

Nama : Asri Ramdani
Nim : 19.01.013.034
Kelas / MK : Pemrograman Phyton (INF020)

1.Cari tentang Phyton

Python adalah sebuah bahasa pemrograman yang digunakan untuk membuat aplikasi, perintah komputer, dan melakukan analisis data. Sebagai *general-purpose language*, Python bisa digunakan untuk membuat program apa saja dan menyelesaikan berbagai permasalahan. Selain itu, Python juga dinilai mudah untuk dipelajari. Namun, jangan salah, Python termasuk bahasa pemrograman tingkat tinggi. Mulai dari profesi *back-end developer*, IT, sampai *data scientist*, Python benar-benar menjadi pilihan favorit.

Python memiliki *endless capabilities*, ada banyak sekali hal yang bisa ciptakan dengannya. Jika disusun berdasarkan bidang pekerjaan, Python bisa membantu industri seperti:

- **Matematika:** menyelesaikan permasalahan matematika seperti aljabar, kalkulus, dan trigonometri.
- **Penulisan skrip sistem:** membuat perintah secara otomatis dan menyelesaikan pekerjaan yang memakan waktu.
- **Software development:** *bug tracking* dan *testing software*.
- **Web development:** *URL routing*, memastikan keamanan *website*, memproses dan mengirim data.
- **Data analysis:** melakukan kalkulasi statistik, visualisasi data, dan menganalisis data.
- **Machine learning:** membuat algoritma untuk modul pembelajaran.

2.Pemrograman Language beserta contohnya

Bahasa Pemrograman (**programming language**) adalah sebuah instruksi standar untuk memerintah komputer agar menjalankan fungsi tertentu. Bahasa pemrograman ini merupakan suatu himpunan dari aturan sintaks dan semantik yang dipakai untuk mendefinisikan program komputer. Melalui *programming language*, seorang programmer memberi tahu komputer apa yang harus ia lakukan apabila diminta untuk mengerjakan suatu hal tertentu. Contoh dari *programming language* adalah Java, C, C++, C#, dan lain-lain.

3.Perbedaan dari markup Language dan contohnya

Markup language bukan bahasa untuk mengajari komputer, melainkan cara untuk menggambar dan menulis suatu halaman *web*. Halaman *website* tidak bisa ditulis layaknya menulis di atas kertas. Oleh karena itu, halaman *website* ditulis dengan *markup language*. Saat mengakses

suatu *website* melalui aplikasi *browser*, yang akan dikirimkan sejatinya banyak tulisan yang ditulis dengan *markup language*. Aplikasi *browser* yang akan menerjemahkan *markup language* ini, sehingga tampilan *website* menjadi rapi dan menarik. Contoh dari *markup language* adalah HTML, CSS, XML, dan lain-lain.

4.Perbedaan pemrograman konvensional dan visual beserta contoh

Pengertian dari Pemrograman konvensional adalah perancangan program dimulai dari perencanaan dan mendefinisikan tujuan program, menuliskan kode, dan Langkah terakhir merancang keluaran dari program beserta hubungan dengan pemakai. Pemrograman visual adalah perencanaan dan mendefinisikan tujuan program, Langkah berikutnya adalah merancang keluaran, dan media hubungan dengan pemakai, dan Langkah terakhir adalah menuliskan kode program tersebut. Contoh pemrograman konvensional adalah java dan turbo pascal. Contoh pemrograman visual adalah netbeand dan visual C++

5.Perbedaan hard code dan soft code beserta contoh

Hard code merupakan salah satu praktek yang dilakukan saat menuliskan kode dengan menaruh data secara langsung di kode sumber / *source code* kita. Hard Coding" berarti sesuatu yang ingin embed dengan program atau proyek apa pun yang tidak dapat diubah secara langsung. Misalnya jika menggunakan server database, maka harus hardcode untuk menghubungkan database dengan proyek dan itu tidak dapat diubah oleh pengguna. Karena memiliki memiliki kode keras. Sedangkan Soft code adalah fitur menggunakan engine yang memperoleh hasil berdasarkan sejumlah nilai parametrik ; aturan diberi kode tetapi ada sebagai parameter dalam sistem dan ditulis dalam bentuk skrip.

