

Komparasi Kinerja Integrated Development Environment (IDE) Dalam Mengeksekusi Perintah Python

Ismail Setiawan

Sistem Dan Teknologi Informasi, Fakultas Sains Dan teknologi, Universitas ‘Aisyiyah Surakarta, Surakarta, Indonesia
Email: ismail@aiska-university.ac.id

ABSTRACT

An IDE is basically a software package consisting of tools used to develop and test software. IDEs help automate developer tasks by reducing manual effort and combining all tools in a common framework. If the IDE does not exist, then the developer will have to do the selection, integration, and deployment process manually. This study tries to compare the performance of ideas in recognizing writing errors or the simplest syntax in the python language, namely indentation. The test results of the 6 IDEs used found 1 IDE that managed to detect syntax writing errors, namely w3schools. The choice lies with the developer. Generally, the idea is used for programs such as compiled languages such as C, C++, Java, .NET. While the code editor is used for scripts and interpreted, such as PHP, JavaScript, Python, Ruby. In terms of features, of course, using an IDE is more complete than a text editor. If the developer uses VS code, the environment can be adjusted like a code editor but still carries the reliability of an IDE.

Keywords: IDE, Syntax, Python

ABSTRAK

IDE pada dasarnya adalah paket perangkat lunak yang terdiri dari peralatan yang digunakan untuk mengembangkan dan menguji perangkat lunak. IDE membantu mengotomatiskan tugas pengembang dengan mengurangi upaya manual dan menggabungkan semua peralatan dalam kerangka umum. Jika IDE tidak ada, maka developer harus melakukan proses seleksi, integrasi, dan deployment secara manual. Penelitian ini mencoba untuk membandingkan kinerja ide dalam mengenali kesalahan penulisan atau sintaks yang paling sederhana pada bahasa python yaitu indentansi. Hasil pengujian dari 6 IDE yang digunakan didapati 1 IDE yang berhasil mendeteksi kesalahan penulisan sintaks yaitu w3schools. Pilihan ada pada pengembang umumnya Ide dipakai untuk program seperti compiled language seperti C, C++, Java, .NET. Sedangkan kode editor dipakai untuk skrip dan interpreted, misalnya PHP, JavaScript, Python, Ruby. Dari sisi fitur tentu lebih lengkap menggunakan IDE dibanding teknik editor. Apabila pengembang menggunakan VS code maka lingkungannya dapat disesuaikan seperti teknik editor namun tetap mengusung kehandalan sebagai sebuah IDE.

Kata Kunci: IDE, Sintaks, Python

1. Pendahuluan

IDE adalah singkatan dari Integrated Development Environment. IDE pada dasarnya adalah paket perangkat lunak yang terdiri dari peralatan yang digunakan untuk mengembangkan dan menguji perangkat lunak. IDE membantu mengotomatiskan tugas pengembang dengan mengurangi upaya manual dan menggabungkan semua peralatan dalam kerangka umum. Jika IDE tidak ada, maka developer harus melakukan proses seleksi, integrasi, dan deployment secara manual. IDE pada dasarnya dikembangkan untuk

menyederhanakan proses SDLC (*system development life cycles*), dengan mengurangi pengkodean dan menghindari kesalahan pengetikan.

Berbeda dengan IDE, beberapa pengembang juga lebih memilih Kode editor. Kode Editor pada dasarnya adalah editor teks di mana pengembang dapat menulis kode untuk mengembangkan perangkat lunak apa pun. Kode editor juga memungkinkan pengembang untuk menyimpan file teks kecil untuk kode tersebut.

Dibandingkan dengan IDE, kode editor cepat dalam pengoperasian dan memiliki ukuran yang kecil. Bahkan kode editor memiliki kemampuan mengeksekusi dan men-debug kode. Penelitian ini mencoba untuk membandingkan kinerja sebuah IDE dalam mengenali kesalahan penulisan sintaks pada python.

2. Metode Penelitian

Berikut ini adalah 10 Python IDE dan Kode editor yang dipakai oleh Pengembang.

