C 言語スタイルのコマンドラインオプションパーサ
logging	Python 用ロギング機能
logging.config	ロギングの環境設定
logging.handlers	ロギングハンドラ
getpass	可搬性のあるパスワード入力機構
curses	文字セル表示を扱うための端末操作
curses.textpad	curses プログラムのためのテキスト入力ウィジェット
curses.ascii	ASCII 文字に関するユーティリティ
curses.panel	curses のためのパネルスタック拡張
platform	実行中プラットフォームの固有情報を参照する
errno	標準の errno システムシンボル
ctypes	Python のための外部関数ライブラリ

2.12. 並行実行

ライブラリ名	機能概要
threading	スレッドベースの並列処理
multiprocessing	プロセスベースの並列処理
multiprocessing.shared_memory	異なるプロセスから参照可能な共有メモリ
concurrent concurrent.futures	複数の処理を並列実行するための機能を提供します。threading と multiprocessing というモジュールがありますが、これらが1つのスレッド・プロセスを扱うのに対して、concurrent.futures モジュールは複数のスレッド・プロセスを扱うことを目的としています
subprocess	子プロセス(child process)と呼ばれるプロセスを起動したりするためのモジュールです
sched	イベントスケジューラ
queue	同期キュークラス
contextvars	コンテキスト変数
_thread	低水準の スレッド API

2.13. ネットワーク通信とプロセス間通信

ライブラリ名	機能概要
asyncio	非同期 I/O
socket	低水準ネットワークインターフェース
ssl	ソケットオブジェクトに対する TLS/SSL ラッパー
select	I/O 処理の完了を待機する
selectors	高水準の I/O 多重化
asyncore	非同期ソケットハンドラ
asynchat	非同期ソケットコマンド/レスポンスハンドラ
signal	非同期イベントにハンドラを設定する
mmap	メモリマップファイル

2.14. インターネット上のデータの操作

ライブラリ名	機能概要
email	電子メールと MIME 処理のためのパッケージ
json	JSON エンコーダおよびデコーダ
mailcap	mailcap ファイルの操作
mailbox	様々な形式のメールボックス操作
mimetypes	ファイル名を MIME 型へマップする
base64	Base16, Base32, Base64, Base85 データの符号化
binhex	binhex4 形式ファイルのエンコードおよびデコード
binascii	バイナリデータと ASCII データとの間での変換
quopri	MIME quoted-printable 形式データのエンコードおよびデコード
uu	uuencode 形式のエンコードとデコード

2.15. 構造化マークアップツール

ライブラリ名	機能概要
html	HyperText Markup Language のサポート
html.parser	HTML および XHTML のシンプルなパーサー
html.entities	HTML 一般実体の定義
XML を扱うモジュール群	
xml.etree.ElementTree	ElementTree XML API
xml.dom	文書オブジェクトモデル (DOM) API
xml.dom.minidom	最小限の DOM の実装
xml.dom.pulldom	部分的な DOM ツリー構築のサポート
xml.sax	SAX2 パーサのサポート
xml.sax.handler	SAX ハンドラの基底クラス
xml.sax.saxutils	SAX ユーティリティ

xml.sax.xmlreader	XML パーサのインタフェース
xml.parsers.expat	Expat を使った高速な XML 解析

2.16. インターネットプロトコルとサポート

ライブラリ名	機能概要
webbrowser	便利なウェブブラウザコントローラー
cgi	CGI (ゲートウェイインターフェース規格) のサポート
cgitb	CGI スクリプトのトレースバック管理機構
wsgiref	WSGI ユーティリティとリファレンス実装
urllib	URL を扱うモジュール群
urllib.request	URL を開くための拡張可能なライブラリ
urllib.response	urllib で使用するレスポンスクラス
urllib.parse	URL を解析して構成要素にする
urllib.error	urllib.request が投げる例外
urllib.robotparser	robots.txt のためのパーザ
http	HTTP モジュール群
http.client	HTTP プロトコルクライアント
ftplib	FTP プロトコルクライアント
poplib	POP3 プロトコルクライアント
imaplib	IMAP4 プロトコルクライアント
nntplib	NNTP プロトコルクライアント
smtplib	SMTP プロトコルクライアント
smtpd	SMTP サーバー
telnetlib	Telnet クライアント
uuid	RFC 4122 に基づく UUID オブジェクト
socketserver	ネットワークサーバのフレームワーク

http.server	HTTP サーバ
http.cookies	HTTP の状態管理
http.cookiejar	HTTP クライアント用の Cookie 処理
xmlrpc	XMLRPC サーバーとクライアントモジュール
xmlrpc.client	XML-RPC クライアントアクセス
xmlrpc.server	基本的な XML-RPC サーバー
ipaddress	IPv4/IPv6 操作ライブラリ