Contoh :

```
//firstName has a hard-coded value of "hello world"
```

```
String firstName = "hello world";
```

```
//firstName has a non-hard-coded provided as input
```

```
Console.WriteLine("first name :");
```

```
String firstName = Console.ReadLine();
```

6.Perbedaan interpreter dan compailer dan contoh

Kategori	Compiler	Interpreter
Penggunaan	Source code telah dikonversi menjadi machine code. Sehingga waktu eksekusi program akan lebih singkat.	Lebih mudah digunakan untuk pemula yang baru belajar.
Hasil keluaran	Menghasilkan program luaran atau berkas executable. Contohnya seperti .exe yang dapat dijalankan secara independent	Tidak menghasilkan program luaran atau berkas executable. Jika ingin menjalankan program, maka harus melibatkan source code secara langsung selama proses eksekusi.
Efektifitas	Hasil kompilasi dari source code akan berjalan lebih cepat.	Berjalan lebih lambat ketika dieksekusi.
Platform	Spesifik ke platform tertentu, misal hasil kompilasi berupa berkas.exe tidak dapat dijalankan di Mac. Begitu pula sebaliknya.	Cross platform. Bisa dijalankan di banyak platform asalkan memiliki interpreter yang sesuai.
Alur pembacaan	Apabila terdapat kesalahan penulisan kode maka compiler akan menampilkan pesan error. Selain itu program tidak akan berjalan atau tidak menghasilkan berkas executable apabila kesalahan kodenya belum diperbaiki. Misal ada kode yang salah di baris ke-5 maka harus diperbaiki dulu supaya bisa berjalan.	Membaca satu per satu baris kode yang ada. Jika terdapat kode yang salah maka interpreter akan menampilkan pesan error dan harus diperbaiki untuk melanjutkan eksekusi baris selanjutnya. Misalnya terdapat error di baris ke-5, maka masih bisa berjalan karena baris pertama sampai keempat tidak ada error.

Kategori	Compiler	Interpreter
Bahasa Pemrograman	Bahasa pemrograman yang memerlukan compiler seperti C, C++, C#, Swift, Java	Bahasa pemrograman yang memerlukan interpreter seperti JavaScript, Python, PHP, Ruby.

Contoh Bahasa Pemrograman yang menggunakan Compiler dan Interpreter

.Menggunakan Compiler:

- Visual Basic (VB)
- Fortran
- Cobol
- Pascal
- C family (C, C++, C#)

7.Perbedaan Phyton dengan PL lainnya

- Dibanding bahasa pemrograman lain, Phyton relatif lebih mudah dipelajari. Bahasa ini memiliki sintaks-sintaks yang cukup sederhana dan gampang dimengerti.
- Selain memiliki keterbacaan kode yang tinggi, sehingga kode mudah dipahami, bahasa pemrograman ini memiliki library yang sangat banyak dan luas. Berbagai macam jenis library ini memuat beragam perlengkapan dan fungsionalitas.
- Mendukung ekosistem Internet of things (IoT) dengan sangat baik.
- Program yang ditulis menggunakan Phyton dapat dijalankan di hampir semua sistem operasi (Unix, Windows, Mac OS X, dll) Melalui mekanisme tertentu, kode Python dapat diintegrasikan dengan aplikasi yang ditulis dalam bahasa pemrograman lain.

8.Perbedaan IDE dan text editor

Text Editor merupakan aplikasi yang hanya berguna untuk mengedit source code saja, tanpa fitur lain yang dimiliki oleh IDE. Namun text editor ini lebih ringan daripada IDE, sehingga banyak orang yang menggunakannya.

IDE merupakan singkatan dari Integrated Development Environment adalah program komputer yang memiliki beberapa fasilitas yang diperlukan dalam pembangunan perangkat lunak. IDE juga dapat mengedit source code, memiliki debugger, dan juga fitur code completion.

