**Rangkuman Library Populer pada Python**

Kita sudah berada di penghujung materi Library Populer pada Python. Sampai sejauh ini, Anda diharapkan telah memahami library populer pada Python yang beragam dan cara penggunaannya. Sekarang, mari kita rangkum secara saksama.

**Pengenalan Library**

Library adalah koleksi dari banyaknya modul dan paket yang saling terkait dan dapat digunakan berulang kali. Paket atau package adalah sebuah direktori yang berisi satu atau lebih modul yang terkait dan saling berhubungan. Jumlah library Python sangat banyak yang terbagi menjadi **Python Standard Library**dan **Python External Library**.

Python Standard Library adalah jenis library yang telah terinstal secara otomatis ketika kita melakukan instalasi Python. Anda tidak perlu melakukan instalasi kembali jika ingin menggunakannya. Beberapa contoh Python Standard Library adalah “os”, “datetime”, “re”, serta lainnya yang dapat Anda baca lebih lengkap pada laman [berikut](https://docs.python.org/3/library/).

Python External Library adalah kumpulan kode yang telah dikembangkan oleh orang lain atau komunitas dan disediakan dalam bentuk paket atau modul yang dapat diimpor. Jenis library ini mengharuskan Anda untuk melakukan instalasi agar dapat digunakan. External library ini dikembangkan oleh individu atau organisasi di luar tim inti pengembang Python.

Untuk melakukan instalasi library eksternal, Anda dapat melakukan beberapa cara, seperti menggunakan PIP dan conda.

**PIP**

PIP adalah package manager resmi dari Python yang dapat digunakan untuk mengunduh, menginstal, menghapus, dan mengelola package Python dari Python Package Index (PyPI) dan repositori lainnya. [PyPI](https://pypi.org/) merupakan repositori online yang menyediakan ribuan package dari Python yang siap digunakan oleh para pengembang.

Selain mengelola paket, Anda juga bisa membuat lingkungan virtual dalam Python menggunakan PIP. Hanya saja, pip cenderung difokuskan untuk mengelola instalasi paket dibanding lingkungan virtual.

Kabar baiknya, pip biasanya telah terpasang secara otomatis jika Anda menggunakan Python 2 untuk versi 2.7.9 ke atas atau Python 3 untuk versi 3.4 ke atas. Maka dari itu, silakan periksa bahwa pip telah terpasang dengan menjalankan perintah berikut.

|  |
| --- |
| pip --version |

Jika lokal komputer Anda belum memiliki pip, Anda bisa mengikuti langkah-langkah berikut.

1. Unduh file berikut: <https://bootstrap.pypa.io/get-pip.py>.
2. Buka terminal dan buka folder tempat Anda menyimpan file yang telah diunduh. Kemudian jalankan perintah berikut.

|  |
| --- |
| python get-pip.py |

Sekarang, Anda telah menyiapkan pip sebagai package manager yang mendukung instalasi package dan library Python. Untuk melakukan instalasi, Anda bisa mengikuti perintah berikut.

|  |
| --- |
| pip install <nama-package> |

Ganti <nama-package> dengan nama package atau library yang ingin Anda unduh. Untuk menghapus package, Anda bisa mengikuti perintah berikut.

|  |
| --- |
| pip uninstall <nama-package> |

Silakan ganti <nama-package> dengan package atau library yang ingin Anda hapus.

**Conda**

Conda adalah package manager dan environment manager untuk Python. Conda memungkinkan kita untuk membuat dan mengelola lingkungan (environment) terisolasi atau terpisah satu sama lain. Dengan terisolasinya setiap lingkungan tersebut, menguntungkan kita untuk mencegah konflik yang terjadi antar proyek.

Dengan adanya lingkungan yang terisolasi, Anda bisa menyesuaikan semua library, modul, hingga versi Python sesuai dengan kebutuhan masing-masing proyek. Conda sendiri dapat diinstal melalui Anaconda dan Miniconda.

Anaconda adalah sistem distribusi perangkat lunak yang di dalamnya mencakup Conda. Ketika menginstal Anaconda, Anda pun dapat menggunakan beberapa library dan plugin tambahan melalui Anaconda tersebut. Miniconda adalah versi ringan dari Anaconda. Miniconda hanya berisi conda dan beberapa package dasar yang diperlukan untuk menjalankannya.

Conda sendiri hadir dalam dua bentuk utama: “conda” sebagai package dan environment manager serta “conda-forge” sebagai sebuah repositori berisi ribuan paket yang disediakan oleh komunitas conda.

Kelebihan Conda adalah sifat tidak terbatasnya pada bahasa pemrograman Python. Conda mendukung paket-paket dalam bahasa pemrograman lain, seperti R.

**Library Text Processing**

Library text processing bertujuan untuk melakukan pemrosesan teks dan menyederhanakan serta mempercepat tugas-tugas pemrosesan teks. Beberapa library populer terkait pemrosesan teks adalah berikut.