2.17. マルチメディアサービス

ライブラリ名	機能概要
audioop	生の音声データを操作する
aifc	AIFF および AIFC ファイルの読み書き
sunau	Sun AU ファイルの読み書き
wave	WAV ファイルの読み書き
chunk	IFF チャンクデータの読み込み
coloursys	色体系間の変換
imgchr	画像の形式を決定する
sndhdr	サウンドファイルの識別
ossaudiodev	OSS 互換オーディオデバイスへのアクセス

2.18. 国際化

ライブラリ名	機能概要
gettext	多言語対応に関する国際化サービス
locale	国際化サービス

2.19. プログラムのフレームワーク

ライブラリ名	機能概要
turtle	タートルグラフィックス
cmd	行指向のコマンドインタプリタのサポート
shlex	単純な字句解析

2.20. Tk を用いたグラフィカルユーザインターフェイス

ライブラリ名	機能概要
tkinter	Tcl/Tk の Python インタフェース
tkinter.colorchooser	Color choosing dialog
tkinter.font	Tkinter font wrapper
tkinter.messagebox	Tkinter message prompts
tkinter.scrolledtext	スクロールするテキストウィジェット
tkinter.dnd	Drag and drop support
tkinter.ttk	Tk のテーマ付きウィジェット
tkinter.tix	Tk の拡張ウィジェット

2.21. 開発ツール

ライブラリ名	機能概要
typing	型ヒントのサポート
pydoc	ドキュメント生成とオンラインヘルプシステム
doctest	対話的な実行例をテストする
unittest	ユニットテストフレームワーク
unittest.mock	モックオブジェクトライブラリ
test	Python 用回帰テストパッケージ
test.support	テストのためのユーティリティ関数

test.support.socket_helper	Utilities for socket tests
test.support.script_helper	Utilities for the Python execution tests
test.support.bytecode_helper	Support tools for testing correct bytecode generation

2.22. デバッグとプロファイル

ライブラリ名	機能概要
bdb	デバッガフレームワーク
faulthandler	Python traceback のダンプ
pdb	Python デバッガ
timeit	小さなコード断片の実行時間計測
trace	Python 文実行のトレースと追跡
tracemalloc	メモリ割り当ての追跡

2.23. ソフトウェア・パッケージと配布

ライブラリ名	機能概要
distutils	Python モジュールの構築とインストール
ensurepip	pip インストーラのブートストラップ
venv	仮想環境の作成
zipapp	Manage executable Python zip archives

2.24. Python ランタイムサービス

ライブラリ名	機能概要
sys	システムパラメータと関数
sysconfig	Python の構成情報にアクセスする
builtins	組み込みオブジェクト
warnings	警告の制御
dataclasses	データクラス
contextlib	Utilities for with-statement contexts
abc	抽象基底クラス
atexit	終了ハンドラ
traceback	スタックトレースの表示または取得
__future__	future 文の定義
gc	ガベージコレクタインターフェース
inspect	活動中のオブジェクトの情報を取得する
site	サイト固有の設定フック

2.25. カスタム Python インタプリタ

ライブラリ名	機能概要
code	インタプリタ基底クラス
codeop	Python コードをコンパイルする

2.26. モジュールのインポート

ライブラリ名	機能概要
zipimport	Zip アーカイブからモジュールを import する
pkgutil	パッケージ拡張ユーティリティ
modulefinder	スクリプト中で使われているモジュールを検索する
runpy	Python モジュールの位置特定と実行
importlib	import の実装

