****

**ALGORITMA DAN STRUKTUR DATA**

INF1083

**LAPORAN PRAKTIKUM 3 : Penjabaran Kelas**

Oleh :

*Akhmad Qasim*

*2211102441237*

Teknik Informatika

Fakultas Sains & Teknologi

Universitas Muhammadiyah Kalimantan Timur

Samarinda, 2023

**Laporan Praktikum 3:**

**Penjabaran Kelas**

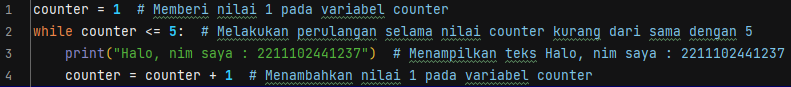
**Pokok Bahasan:**

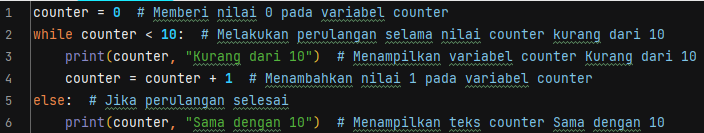
* Pengendali Struktur
* Pengendali Pengecualian
* Penjabaran Fungsi
* Penjabaran Kelas

**Tujuan Pembelajaran:**

* Pengendali Struktur.
* Pengendali Pengecualian.
* Penjabaran Fungsi.
* Penjabaran Kelas.

**Percobaan & Latihan:**

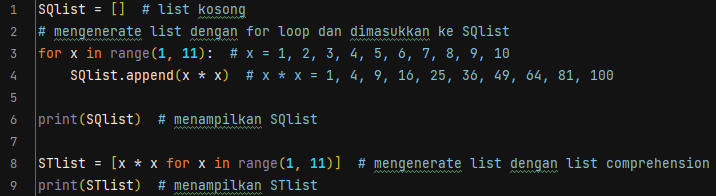
1. **Pengendali Struktur**
   1. **Berikan tampilan output dan keterangan penjelasan!**  
      Syntax:  
        
        
      Tampilan Output:  
      Text

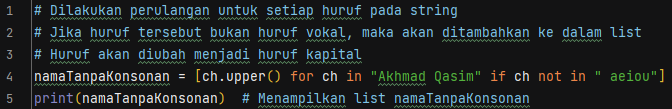
      Description automatically generated
2. **Pengendali Struktur**
   1. **Berikan tampilan output dan keterangan penjelasan!**  
      Syntax:  
        
        
      Tampilan Output:  
      Text

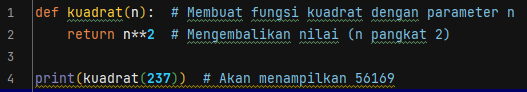
      Description automatically generated with medium confidence
   2. **Apa yang terjadi jika iterasi terjadi hingga <=10 ? (berikan output & penjelasan)**  
      Output:  
      Text

      Description automatically generated with medium confidence  
        
      Tidak terjadi error, tetapi perulangan akan di eksekusi sebanyak 12 kali karena perulangan kurang dari sama dengan 10 yang artinya pada saat berhendi di angka 11 masuk ke percabangan else yang memberikan output berupa sebuah pernyataan yang tidak sesuai dengan yang ditampilkan.
3. **Pengendali Struktur**
   1. **Berikan tampilan output dan keterangan penjelasan!**Syntax:  
      Text