1. Pycharm

IDE ini menonjol dari segi persaingan karena produktivitas tools yang ada di dalamnya seperti quick fixes (perbaikan cepat) [1]. Tersedia dalam tiga versi, versi Komunitas berlisensi Apache, versi Pendidikan (Edu), dan versi Profesional berpemilik. Dua versi pertama adalah open source dan dapat Pengembang gunakan secara gratis, sedangkan versi Profesional merupakan berbayar.

Versi Komunitas dari IDE ini mungkin akan sangat menarik bagi Pengembang karena memiliki fitur berbeda seperti penyorotan sintaks, auto-completion, dan verifikasi kode secara langsung. Versi berbayar memiliki fitur yang lebih canggih seperti manajemen database penuh dan banyak framework yang lebih penting daripada versi komunitas seperti Django, Flask, Google App, Engine, Pyramid, dan web2py.

Keunggulan:

- Dukungan komunitas yang aktif.
- Verifikasi kode langsung dan penyorotan sintaksis
- Menjalankan pengeditan dan debug kode Python tanpa persyaratan eksternal apa pun

Kekurangan:

- Waktu pemuatan yang cenderung lambat
- Pengaturan default mungkin memerlukan penyesuaian sebelum proyek yang ada dapat digunakan.

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 [2]. 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.

Pydev tersedia secara gratis, tidak bergantung platform, dan mempromosikan integritas yang sangat baik dengan Eclipse. Berkat upaya para pengembang dan evolusi teknologi, pengembangan Python menjadi hampir sebanding dengan pengembangan Java di Eclipse, membuat pengalamannya jauh lebih baik.

Keunggulan:

- Mudah dipelajari dan juga menyertakan beberapa fitur (CPython, Jython, IronPython)
- Plugin Pydev untuk membuat kode dengan Python di Eclipse
- Penyelesaian Kode dan Analisis Kode

Kekurangan:

- Antarmuka pengguna tidak terlalu baik.

3. Visual Studio Code

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

Inilah 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.

VS Code adalah proyek Microsoft dengan jumlah kontributor tertinggi di GitHub. Inisiatif ini telah meningkatkan ketenaran Microsoft dan memposisikannya sebagai salah satu pemain utama dalam perangkat lunak komunitas. Pengembang dapat menambahkan bahasa baru ke lingkungan, seperti Python menggunakan VS Code.

Cukup unduh dan instal plugin yang sesuai untuk menyesuaikannya dengan lingkungan. VS Code ditingkatkan dengan fitur seperti integrasi mesin penyelesaian otomatis kode yang kuat (IntelliSense), konsol debugging dan terminal untuk meluncurkan perintah server.

VS Code dirancang dengan sangat baik secara keseluruhan dan keuntungan utamanya adalah mereka menawarkan arsitektur berbasis ekstensi [3]. Karena IDE dari Microsoft ini cukup ringan, ia pun dapat diperluas dengan menambahkan komponen berurutan sesuai kebutuhan.

Keunggulan:

- Memiliki lebih dari 4700 ekstensi

- Code Management Machine (Mesin Manajemen Kode) yang kuat
- 3 Impor pintasan keyboard sesuai permintaan dari editor Python lain seperti Sublime Text atau Atom

Kekurangan:

- Pengembang akan kesulitan menemukan ekstensi yang paling sesuai dengan kebutuhan Pengembang karena ribuan ekstensi yang tersedia

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 autocompletion.

Pengembang dari Atom sendiri sedang mengerjakan integrasi bahasa pemrograman utama seperti Rust atau Go. Atom membuat kemajuan dalam meningkatkan kinerjanya, dan pengembang sangat memperhatikan kebutuhan dan pendapat komunitas, berusaha untuk membuat pengalaman pengguna lebih bermanfaat.

Ada peningkatan waktu tunggu untuk membuka perangkat lunak dan peningkatan kecil dalam waktu yang diperlukan untuk penghitungan penting guna mengurangi latensi. Namun, kemajuan yang lebih dan lebih diharapkan dalam beberapa bulan mendatang.