1. String  
   [String](https://docs.python.org/3/library/string.html)adalah salah satu modul bawaan Python yang tidak perlu dideklarasikan. Pada modul string ada fungsi-fungsi yang dapat dioperasikan pada variabel bertipe string seperti di bawah.
   * upper(): Ubah setiap huruf dalam string menjadi huruf kapital.
   * lower(): Ubah setiap huruf dalam string menjadi huruf kecil.
   * split(): Pisahkan teks berdasarkan delimiter (karakter pemisah).
   * title(): Jadikan setiap awal kata kapital.
   * zfill(): Tambahkan nol di awal string sebanyak nilai yang ada pada parameter.
2. Regex  
   [Regex](https://docs.python.org/3/library/re.html) atau regular expression adalah sebuah cara untuk mencari teks berdasarkan pola tertentu. Umpamanya, ketika ingin mencari sebuah kata dalam kamus, misalnya arti dari kata parsing, kita akan mencari kata tersebut di halaman yang memiliki kata dengan awalan p, lalu pa. Regex bekerja dengan konsep yang sama. Pada regex, kita mencari sebuah kata atau kumpulan kata dengan memberikan pola yang diinginkan.

**Library Matematika**

Library yang dapat digunakan untuk permasalahan matematika adalah library [math](https://docs.python.org/3/library/math.html) yang termasuk salah satu modul bawaan Python. Anda hanya perlu melakukan impor untuk modul math. Berikut contoh penerapannya.

1. import math
3. print(math.sqrt(25))  # Output: 5.0
4. print(math.pi)  # Output: 3.141592653589793

**Library Parser**

Library parser pada Python menyediakan fasilitas untuk menguraikan kode Python menjadi struktur data yang dapat diproses dan dianalisis. Anda dapat menggunakan [Getopt](https://docs.python.org/3.8/library/getopt.html) atau [ArgParse](https://docs.python.org/3.8/library/argparse.html).

Argument parser bermanfaat jika kita ingin membuat program atau skrip kecil yang langsung menerima parameter pada saat pemanggilan program. Hal ini biasa digunakan dalam pemanggilan aplikasi atau skrip di CLI/terminal \*nix-based, misalnya Linux dan MacOS. Contoh penerapannya berikut.

1. import argparse
3. parser = argparse.ArgumentParser()
4. parser.add\_argument('-o', '--output', action='store\_true', help="tampilkan output")
5. args = parser.parse\_args()
7. if args.output:
8. print("Halo, ini merupakan sebuah output dari panggildicoding.py")

**Library Pengolahan Data**

Library pengolahan data bertujuan untuk membantu dalam manipulasi, analisis, dan pemrosesan data. Library ini menyediakan berbagai fungsi dan metode yang memudahkan pengguna untuk melakukan operasi pengolahan data dengan lebih efisien dan cepat.

Tujuan dari library ini untuk menyederhanakan tugas-tugas kompleks yang berkaitan dengan pengolahan data. Jadi, Anda tidak perlu mengimplementasikan semuanya dari awal. Berikut adalah beberapa library populer yang digunakan untuk pengolahan data.

1. **Pandas**  
   [Pandas](https://pandas.pydata.org/) adalah library populer yang digunakan untuk pengelolaan dan analisis data. Library ini menyediakan struktur data dan alat untuk membantu pengguna dalam melakukan manipulasi, pembersihan, transformasi, dan analisis data dengan mudah dan efisien.
2. **NumPy**  
   Library [NumPy](https://numpy.org/) adalah package fundamental yang sering digunakan untuk scientific computing pada Python. Library ini menyediakan objek array multidimensi, berbagai jenis objek lainnya, seperti masked array dan matrix, dan sebagainya.
3. **Matplotlib**  
   Selanjutnya adalah [matplotlib](https://matplotlib.org/) yang merupakan library untuk melakukan visualisasi data. Matplotlib termasuk jenis library eksternal sehingga Anda perlu melakukan instalasi matplotlib terlebih dahulu.
4. **Seaborn**  
   Terakhir adalah library [seaborn](https://seaborn.pydata.org/) yang termasuk jenis library dengan tujuan untuk visualisasi data sama seperti matplotlib. Bahkan library seaborn dibangun berdasarkan pada library matplotlib.

**Library File Management**

Library file management adalah kumpulan library yang dirancang untuk membantu pengguna dalam mengelola serta berinteraksi dengan berkas dan direktori pada sistem file. Beberapa library file management adalah berikut.