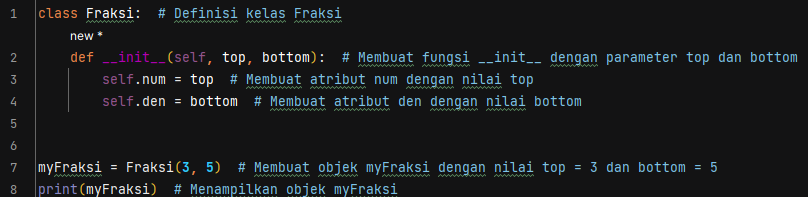
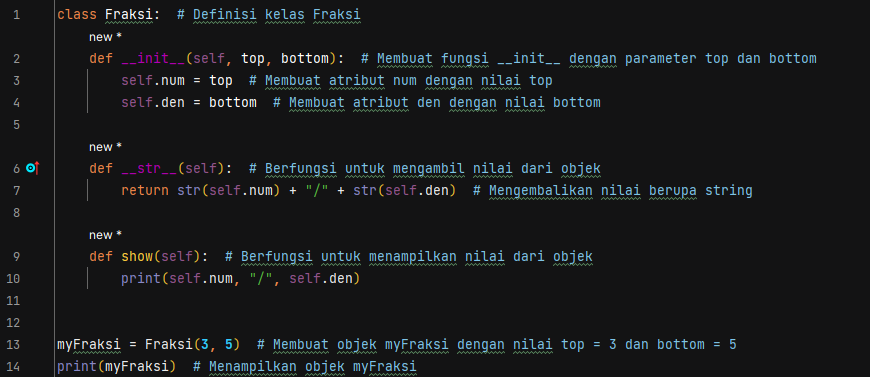
      Description automatically generated  
        
      Tampilan Output:  
      
4. **Pengendali Struktur**
   1. **Buatlah sebuah perintah program input untuk memasukkan nilai variabel n, kemudian jalankan perintah pada gambar diatas!**Syntax:  
      Text

      Description automatically generated
   2. **Berikan 2 nilai input variabel (n = -2) dan (n = 2)!**  
        
      
   3. Berikan tampilan output dan penjelasannya!  
        
      Jika angka -2 dimasukkan maka akan menampilkan pesan error karena angka yang di input negatif.  
        
        
      Jika angka 2 dimasukkan kedalam variable n, maka akan menampilkan hasil 1.4142135623730951 karena 2 adalah bilangan positif dan akar dari 2 adalah 1.4142135623730951.
5. **Pengendalian Struktur**
   1. **Berikan tampilan output!**  
      Syntax:  
        
        
      Tampilan Output:  
      
   2. **Jelaskan perbedaan kedua metode dalam pemberian sebuah nilai di variabel tersebut (SQlist dan STlist)!**  
      Perbedaan antara list comprehension dan for loop adalah:  
      1. List comprehension lebih singkat.  
      2. List comprehension lebih cepat.  
      3. List comprehension lebih mudah dibaca.
   3. **Buatlah variabel S, V, M dengan menggunakan metode List Comprehension sehingga membuat output seperti gambar dibawah ini:**  
      Text

      Description automatically generated
6. **Pengendali Struktur**
   1. **Berikan tampilan output dan penjelasannya!**Syntax:  
      ****  
        
      Tampilan Output:  
      
7. **Pengendali Pengecualian**
   1. **Jelaskan mengapa terjadi ValueError diperintah tersebut!**  
      Jika nilai yang di input -23 maka akan muncul pesan error karena nilai yang di input negatif.
   2. **Berikan pengendali pengecualian menggunakan metode try & except untuk mengatasi masalah diatas!**Text

      Description automatically generated
8. Penjabaran Fungsi
   1. **Berikan tampilan output dan penjelasannya!**  
      Syntax:  
        
        
      Tampilan Output:  
      
   2. **Buatlah fungsi-fungsi yang menyatakan menjabarkan fungsi berikut:**
      1. Luas lingkaran (n = jari-jari).  
         Text