Salah satu alasan yang berkontribusi pada kesuksesan Atom adalah antarmukanya yang dapat disesuaikan sepenuhnya [4]. Semuanya bisa diubah, dari antarmuka hingga fungsi dasar. Di sisi lain, inisiatif yang agak menguntungkan ini juga merupakan salah satu masalah yang menjadi akar dari latensi program. Namun secara keseluruhan, Atom tetap merupakan IDE yang sangat praktis dan fungsional.

Keunggulan:

- Antarmuka yang dapat disesuaikan sepenuhnya
- Didokumentasikan dengan baik
- Menawarkan fitur hampir semua fitur yang ada pada VS Code

Kekurangan:

- Konsumsi RAM yang cukup besar.
- Peningkatan lebih lanjut diperlukan dalam latensi dan pengoptimalan sistem.
- Tidak terlalu cocok untuk menangani file kode besar

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 [4].

Keunggulan:

- IDLE dapat digunakan untuk menjalankan satu statement.
- Dapat digunakan untuk membuat, memodifikasi, dan menjalankan skrip Python.
- Menawarkan fitur seperti syntax highlighting, auto-completion & indentasi cerdas.
- Memiliki debugger dengan fitur melangkah & breakpoint.

Kekurangan:

- IDLE tidak tersedia secara default dalam distribusi Python untuk Linux.
- Membutuhkan manajer paket masing-masing untuk instalasi.

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.

Spyder memiliki fitur dasar seperti syntax highlighter dan auto-completion dan juga mengintegrasikan banyak pustaka penggunaan ilmiah seperti Matplotlib, Numpy, IPython, Scipy [5]. Spyder adalah IDE bersifat open-source dan gratis dan juga instalasi sangat mudah berkat pengelola paket Python.

Spyder menawarkan kombinasi unik dari analisis lanjutan, debugging, pengeditan, eksekusi interaktif, inspeksi mendalam, dan kemampuan visualisasi dari paket perangkat lunak ilmiah. Komponen utama ini adalah konsol interaktif, penampil dokumentasi, penjelajah variabel, dan development tools.

Keunggulan:

- Dukungan komunitas yang cukup besar.
- Kaya akan fitur alat pengembangan.
- Dokumentasi lengkap

Kekurangan:

- Dependensi pada eksekusi
- Dependensi opsional

7. Thonny

Thonny adalah IDE yang dikembangkan oleh University of Tartu di Estonia. Perangkat lunak ini telah dirancang terutama untuk mempermudah para developer pemula yang baru menggunakan Python dengan menyediakan IDE yang sederhana dan ringan. Namun, dengan fitur-fitur unggulan, ini agak mirip dengan beginner kit. Oleh karena itu, perangkat lunak ini sangat cocok untuk pemula yang ingin memulai pemrograman dan pengembangan dengan Python dan oleh karena itu sama sekali tidak cocok untuk pakar pengembangan [6].

Antarmuka pengguna diisolasi dari semua fitur yang dapat mengalihkan perhatian para pemula. Hal ini adalah kursus pedagogis yang dipikirkan dengan matang untuk pemula yang ingin mengembangkan Python dengan cepat, mudah, dan sederhana.

Keunggulan:

- IDE diadaptasi untuk pembelajaran pemula
- Antarmuka pengguna dasar dan fungsional
- Tidak membutuhkan memori yang besar untuk menjalankannya

Kekurangan:

- Kurang cocok digunakan untuk para ekspertis.
- Hanya menyediakan fungsi dasar

8. 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.

Setelah Pengembang menginstal Sublime Text sebagai Python IDE berkat plugin dan paket yang berbeda, Pengembang mendapatkan IDE berkualitas tinggi dan kuat yang memungkinkan Pengembang mengembangkan dengan sangat mudah dengan Python [7]. Sublime Text menggabungkan sebagian besar fitur editor teks Python dasar, termasuk penyorotan sintaks yang dapat disesuaikan.