2.27. Python 言語サービス

ライブラリ名	機能概要
parser	Python 解析木にアクセスする
ast	抽象構文木
symtable	コンパイラの記号表へのアクセス
symbol	Python 解析木と共に使われる定数
token	Python 解析木と共に使われる定数
keyword	Python キーワードチェック
tokenize	Python ソースのためのトークナイザ
tabnanny	あいまいなインデントの検出
pyclbr	Python module browser support
py_compile	Python ソースファイルのコンパイル
compileall	Python ライブラリをバイトコンパイル
dis	Python バイトコードの逆アセンブラ
pickletools	pickle 開発者のためのツール群

2.28. 各種サービス

ライブラリ名	機能概要
formatter	汎用の出力書式化機構

2.29. MS Windows 固有のサービス

ライブラリ名	機能概要
msilib	Microsoft インストーラーファイルの読み書き
msvcrt	MS VC++実行時システムの有用なルーチン群
winreg	Windows レジストリへのアクセス
winsound	Windows 用の音声再生インタフェース

2.30. Unix 固有のサービス

ライブラリ名	機能概要
posix	最も一般的な POSIX システムコール群
pwd	パスワードデータベースへのアクセスを提供する
spwd	シャドウパスワードデータベース
grp	グループデータベースへのアクセス
crypt	Unix パスワードをチェックするための関数
termios	POSIX スタイルの端末制御
tty	端末制御のための関数群
pty	擬似端末ユーティリティ
fcntl	fcntl および ioctl システムコール
pipes	シェルパイプラインへのインタフェース
resource	リソース使用状態の情報
nis	Sun の NIS (Yellow Pages) へのインタフェース
syslog	Unix syslog ライブラリルーチン群

3. 外部ライブラリ

python には豊富なライブラリがあり、一章の” pip3 コマンド”で簡易にインストールする事ができます。どのような外部ライブラリがあるかを
知る事で、できる幅を拡げる事ができます。ここではこれからプログラムを作成する時に使用すればコーディング量や工期を減らせ、品質向上
も期待できるライブラリを紹介します。各ライブラリにはライセンスがありますので、利用する前に良く確認して下さい。殆どのライブラリは利用
だけでしたら商用利用にも問題ありません。ライブラリ本体を改良した場合に注意する必要があります。

3.1. データ解析に関するライブラリ

一般的にデータ解析用ライブラリは C/C++ 言語で作成される為、動作速度が python のみで作成するより断然に速いです。

ライブラリ名	機能概要	ドキュメント
numpy	多次元配列の数値のみを対象とし、計算速度が速いです。配列に対し、四則演算、累乗、平方根、三角関数、指数関数、対数関数、双曲線関数、切り捨て、切り上げ、四捨五入、複素数、絶対値、平均、中央値、標準偏差、分散、共分散、相関係数、行列演算・・・等が配列に対し計算できます。 python のみでループで回したら負けと思ひましょう。しかも配列のメモリ占有量が数ギガになるなど多量になる配列の場合、速度差は歴然となります。	https://numpy.org/doc/
matplotlib	numpy データを元に、折れ線グラフ、散布図、棒グラフ、ヒストグラム、円グラフ、箱ひげ図、バイオリンプロットなどのグラフ描画ライブラリ。	https://matplotlib.org/stable/users/index.html
cupy	numpy 互換ライブラリで高性能なグラフィックスボードで計算させる事で更なる高速度で計算させる事ができる。NVIDIA 製 GPU とドライバをインストールした上で使用する。自分専用スパコンをご自宅に置きたい人向け。	https://docs.cupy.dev/en/stable/
pandas	numpy をベースに CSV ファイル、エクセルファイルなどの入出力を追加し、数値の他、文字列、時系列データなどのデータを取扱える。SQL に似た操作や統計処理も可能。但し数値計算は numpy のほうが高速に動作する。	https://pandas.pydata.org/pandas-docs/stable/
scipy	数学、科学、工学などのライブラリ。例えば線形代数演算、FET、最適化、統計処理、補完、積分、信号解析、画像処理などの機能がある。	https://docs.scipy.org/doc/scipy/reference/