Perbedaan yang terdapat dari Text Editor dengan IDE hanyalah fitur-fiturnya saja, Text Editor cenderung lebih enteng dan tidak berat seperti IDE, karena tugasnya sebagai mengedit sebuah Plain Text, namun Text Editor memiliki banyak Extension yang bisa digunakan sehingga

bisa mempercepat pekerjaan. Sedangkan, IDE memiliki fitur yang lebih banyak dibandingkan Text Editor, seperti :

- Editor, yaitu fasilitas untuk menuliskan kode sumber dari perangkat lunak.
- Compiler, yaitu fasilitas untuk mengecek sintaks dari kode sumber kemudian mengubah dalam bentuk binari yang sesuai dengan bahasa mesin.
- Linker, yaitu fasilitas untuk menyatukan data binari yang beberapa kode sumber yang dihasilkan compiler sehingga data-data binari tersebut menjadi satu kesatuan dan menjadi suatu program komputer yang siap dieksekusi.
- Debugger, yaitu fasilitas untuk mengetes jalannya program, untuk mencari kesalahan yang terdapat dalam program.

Contoh Text Editor

- Notepad++
- Sublime Text
- Sublime Text
- Visual Studio Code

Contoh IDE

- Arduino IDE
- Android Studio
- XCode
- IntelliJ

9. IDE khusus untuk python dan text editor untuk python

1. Pycharm

Tersedia dalam tiga versi, versi Komunitas berlisensi Apache, versi Pendidikan (Edu), dan versi Profesional berlisensi. Dua versi pertama adalah open source dan dapat digunakan secara gratis, sedangkan versi Profesional merupakan berbayar.

2. Pydev

Pydev adalah plugin yang memungkinkan Eclipse digunakan sebagai IDE Python yang juga mendukung Jython dan IronPython. Eclipse adalah lingkungan pengembangan terintegrasi (IDE) yang secara historis dirancang untuk bahasa Java. Namun, berkat sistem plugin atau ekstensi,

Pydev pun dapat digunakan dengan bahasa pemrograman lain, termasuk C / C ++ dan PHP. Pydev menggunakan teknik inferensi lanjutan untuk menyediakan elemen seperti penyelesaian kode dan analisis kode. IDE juga menyediakan fitur berikut: debugger, Django, konsol interaktif, penyorotan sintaks dasar, cakupan kode, dll. Ini adalah salah satu Ide terbaik untuk python dan Editor Python terbaik juga.

3. Visual Studio Code

Dikembangkan oleh Microsoft untuk berbagai sistem operasi. Visual Studio Code (atau disingkat pula dengan VS Code) adalah editor kode yang dapat diekstensikan dan perangkat lunak ini adalah perangkat lunak open source di bawah lisensi MIT.

Ini yang membedakan Visual Studio dengan VS Code. VS Code pertama kali diterbitkan pada tanggal 29 April 2015. IDE ini pun sebanding dengan Atom dalam hal fungsionalitas.

4. Atom IDE

Atom adalah salah satu editor teks terbaik untuk Python. Atom mencakup sebagian besar fungsi IDE dasar. Beberapa fitur yang dimiliki oleh Atom diantaranya adalah *syntax highlighting* dan *auto-completion*.

5. IDLE

IDLE merupakan singkatan dari Integrated Development and Learning Environment. IDLE dirilis pada bulan Desember 1998 oleh Guido Van Rossum untuk pengembangan Python. IDLE merupakan IDE sederhana dan oleh karena itu IDLE sangat cocok digunakan untuk pemula. IDLE terdiri dari editor teks multi-window dengan *syntax highlighting* dan debugger terintegrasi dengan *stepping*, *persistens breakpoint*, dan *call stack visibility*.

6. Spyder

Pada versi pertamanya, Spyder dinamai sebagai Pydee yang dibuat oleh Pierre Raybaut pada tahun 2008. Spyder adalah IDE lintas platform dan di bawah lisensi non-copyleft (Perangkat lunak bebas non-copyleft dirilis oleh pembuatnya dengan izin untuk mendistribusikan dan memodifikasinya). Sejak 2012, komunitas ilmiah Python yang telah memelihara Spyder berkat kontribusinya.