Keunggulan:

- Cepat dengan sedikit bug (keuntungan besar)
- Dapat membuka file besar
- Mendukung banyak bahasa

Kekurangan:

- Sulit untuk dimodifikasi karena seluruhnya melalui JSON.
- Lisensi diperlukan
- Mempelajari jalan pintas

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.

Vim berbeda dari kebanyakan editor teks Python lainnya dalam mode operasinya, perhatikan bahwa Vim memiliki tiga mode dasar: mode insert, mode normal atau mode perintah dan mode baris perintah. Vim adalah perangkat lunak gratis dan sebagian besar dapat disesuaikan dengan menambahkan ekstensi atau memodifikasi file konfigurasinya [8]. Hal ini berarti Pengembang dapat dengan mudah menyesuaikannya untuk pengembangan dengan Python.

Perangkat lunak ini mencakup banyak fitur seperti multi-buffer yang mendukung pengeditan file secara bersamaan, indentasi otomatis yang disesuaikan dengan bahasa, dan banyak lainnya. Untuk mengunduh Vim, cukup klik tautan berikut <https://www.vim.org/download.php>.

Keunggulan:

- Perangkat lunak kaya fitur dan memberikan pengalaman pengguna yang baik dengan dukungan komunitas.
- Pengenalan dan konversi format file (UNIX, MS-DOS atau Mac)

Kekurangan

- Fitur inovatif tidak mencukupi

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 Running on TECO. Perangkat lunak gratis ini sangat populer di komunitas pengembang.

Richard Stallman menekankan fakta bahwa perangkat lunak harus sepenuhnya gratis dan dapat disesuaikan

tanpa batasan dalam pemrograman. Setelah beberapa tahun, banyak versi Emacs telah muncul, termasuk GNU Emacs, yang diprakarsai oleh Richard Stallman pada tahun 1984 dan XEmacs diluncurkan pada tahun 1991.

GNU Emacs menggunakan bahasa ekstensi yang kuat yang disebut Emacs Lisp yang mendukung tugas-tugas tingkat lanjut seperti menulis, menyusun program, menjelajahi internet, membaca email, dan forum diskusi.

Perangkat lunak ini tersedia untuk diunduh gratis dan tersedia di semua platform. GNU Emacs menggunakan berbagai skrip penyesuaian untuk pengembangan dalam beberapa bahasa, termasuk Python.

Untuk pengembangan Python, Pengembang dapat menggunakan ekstensi Elpy di Emacs untuk mengerjakan proyek Python. Perangkat lunak ini sangat kaya fitur [9]. Beberapa fitur yang dimiliki oleh Emacs diantaranya syntax highlighting untuk membedakan elemen dokumen seperti kata kunci dan komentar; Indentasi otomatis agar pemformatan yang konsisten dalam file dan berbagai fitur lainnya.

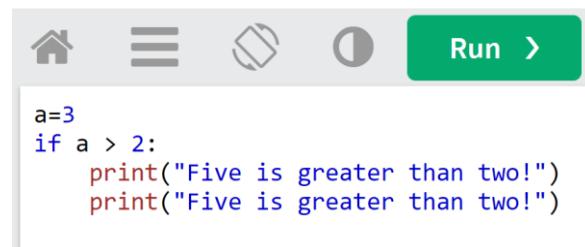
Keunggulan:

- Emacs tersedia secara gratis dan dapat disesuaikan sepenuhnya.
- Penyisipan otomatis elemen yang diperlukan oleh struktur dokumen, seperti spasi, baris baru, dan tpingembang kurung;
- Dukungan yang dikodekan warna 24-bit untuk terminal yang mendukungnya.