3.2. 機械学習と認識

機械学習とは経験をさせて学習し自動的に改善を繰り返していくコンピュータアルゴリズムの研究分野で人口知能の一種と見なす事もできる。例えば野菜のキュウリの選別作業のケースを考えて下さい。まっすぐに出荷にピッタリサイズで色艶がきれいなら A 級品、サイズだけ短いか長いなら B 級品、曲がってたら C 級品の選別作業をアルバイト A は間違いなくこなすが、アルバイト B は時々間違える出荷作業があったとする。学習させた選別機器を製作することで、ほぼ間違える事が無く、全てのアルバイトが A さんレベルになるようにできる。(※かも知れない。学習レベルによる。)

ライブラリ名	機能概要	ドキュメント
tensorflow	google が作成した、データフローグラフを用いた数値計算フレームワーク。機械学習のプラットフォームとしても有名。開発された目的は、人間が用いる学習や論理的思考と似たように、パターンや相関を検出し解釈するニューラルネットワークを構築、訓練することができるシステム。	https://www.tensorflow.org/?hl=ja
nervananeon (neon)	深層学習フレームワーク インテルがバックアップしている。深層学習(ディープラーニング)とは、コンピュータが人間の脳神経回路を模して物事やデータをより正確かつ効率的な判断を実現させる技術を挿します。	https://www.intel.co.jp/content/www/jp/ja/artificial-intelligence/overview.html
keras	深層学習ライブラリ TensorFlow ライブラリの上で動くような作りになっており、より簡単にモデルを作ることができるように設計されています。	https://keras.io/ja/
scikit-learn	機械学習ライブラリ 機械学習とは、コンピュータがデータから反復的に学習しそこに潜むパターンを見つけ出すデータ解析テクニックです。新たなデータを入力すると、学習したパターンにしたがって結果を予測させる事ができます。 scikit-learn は様々な機械学習の手法が統一的なインターフェースで利用できるようになっています。	https://scikit-learn.org/stable/user_guide.html

3.3. 自然言語処理

SNSでのトレンド分析やアンケートからの分析など、自然言語(例えば日本語)を解析したい場合があります。自然言語処理とは端的に言うと、「人間の言葉を機械が理解するルール作り」です。

自然言語処理は一般的に形態素解析、構文解析(係り受け解析)、意味解析、文脈解析の工程からなります。それぞれの役割は次の通りです。

形態素解析:文字で表記された自然言語の文を、意味を持つ最小単位(形態素)まで解析する

構文解析(係り受け解析):分解した形態素の関係性を解析する

意味解析:コンピュータの持つ「辞書」をもとに正しく意味内容を解析する

文脈解析:文同士の関係を解析する

これらの機能を利用することで、検索エンジン、機械翻訳、かな漢字文字変換、対話システム、テキストマイニングなどの処理を高速にこなせます。

ライブラリ名	機能概要	ドキュメント
mecab-python3	形態素解析エンジン MeCab というプログラムを利用する python ライブラリ。最初に MeCab をインストールし、次に必要な辞書データ、そして python ライブラリをインストールして環境を構築します。	https://pypi.org/project/mecab-python3/
janome	python で記述された辞書内包の形態素解析ライブラリ。依存ライブラリ等無しで簡単にインストールでき、アプリケーションに組み込みやすいシンプルな API	https://mocobeta.github.io/janome/
spacy	大量のテキストの処理および理解を行うアプリの構築に役立ちます。「情報抽出」「自然言語理解」「深層学習のテキストの前処理」に使用できます。	https://spacy.io/
ginza	「GiNZA」は、オープンソースな日本語の自然言語処理ライブラリです。最先端の機械学習技術を取り入れた自然言語処理ライブラリ「spaCy」をフレームワークとして利用しており、トークン化処理に形態素解析器「SudachiPy」が使われています。	https://megagonlabs.github.io/ginza/

3.4. 音声、画像、動画などのマルチメディア

pythonには画像のトリミングや拡大・縮小、画像の合成など画像を処理する分野においても様々な画像処理をするライブラリがあります。特にOpenCVは、画像系の機械学習の前処理などに使われます。音声はGStreamerという音声や動画を取扱うフレームワークをpythonから利用するライブラリがあります。