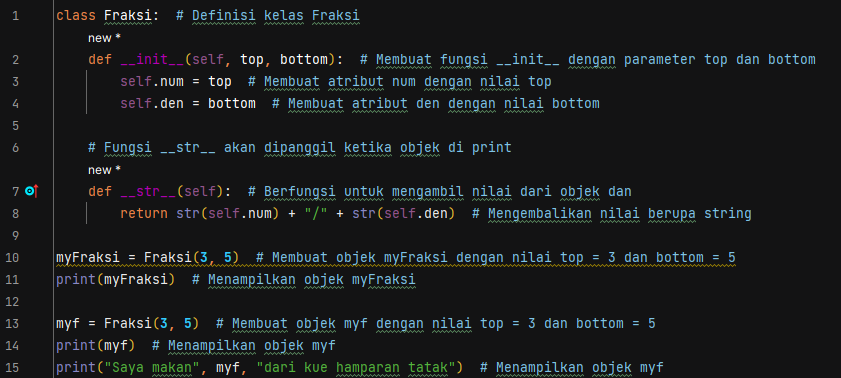
         Description automatically generated
      2. Luas segitiga siku-siku (a = alas, t = tinggi).  
         Text

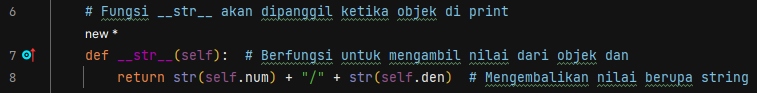
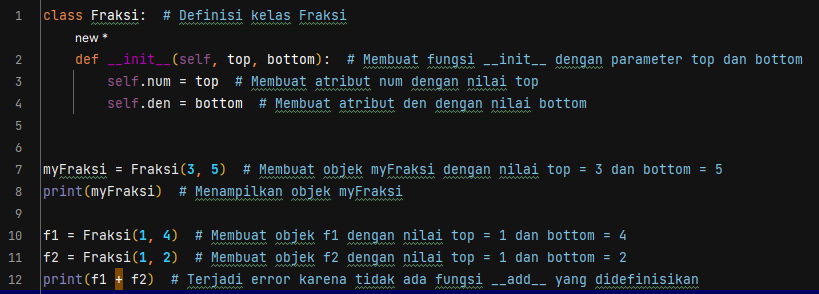
         Description automatically generated
9. **Penjabaran Kelas**
   1. **Berikan tampilan output dan penjelasan keterangan output tersebut!**Syntax:  
        
        
      Tampilan Output:  
      
   2. **Apa peran ‘\_\_init\_\_’ pada baris ketiga pada perintah diatas?**  
      Peran \_\_init\_\_ adalah untuk menginisialisasi objek yang baru dibuat
10. **Penjabaran Kelas**
    1. **Tambahkan fungsi diatas kedalam kelas Fraksi yang terdapat di Percobaan & Latihan 3.9!**Syntax:  
       
    2. **Bagaimana cara menampilkan output menggunakan fungsi diatas agar tampilan output menghasilkan “myFraksi = 3/5” ?**Syntax: **Text

       Description automatically generated**  
       Tampilan Output:Text, logo

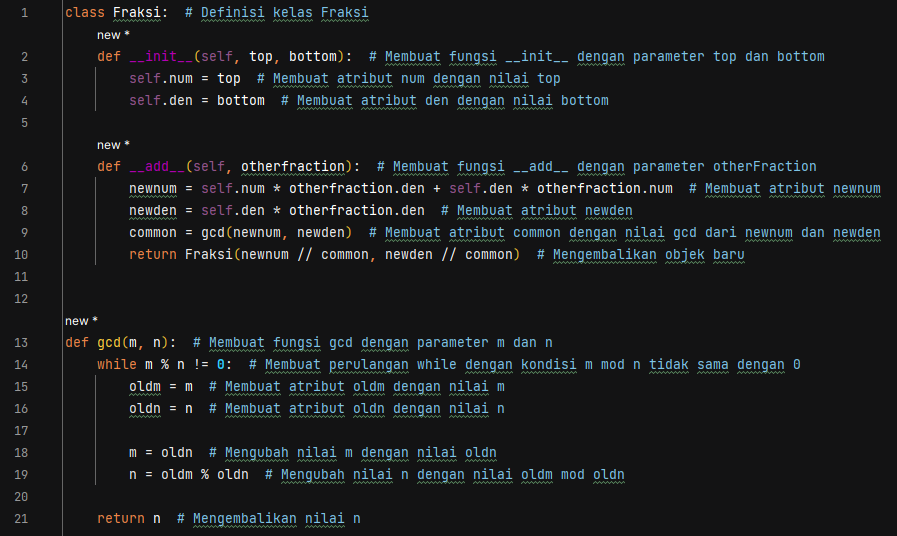
       Description automatically generated
11. **Penjabaran Kelas**
    1. **Tambahkan fungsi diatas kedalam kelas Fraksi yang terdapat di percobaan & Latihan 3.9!**Syntax:  
       A screenshot of a computer