Kekurangan:

- Tidak mudah digunakan untuk pemula dalam pemrograman

Pada penelitian ini IDE yang digunakan diantaranya adalah pycharm[1] WebTigerJython [10], visual studio code [3], google colab [11], w3schools [12], onlinedb [13] dan pyscripter [14]. Pengujian dilakukan dengan memberikan perintah if dan print dengan indentasi tab dan spasi. Pemilihan IDE tersebut melihat dari kemudahan dalam



```
a=3
if a > 2:
    print("Five is greater than two!")
    print("Five is greater than two!")
```

Gambar 1. IDE w3schools

3. Hasil dan Pembahasan

Fitur-Fitur Penting dalam Python IDE dan Kode editor

1. Syntax highlighting

Syntax highlighting berfungsi untuk menyoroti berbagai tipe syntax yang berbeda secara otomatis. Fitur ini sangat penting agar syntax mudah dibaca. Misalnya dalam suatu kode, variabel memiliki warna putih, kata kunci atau keyword berwarna hijau. Syntax highlighting inilah yang bertugas melakukan hal tersebut, sehingga pengembang lebih mudah memahami syntax yang di lihat.

2. Automatic code formatting

IDE dan Kode editor yang mendukung Python dapat secara otomatis melakukan indentasi, menambahkan kolom dan memasukan seluruh end brackets yang hilang berdasarkan aturan style Python. Hal ini berarti Pengembang dapat menghabiskan lebih sedikit waktu untuk mengubah-ubah format sehingga Pengembang memiliki lebih banyak waktu untuk melakukan pengkodean.

3. Debugging

Walaupun tools yang berbeda dapat melakukan debug secara langsung. Sebaiknya tool ini memiliki kemampuan untuk membuat kode Pengembang secara bertahap untuk melacak perubahan. Idealnya, Pengembang menginginkan tool yang dapat menunjukkan atau menyoroti potensi kesalahan dalam kode Pengembang sebelum dilakukan runtime.

4. Build automation

Pengembang memilih alat yang memungkinkan Pengembang untuk menulis dan menjalankan kode Python dalam jendela (window) yang sama. Pengembang dapat mengkopi dan menyalin source code atau kode sumber compiler eksternal untuk menjalankannya sehingga membebani Pengembang dengan waktu ekstra dan mengurangi hal-hal yang tidak perlu.

5. Save and reload

Seluruh IDE dan kode editor semestinya berisikan beberapa cara untuk menyimpan dan memuat ulang kode dalam state yang sama. Fitur ini adalah tool terbaik dan menyertakan kontrol versi yang memungkinkan Pengembang untuk kembali ke versi sebelumnya jika dibutuhkan.

Berikut adalah sintaks sederhana yang digunakan dalam pengujian dari masing masing IDE.

```
a = 4
if a > 5:
    (indentasi tab)           print ("salah")
    (indentasi dengan spasi)  print ("salah")
```

Tabel berikut menjelaskan hasil pengujian terhadap skrip diatas. Apabila IDE di uji dengan skrip diatas dan memunculkan hasil seperti apa yang tertulis dalam perintah print maka dianggap lolos atau tidak bisa

mendeteksi indentasi perintah print yang seharusnya menjadi aturan dasar dalam penulisan sintaks.

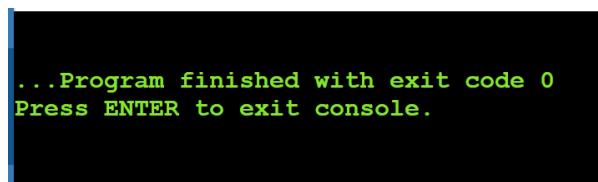
Table 1. hasil pengukuran kinerja

No	IDE	Hasil
1	WebTigerJython	Lolos
2	Visual studio code	Lolos
3	Google colab	Lolos
4	W3schools	Terdeteksi / tidak lolos
5	Onlinegdb	Lolos
6	Pyscripter	Lolos

Hasil eksekusi dari masing masing IDE

3.1 WebTigerJython

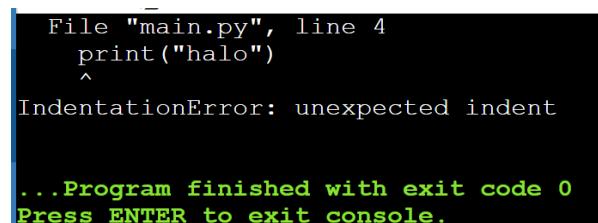
Website ini menyediakan IDE secara online saat dilakukan pengetesan dengan perintah yang benar hanya memunculkan interface.



```
...Program finished with exit code 0
Press ENTER to exit console.
```