ライブラリ名	機能概要	ドキュメント
pillow	画像認識など高度な画像処理を行なうことのできるOpenCVと比較して、単純な操作や基本的な操作を行なうことができるという点がPillowの特徴です。リサイズ(拡大・縮小)や回転、トリミング(部分切り出し)のような単純な処理がより簡単にできる。	https://python-pillow.org/
opencv-python	以下の機能があります。 <ul style="list-style-type: none"> ・画像のトリミング・リサイズ・重ね合わせ ・画像の回転・上下反転・左右反転 ・グレースケール変換・色チャンネル分析・減色処理 ・モザイク処理・マスク処理・2枚の画像を合成 ・図形の描画・文字の描画 ・ノイズ除去・平滑化・ぼかしフィルタ・メディアンフィルタ・ガウシアンフィルタ ・物体検出 ・テンプレートマッチング 	https://opencv.org/
scikit-image	・他のライブラリ「NumPy」「SciPy」「Pandas」「Matplotlib」とやり取りしやすい ・機械学習ライブラリscikit-learnと組み合わせやすい 画像処理と機械学習アルゴリズムを組み合わせたい場合は、OpenCVよりもこちらの方がより簡易	https://scikit-image.org/
gstreamer python binding	マルチメディア・フレームワークGstreamerをpythonから扱う為のライブラリ。GStreamerはビデオ編集ソフトやストリーミング、そしてメディアプレーヤーなどのようなマルチメディアアプリケーションのベースとなる機能を提供する。	https://gstreamer.freedesktop.org/
pyaudio	オーディオライブラリPortAudioを利用できる。音声の再生や録音が可能。	https://people.csail.mit.edu/hubert/pyaudio/docs/
python-vlc	VLCのPythonバインディング。VLCはフリーなマルチプラットフォーム対応のマルチメディアプレーヤーであり、DVD、オーディオCD、VCDや様々なストリーミングプロトコルを再生可能なフレームワークです。	https://wiki.videolan.org/Python_bindings/

3.5. ネットワークと通信

rasberry pi などの小さいワンボードで IoT を実現するには、通信機能や無線が欠かせません。このような場合にも python にはライブラリが揃っているので、まずは試して効果を確認してからシステム作りに入れます。

ライブラリ名	機能概要	ドキュメント
pyserial	シリアル通信ライブラリ。IoT では今でも使用されています。	https://pyserial.readthedocs.io/en/latest/
paramiko	SSH 接続や SCP 転送を行うライブラリ。SSH 接続して遠隔操作などを行う事ができる。	http://docs.paramiko.org/en/stable/
requests	Python には標準で urllib というライブラリが存在しますが、Requests はそれよりもシンプルに、人が直感的に分かりやすいプログラムを記述できます	https://docs.python-requests.org/en/master/index.html
ws4py	ソケットライブラリです	https://ws4py.readthedocs.io/en/latest/
python-nmap	ポートスキャン用ライブラリ	http://xael.org/pages/python-nmap-en.html
wifi	wifi ネットワークの検出、設定、接続用ライブラリ	https://wifi.readthedocs.io/en/latest/
pyblues	pybluez は Linux で標準搭載されている Bluetooth プロトコルの Bluez を Python でラッパーしたライブラリ	https://pybluez.readthedocs.io/en/latest/index.html

3.6. Web 開発/便利ライブラリ

今は Web 開発として外せないのは「WordPress」ではないでしょうか。オープンソースの CMS でライブラリも多量にあるので、最初に Web サイトを作るとしたら外せません。WordPress でしたら Python より PHP になりますね。(◦)ゞ