7. Sublime text

Pertama kali dipahami sebagai ekstensi untuk Vim, Sublime Text adalah editor teks umum yang dikodekan dalam C++ dan Python. Sejak versi 2.0, software ini mendukung 44 bahasa pemrograman utama, termasuk Python.

Sublime text ini dikembangkan oleh Jon Skinner dan pertama kali dirilis pada tahun 2007. Untuk membuat perangkat lunak ini, Jon Skinner mengadopsi tiga prinsip:

- Antarmuka yang minimalis agar dapat fokus pada teks dan bukan pada banyak toolbar
- Teks tidak disembunyikan oleh *window*
- Gunakan ruang sebanyak mungkin: pengeditan file layar penuh, multi-layar, berdampingan harus dimungkinkan.

9. Vim

Vim adalah editor teks yang memungkinkan manipulasi file teks. Kode sumbernya dikembangkan oleh Bram Moolenaar dan pertama kali dirilis pada tahun 1991. Sejak itu, perangkat lunak telah berkembang dan ditingkatkan oleh pengembangnya dan juga oleh kontribusi komunitas pengembang.

10. GNU/Emacs

Dibuat pada tahun 1976 oleh Richard Stallman, Emacs adalah keluarga editor teks dengan serangkaian fitur yang dapat diperluas. Emacs merupakan singkatan dari Editing MACroS

10. Perbedaan Bahasa natural,algoritma,flowchart dan pseudocode

a. Algoritma Bahasa natural

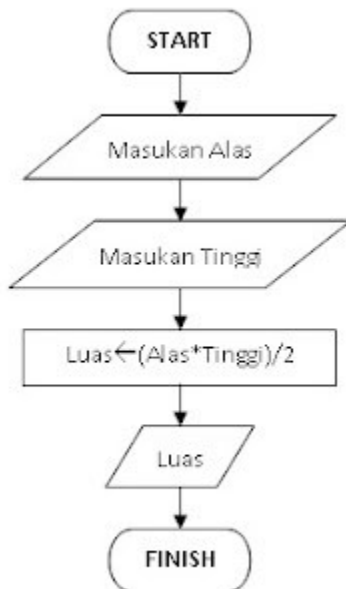
Algoritma bahasa natural merupakan cara penyajian suatu algoritma yang paling sederhana dan paling gampang untuk dimengerti. Algoritma ditulis menggunakan bahasa yang kita pakai sehari-hari (bahasa Indonesia), atau bahasa apapun yang dipahami. Tidak ada hukum standar dalam hal penamaan Start & End itu sendiri, tidak problem jika Anda ingin menunjukkan penamaan yang tidak sama yang terpenting ialah permanen mengandung arti/maksud yang sama. Contoh: Salah satu pola penulisan bahasa natural algoritma dalam kehidupan sehari-hari adalah, Menukar dua isi gelas, yaitu gelas berisi teh & kopi, gelas berisi teh sebagai isinya kopi & begitu sebaliknya.

b. Algoritma Flowchart

Flowchart adalah suatu skema dengan simbol-simbol tertentu yang menggambarkan urutan proses secara mendetail dan korelasi antara suatu proses (instruksi) dengan proses lainnya dalam suatu program.

Contoh:

Berikut adalah salah satu pola memakai flowchart buat menyelesaikan perkara menghitung luas segitiga:



c. Algoritma Pseudocode

Secara sederhana algoritma pseudocode ialah penulisan langkah-langkah penyelesaian persoalan menggunakan pendekatan arahan bahasa pemrograman, dengan tujuan semoga lebih gampang dipahami saat arahan tersebut ditulis atau dikonversi kedalam bahasa pemrograman sebenarnya. Penulisan algoritma pseudocode bergotong-royong tidak mengenal hukum syntax khusus, namun syntax atau arahan yang dipakai umumnya disesuaikan dengan bahasa pemrograman yg nanti akan dipakai.