Gambar 2. Hasil output WebTigerJython

Sedangkan saat ada kesalahan dalam penulisan kode IDE akan menampilkan output seperti berikut



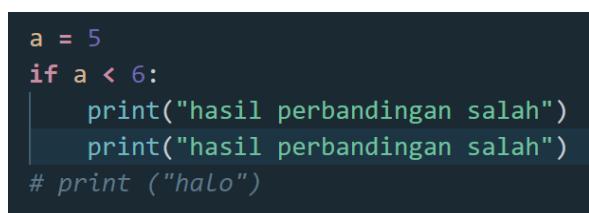
```
File "main.py", line 4
    print("halo")
    ^
IndentationError: unexpected indent

...Program finished with exit code 0
Press ENTER to exit console.
```

Gambar 3. Hasil Ouput Error

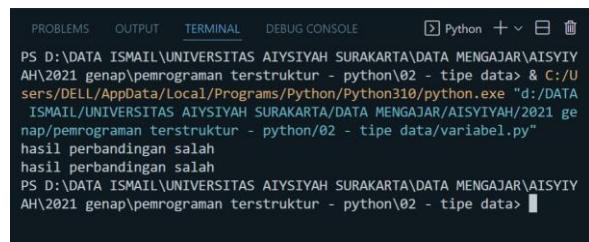
3.2 Visual studio code

Visual studio code adalah IDE yang bersifat free. Hasil pengujian dengan sour code yang benar dan salah akan tetap menampilkan output



```
a = 5
if a < 6:
    print("hasil perbandingan salah")
    print("hasil perbandingan salah")
# print ("halo")
```

Gambar 3. Source code yang digunakan

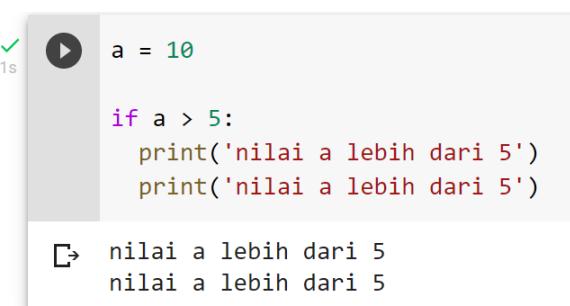


```
PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE Python + 
PS D:\DATA ISMAIL\UNIVERSITAS AIYSIYAH SURAKARTA\DATA MENGAJAR\AIYSIYAH\2021 genap\pemrograman terstruktur - python\02 - tipe data> & C:/Users/DELL/AppData/Local/Programs/Python/Python310/python.exe "d:/DATA ISMAIL\UNIVERSITAS AIYSIYAH SURAKARTA\DATA MENGAJAR\AIYSIYAH/2021 genap\pemrograman terstruktur - python\02 - tipe data/variabel.py"
hasil perbandingan salah
hasil perbandingan salah
PS D:\DATA ISMAIL\UNIVERSITAS AIYSIYAH SURAKARTA\DATA MENGAJAR\AIYSIYAH\2021 genap\pemrograman terstruktur - python\02 - tipe data> 
```

Gambar 4. Output yang dihasilkan

3.3 Google colab

Google colab adalah aplikasi yang banyak digunakan untuk melakukan data science [15]. Secara sederhana google colab dapat mengenali perintah python dan dapat mengeksekusinya secara langsung. Hasil pengujian langsung dimunculkan saat tombol play diklik. Dapat dilihat google colab tidak memperhatikan indentasi yang diketik menggunakan tab atau spasi.



```
✓ 1s a = 10

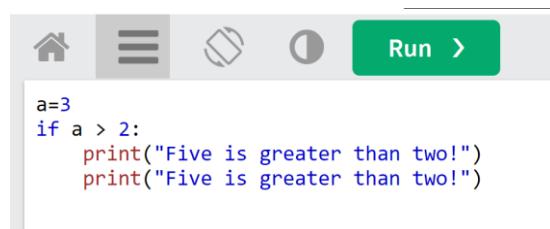
if a > 5:
    print('nilai a lebih dari 5')
    print('nilai a lebih dari 5')

 nilai a lebih dari 5
 nilai a lebih dari 5
```