ここでは Python での Web 開発ライブラリを説明します。フレームワークライブラリやクライアントから利用するライブラリを説明します。

WordPress は「完成品としての Web サイト、フレームワークは Web サイトを作る為の道具」が例えになります。

ライブラリ名	機能概要	ドキュメント
django	利用実績が多く良く使われる Web フレームワーク Django を使えば、コンテンツ管理システムや Wiki からソーシャルネットワーク、ニュースサイトなど、高品質な Web アプリケーションを簡単に、少ないコードで作成できます	https://www.djangoproject.com/
flask	小規模向けの簡単な Web アプリケーションを作るのに適しています	https://palletsprojects.com/p/flask/
tornado	Tornado は Python で作られた Web フレームワークで、非同期通信を行うライブラリ。 非同期通信とは、その名の通りネットワークで接続されているサーバーとアプリケーションが同期を取らずに通信を行う方式で、リクエストを送信してからレスポンスが来るまでの間に他の処理が実行できるというメリットがある。	https://www.tornadoweb.org/en/stable/
feedparser	RSS/Atom パーサライブラリ	https://feedparser.readthedocs.io/en/latest/
BeautifulSoup	BeautifulSoup は、HTML や XML ファイルからデータを取得し、解析する Python の WEB スクレイピング用のライブラリです。インターネット上に公開されている WEB サイトでは広く HTML や XML が使われており、これらの情報の取得や解析に、大変便利なライブラリです。	https://www.crummy.com/software/BeautifulSoup/
pytube	Youtube から音声・動画をダウンロードするライブラリです。ダウンロードしたファイルは個人で利用する場合以外は規約違反になりますので注意して下さい。 最高画質でダウンロードする場合は、音声と動画を別々にダウンロードし、ffmpeg ライブラリで、結合操作が必要になります。	https://pytube.io/en/latest/

3.7. 画面操作

単独アプリケーションとしての GUI ライブラリや三次元グラフィックスライブラリがあります。
ターミナル(VT100 端末)の場合、python が標準で持つ curses ライブラリがあります。

ライブラリ名	機能概要	ドキュメント
pygtk	PyGTK は、グラフィカルなソフトウェアの作成を支援するライブラリ GTK を Python 言語から利用可能にするライブラリです。商用利用可。	https://python-gtk-3-tutorial.readthedocs.io/en/latest/#
pyqt5	qt5 を利用するためのライブラリ。素晴らしいライブラリであるが、商用利用時は別途ライセンスが必要。	https://pypi.org/project/PyQt5/
PyOpenGL	様々な環境、ハードウェア、OS に対応したマルチプラットフォームで利用できる汎用グラフィックスライブラリで 2D グラフィックス・3D グラフィックスどちらも扱うことが可能。OpenGL は、Intel や NVIDIA、AMD が開発・製造しているグラフィックボード (GPU) を活用して、高速な映像処理を実現しています。	http://pyopengl.sourceforge.net/
sdl2.ext	SDL (Simple DirectMedia Layer) は、C 言語で書かれたクロスプラットフォームのマルチメディアライブラリである。グラフィックの描画やサウンドの再生などの API を提供する。オーディオ、キーボード、マウス、ジョイスティック、そして OpenGL および Direct3D を経由したグラフィックスハードウェアへのローレベルなアクセスを提供するよう設計されている。	https://pysdl2.readthedocs.io/en/latest/

3.8. データベース

データベースは外せません。データベースこそ情報を保存し、自在に扱えてこそ、使い勝手の良いアプリケーションを作成することができます。オープンソースから商用まで豊富にあります。オープンソースでお勧めは PostgreSQL です。商用に負けず劣らず大規模な商用システム開発に使われています。Python 標準では sqlite3 があります。

ライブラリ名	機能概要	ドキュメント
cx_Oracle	cx_Oracle は Python の DB 接続 AP に基づいて実装され、OracleDB に簡単にアクセスするためのモジュールです。	https://oracle.github.io/python-cx_Oracle/
py-postgresql	Python3 系のみ動作、libpq 非依存の PostgreSQL 向けドライバとツール類です。	https://pythonhosted.org/py-postgresql/
mysql-connector-python	MySQL データベース用モジュール ・MySQL の開発元、Oracle 社によってアクティブに開発されている ・依存が Python の標準ライブラリだけなのでインストールが楽	https://dev.mysql.com/doc/connector-python/en/

3.9. その他、便利なライブラリ

今までの分類に入らないけど、知っておくととても便利なライブラリを掲載します。

ライブラリ名	機能概要	ドキュメント
openpyxl	エクセルファイルを読み書きするライブラリです。Linux 上でエクセルファイルを取扱えます。	https://openpyxl.readthedocs.io/en/stable/



本文書はフリーランスエンジニアである Far East Information Engineering が作成しました。
自由に利用できますが、無保証になります。

<https://feie.tech/>

ライセンスは以下とさせていただきます。

<https://doclicenses.opensource.jp/GFDL-1.2/GFDL-1.2.html>



本文書は Libre Office にて作成しました。

<https://ja.libreoffice.org/>

ユーザーガイド

<https://documentation.libreoffice.org/ja/documentation-in-japanese/>