1. OS  
   Modul [OS](https://docs.python.org/3/library/os.html) pada Python berguna untuk fungsi-fungsi yang berkaitan dengan sistem operasi, misalnya open(), path(), getcwd(), dan fungsi lainnya. Modul ini memungkinkan Anda untuk memanfaatkan fungsi yang sama dan mengeksekusi fungsi terkait OS yang mungkin berbeda di setiap sistem operasi. Ada beberapa fitur yang hanya bekerja pada sistem operasi tertentu.
2. JSON  
   Untuk serialization dengan bahasa lain, umumnya kita menggunakan [JSON](https://docs.python.org/3/library/json.html)(JavaScript Object Notation) yang memiliki beberapa perbedaan karakteristik dengan pickle, yakni berikut.
   * JSON adalah format text-serialization dan umumnya menggunakan Unicode atau UTF-8. Sementara pickle bersifat binary serialization.
   * JSON dapat dibaca dengan mudah oleh manusia, sementara pickle tidak.
   * JSON dapat dioperasikan dan digunakan di luar ekosistem Python. Pickle adalah Python-specific.
   * JSON secara default hanya dapat merepresentasikan subset dari built-in type pada Python.
   * Pickle dapat merepresentasikan hampir (jika tidak seluruh) tipe Python dan secara default melakukan kompresi data.  
       
     Sebagaimana yang telah disebutkan sebelumnya, JSON adalah format text yang ditujukan untuk serialization. Agar data dapat dengan mudah ditransmisikan antar berbagai sumber tanpa khawatir bentuknya kacau, menggunakan JSON adalah salah satu pilihan yang tepat.  
       
     JSON memiliki format yang hampir mirip dengan dictionary tempat data disimpan dengan format key & value pair.
3. **Pickle**  
   Jika Anda memiliki sebuah list yang ingin disimpan atau ditransmisikan tanpa khawatir bentuknya akan rusak atau kacau, fungsi dari library pickle dapat dimanfaatkan. Pickle termasuk fungsi Object Serialization pada Python. Pickling adalah istilah untuk mengubah objek menjadi byte stream, sedangkan unpickling adalah perlakuan sebaliknya.

**Library Web Scraping**

Library web scraping adalah jenis library untuk membantu pengguna mengumpulkan data dari halaman web. Proses ini disebut sebagai “web scraping” atau “web crawling”. Anda bisa menggunakan fungsi dan metode pada library ini untuk mengekstraksi informasi dari situs web dan menyimpannya dalam format yang dapat diakses dan digunakan dalam analisis atau aplikasi lainnya.

1. **Beautifulsoup**  
   Beautifulsoupadalah library untuk mengambil data dari halaman web dan mengekstrak informasi yang diperlukan.
2. **Urllib**  
   Urllibadalah library bawaan dari Python yang bertujuan untuk scraping konten dari sebuah website. Penggunaan urllib berbeda dengan beautifulsoup. Bisa dikatakan bahwa cara penggunaan urllib sedikit kompleks dibandingkan beautifulsoup.

**Library Machine Learning**

Selanjutnya adalah library yang digunakan untuk melakukan pemelajaran mesin. Anda dapat menggunakan library berikut untuk membantu Anda menyelesaikan permasalahan machine learning. Berikut adalah beberapa library populer untuk machine learning.

1. **Scikit-learn**  
   Pertama adalah scikit-learn yang menyediakan berbagai algoritma pemelajaran mesin siap pakai untuk membantu dalam pengembangan model pemelajaran mesin, pemrosesan data, dan evaluasi kinerja model.
2. **TensorFlow**  
   Selanjutnya adalah TensorFlow yang termasuk salah satu library paling populer terkait machine learning. Dengan menggunakan TensorFlow, Anda bisa mengembangkan machine learning hingga tahap deployment.
3. **PyTorch**  
   Terakhir ada PyTorch, yakni library machine learning yang dikembangkan oleh Facebook’s AI Research lab (FAIR). PyTorch menyediakan alat dan kerangka kerja yang kuat untuk mengembangkan model pemelajaran mesin, terutama dalam konteks jaringan saraf tiruan (neural networks).

**Library Web Development**

Terakhir, ada library yang bertujuan untuk pengembangan aplikasi web. Sebagaimana yang sudah dijelaskan dalam materi-materi sebelumnya, Python dapat digunakan untuk pengembangan aplikasi web pada sisi server. Berikut adalah library yang dapat digunakan untuk membantu Anda mengembangkan web.

1. **Django**  
   Django adalah high-level Python web framework yang mendukung pengembangan secara cepat, bersih, serta pragmatis.
2. **Flask**  
   Flask adalah web framework untuk Python yang ditujukan untuk membangun aplikasi web. Flask dirancang dengan tujuan menjadi ringan, fleksibel, dan sederhana.
3. **Fast API**  
   FastAPI adalah web framework untuk Python yang tujuannya merancang dan membangun API dengan cepat, efisien, dan aman. FastAPI memberikan kinerja yang tinggi, sintaks yang intuitif, serta dukungan otomatisasi dokumentasi yang kuat. Jadi, ia cocok untuk pengembangan mikroservis, layanan web responsif, dan sebagainya.