Gambar 5. Interface google colab

3.4 W3schools

W3scholl adalah ruang belajar yang dapat digunakan secara gratis kapan saja dimana saja [16]. Pengujian untuk mengetahui aturan sintaks pyhton terkait indentasi dilakukan dengan mencoba code berikut



```
a=3
if a > 2:
    print("Five is greater than two!")
    print("Five is greater than two!")
```

Gambar 6. Interface W3scholl

```

Traceback (most recent call last):
  File "/usr/lib/python3.8/py_compile.py", line 144, in compile
    code = loader.source_to_code(source_bytes, dfile or file,
      File "<frozen importlib._bootstrap_external>", line 846, in source
      File "<frozen importlib._bootstrap>", line 219, in _call_with_framesound
      File "./prog.py", line 4
        print("Five is greater than two!")
        ^
IndentationError: unindent does not match any outer indentation level

During handling of the above exception, another exception occurred:

Traceback (most recent call last):
  File "<string>", line 1, in <module>
    File "/usr/lib/python3.8/py_compile.py", line 150, in compile
      raise py_compile.PyCompileError("Sorry: IndentationError: unindent does not
py_compile.PyCompileError: Sorry: IndentationError: unindent does not

```

Gambar 6. W3scholl mendeteksi indetansi yang salah

3.5 Onlinegdb

Secara umum interface onlinegdb bisa dikatakan mirip dengan visual studio code [17]. Karena dapat melakukan autocomplete sintaks yang diketikan. Untuk pengujian diberikan perintah seperti gambar 7. Dan hasilnya ditunjukan pada gambar 8.

```

1 a = 4
2 if a < 6:
3     print ("halo")
4     print("halo")

```

Gambar 7. Interface W3scholl

```

halo
halo

...Program finished with exit code 0
Press ENTER to exit console.

```

Gambar 8. Hasil jika kode benar

```

File "main.py", line 4
    print("halo")
    ^
IndentationError: unexpected indent

...Program finished with exit code 0
Press ENTER to exit console.

```

Gambar 9. Hasil jika kode salah

3.6 Pyscripter

PyScripter memiliki semua fitur yang diharapkan dalam Python IDE modern dalam paket yang ringan [18]. Kode juga dikompilasi secara langsung di sistme operasi Windows untuk menggabungkan konsumsi memori minimal dengan kinerja maksimum. PyScripter adalah program yang bersifat open-source yang dikembangkan menggunakan Delphi dengan ekstensibilitas melalui skrip Python. Source code yang digunakan dapat dilihat pada gambar 10 dan hasil eksekusi ditunjukan pada gambar 11.

```

a = 4
if 5 > 2:
    print("Five is greater than two!")
    print("Five is greater than two!")

```

Gambar 10. Interface pyscripter

```

>>>
*** Remote Interpreter Reinitialized ***
Five is greater than two!
Five is greater than two!
>>>

```

Gambar 11. Hasil output pyscripter

4. Kesimpulan

Pentingnya penulisan sintaks yang benar menjadi kebiasaan yang sangat penting bagi pengembang. Karena akan mempersingkat waktu penggeraan dan memudahkan kolaborator dalam memahami sintaks yang digunakan. Penggunaan IDE menjadi alat bantu yang dapat dipilih selain text editor. Karena penggunaan IDE akan semakin mempercepat dalam penulisan kode program. Pilihan ada pada pengembang umumnya Ide dipakai untuk program seperti compiled language seperti C, C++, Java, .NET. Sedangkan kode editor dipakai untuk skrip dan interpreted, misalnya PHP, JavaScript, Python, Ruby. Dari sisi fitur tentu lebih lengkap menggunakan IDE disbanding teks editor. Apabila pengembang menggunakan VS code maka lingkungannya dapat di sesuaikan seperti kode editor namun tetap mengusung kehandalan sebagai sebuah IDE.