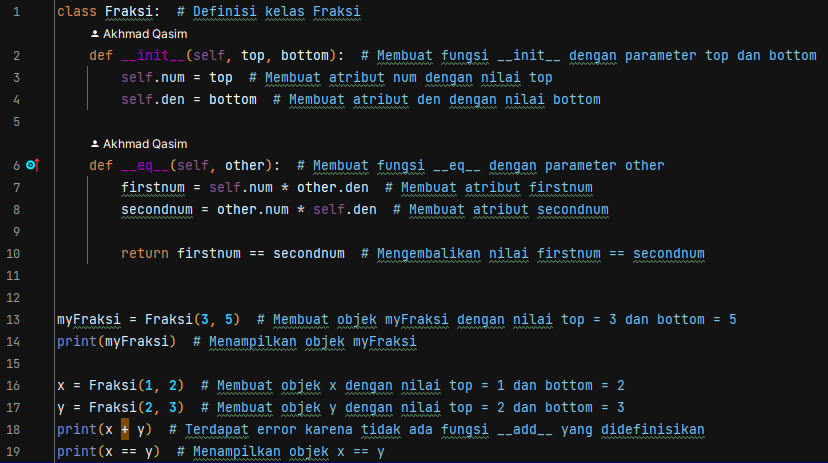
       Description automatically generated with medium confidence
    2. **Jalankan perintah dibawah ini:**A picture containing text

       Description automatically generated  
       Syntax:  
         
         
       Tampilan Output:  
       Text

       Description automatically generated
    3. **Apa peran ‘\_\_str\_\_’ yang terdapat di fungsi tersebut?**  
       
12. **Penjabaran Kelas**
    1. **Berikan tampilan output dan jelaskan mengapa terjadi Error?**  
       Syntax:  
         
         
       Tampilan Output:  
       A screenshot of a computer

       Description automatically generated with medium confidence
    2. **Tambahkan fungsi berikut ini kedalam kelas Fraksi yang terdapat di Percobaan & Latihan 3.9!**  
         
       Syntax:  
       Graphical user interface, text

       Description automatically generated
    3. Jalankan kembali perintah print diatas!  
       Tampilan Output:  
       
    4. **Apa peran ‘\_\_add\_\_’ yang terdapat di fungsi tersebut?**  
       Fungsi dari \_\_add\_\_ pada kode tersebut adalah untuk menambahkan dua objek Fraksi. Metode ini mendefinisikan perilaku operasi penjumlahan (+) pada objek Fraksi. Dalam hal ini, self merepresentasikan objek Fraksi yang saat ini dipanggil, sementara otherfraction merepresentasikan objek Fraksi lain yang digunakan sebagai argumen pada operasi penjumlahan.  
         
       Operasi penjumlahan antara dua objek Fraksi melibatkan penjumlahan numerik pembilang dan penyebut dari kedua objek tersebut. Fungsi \_\_add\_\_ melakukan operasi ini dengan menghitung numerik pembilang dan penyebut dari objek Fraksi baru, yang kemudian digunakan untuk membuat objek Fraksi baru yang merupakan hasil penjumlahan kedua objek Fraksi.  
         
       Pada akhirnya, fungsi \_\_add\_\_ mengembalikan objek Fraksi baru yang merupakan hasil penjumlahan dari dua objek Fraksi.
13. **Penjabaran Kelas**
    1. **Jalankan perintah berikut!**  
       f1 = Fraksi(1, 4)  
       f2 = Fraksi(1, 2)  
       print(f1 + f2) .  
         