Ucapan Terimakasih

Ucapan terimakasih kami ucapkan kepada Universitas 'Aisyiyah Surakarta melalui P3M karena telah memberikan dana untuk dilakukannya penelitian ini.

Referensi

- [1] Q. I. Sarhan, A. Szatmári, R. Tóth, and A. Beszédes, “CharmFL: A Fault Localization Tool for Python,” in *2021 IEEE 21st International Working Conference on Source Code Analysis and Manipulation (SCAM)*, 2021, pp. 114–119.
- [2] A. L. S. Saabith, T. Vinothraj, and M. M. M. Fareez, “A Review on Python Libraries and Ides for Data Science.”
- [3] P. Joshi and D. Bein, “Audible code, a voice-enabled programming extension of visual studio code,” in *17th International Conference on Information Technology—New Generations (ITNG 2020)*, 2020, pp. 335–341.
- [4] C. H. Chen and P. J. Guo, “Improv: Teaching programming at scale via live coding,” in *Proceedings of the Sixth (2019) ACM Conference on Learning@ Scale*, 2019, pp. 1–10.
- [5] M. Petrelli, “Python Essentials for a Geologist,” in *Introduction to Python in Earth Science Data Analysis*, Springer, 2021, pp. 11–23.
- [6] H. Fajar, S. El Yamin, and M. Siregar, “Desain Komunikasi Data Digital Pada Radio HF Dengan Metode Frequency Shift Keying Berbasis Thonny Python,” *JTIM J. Teknol. Inf. dan Multimed.*, vol. 1, no. 3, pp. 181–187, 2019.
- [7] K. M. Abdusaidovich, “Create a Great Project Using Sublime Text Software,” in “*ONLINE-CONFERENCES*” PLATFORM, 2021, pp. 253–262.
- [8] S. H. Tauqueer, “Flexible Visualization for Data Analytics on the Virtual Mobility World (ViM) Experimentation Platform.”
- [9] A. Corallo, L. Nassi, and N. Manca, “Bringing GNU Emacs to Native Code,” *arXiv Prepr. arXiv2004.02504*, 2020.
- [10] N. Trachsler, “WebTigerJython-a browser-based programming IDE for education.” ETH Zurich, 2018.
- [11] M. D. da Silva, “Aplicação da Ferramenta Google Colaboratory para o Ensino da Linguagem Python,” in *Anais da IV Escola Regional de Engenharia de Software*, 2020, pp. 67–76.
- [12] A. Chapagain, *Hands-On Web Scraping with Python: Perform advanced scraping operations using various Python libraries and tools such as Selenium, Regex, and others*. Packt Publishing Ltd, 2019.
- [13] L. Rai, *Programming in C++: Object Oriented Features*, vol. 5. Walter de Gruyter GmbH & Co KG, 2019.
- [14] S. T. O’Neil, J. A. Vucetich, D. E. Beyer, S. R. Hoy, and J. K. Bump, “Code, data, and metadata for the manuscript: Territoriality drives preemptive habitat selection in recovering wolves: implications for carnivore conservation,” 2020.
- [15] M. Kuroki, “Using Python and Google Colab to teach undergraduate microeconomic theory,” *Int. Rev. Econ. Educ.*, vol. 38, p. 100225, 2021.
- [16] R. Taylor and M. Smith, “Incorporating Online Simulated Environments into an MIS Curriculum to Accommodate for Disruptions Caused by the COVID-19 Pandemic,” *Southwest. Bus. Adm. J.*, vol. 19, no. 1, p. 5, 2021.
- [17] P. Xiao, *Practical Java Programming for IoT, AI, and Blockchain*. John Wiley & Sons, 2019.
- [18] W. Pipicano Chicangana, “Aplicación de la potencialidad de integración de un DBMS con Python y una herramienta SIG para la selección de la muestra en el estudio de encuestas origen y destino domiciliaria,” 2021.