       Syntax:  
         
       A picture containing text, screen, screenshot, flat

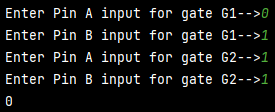
       Description automatically generated  
         
       Tampilan Output:  
       
    2. **Jelaskan mengapa perintah diatas mempunyai hasil yang berbeda dengan hasil pada Percobaan & Latihan 3.12c?**  
       Karena terdapat perbedaan antara fungsi \_\_add\_\_ yang ada didalam kelas Fraksi, dimana pada Percobaan & Latihan 3.13a terdapat tambahan syntax.
14. **Penjabaran Kelas**
    1. **Berikan tampilan output dan penjelasannya?**  
       Syntax:  
         
         
       Tampilan Output:  
       A screenshot of a computer

       Description automatically generated with medium confidence

**Kesimpulan:**

Kelas dalam Python merupakan sebuah blueprint atau template yang digunakan untuk membuat objek. Kelas mendefinisikan atribut dan metode yang akan dimiliki oleh objek yang dibuat. Berikut adalah penjelasan mengenai beberapa konsep yang terkait dengan kelas dalam Python:

1. Pengendali struktur: Pengendali struktur seperti if, while, dan list comprehension dapat digunakan dalam kelas untuk mengatur alur program dalam objek yang dibuat. Misalnya, kita bisa menggunakan if dalam metode kelas untuk menentukan apakah suatu objek memenuhi syarat tertentu atau tidak.
2. Pengendali pengecualian: Pengendali pengecualian seperti else, try-except dapat digunakan dalam kelas untuk menangani kesalahan atau kondisi yang tidak diinginkan dalam objek. Misalnya, kita bisa menggunakan try-except dalam metode kelas untuk menangani kesalahan saat memproses data dalam objek.
3. Metode: Kelas memiliki metode yang merupakan fungsi yang terkait dengan objek yang dibuat dari kelas tersebut. Dalam kelas, kita dapat mendefinisikan metode khusus seperti \_\_init\_\_ yang digunakan untuk menginisialisasi objek, atau metode kustom lainnya seperti fraction yang digunakan untuk melakukan operasi pada objek.
4. Atribut: Kelas juga memiliki atribut, yaitu variabel yang terkait dengan objek yang dibuat dari kelas tersebut. Atribut dapat digunakan untuk menyimpan informasi tentang objek, seperti nama, usia, atau nilai-nilai lainnya.
5. Inheritance: Inheritance adalah konsep di mana kelas baru dibuat dengan cara menurunkan sifat dan perilaku dari kelas yang sudah ada. Kelas baru ini dapat menambahkan atribut atau metode tambahan, atau mengubah atau memperluas metode dan atribut yang ada pada kelas yang diturunkan.

Dengan menggunakan konsep-konsep ini, kita dapat membuat kelas yang kompleks dan fleksibel dalam Python. Kelas memungkinkan kita untuk membuat objek dengan perilaku dan sifat yang sesuai dengan kebutuhan kita, sehingga memudahkan pengembangan program yang lebih besar dan kompleks.  
  
**Tugas Mandiri:**  
  


Tugas mandiri menjelaskan pengantar konsep pemrograman berorientasi objek dengan Python, khususnya menggunakan teknik inheritance (pewarisan) untuk membuat kelas dalam gerbang logika. Kelas LogicGate adalah kelas utama yang mendefinisikan fungsionalitas dasar gerbang, dan BinaryGate dan UnaryGate adalah subkelas yang menambahkan pin input ke gerbang. Kelas AndGate, OrGate, dan NotGate adalah subkelas selanjutnya yang menentukan perilaku spesifik setiap gerbang, menggunakan metode performGateLogic untuk menghitung output gerbang. Penggunaan pewarisan memungkinkan programmer menggunakan kembali kode dan ekstensi agar menjadi lebih mudah berdasarkan dari hierarki kelas untuk memasukkan jenis gerbang baru.