

Laporan Praktikum
Pemrograman Berorientasi Objek



Disusun oleh:
Harun Abdulkarim Khafid – 121140147

PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI PRODUKSI DAN INDUSTRI
INSTITUT TEKNOLOGI SUMATERA

2023

1. Apa itu interface dan Kapan kita perlu memakainya?

Interface berarti mengacu pada desain pada sebuah pemrograman, dalam python interface digunakan untuk menggambarkan setiap kelas yang ada di dalam pemrograman. Lebih jelasnya interface akan mendefinisikan perilaku apa yang akan terjadi di dalam kelas tersebut tetapi tidak menjelaskan secara detail isi dari kelas tersebut [1]

Pendefinisian pada python tidak ada secara eksplisit dalam Bahasa biasanya lebih terfokuskan pada polymorfisme dan duck typing, tapi dalam beberapa kondisi duck typing sering memberikan kesan yang membingungkan maka dibuatlah formal interface yaitu abstract method

Interface dipakai pada situasi tertentu yaitu pada saat penggunaan abstraksi, polimorfisme. Sehingga kode dapat terlihat lebih bersih dan mudah dipelihara, dan saat pengujian dan simulasi Pada python ada beberapa framework yang digunakan untuk membuat graphical user interface, diantaranya adalah

- Tkinter
- PyQt
- PySide
- Kivy
- WxPython
- PyGTK
- Pygame

2. Apa itu kelas abstrak dan kapan kita perlu memakainya? Apa perbedaannya dengan interface?

Kelas abstrak merupakan sebuah kelas yang berisi detail penting pada program pada kelas tertentu yang nantinya Ketika ditampilkan hanya memunculkan bagian yang pentingnya saja. Bisa dibilang kelas abstrak merupakan kelas yang melakukan perintah secara formal, karena dalam beberapa kondisi penggunaan protocol atau duck typing terkadang bisa membingungkan sehingga solusi dari hal tersebut dengan menggunakan formal interface. Pada formal interface setidaknya dibutuhkan sebuah abstract base class atau ABC cara penggunaannya pada python yaitu dengan mengimpor modul ABC lalu pendefinisian sebuah kelas sebagai sebuah abstrak yang nantinya semua kelas turunan dari kelas abstrak ini harus mengimplementasikan semua method yang ada di kelas abstrak.

```

1  from abc import ABC, abstractmethod
2
3  class abstrak(ABC):
4
5      @abstractmethod
6      def jalan(self):
7          pass
8
9      @abstractmethod
10     def serang(self, parameter):
11         pass
12
13     class manusia(abstrak):
14
15         def jalan(self):
16             print("anda berjalan")
17             # implementasi metode
18
19         def serang(self, parameter):
20             print("anda menyerang")
21             # implementasi metode
22
23     harun = manusia()
24     harun.jalan()
25     harun.serang()

```

Jadi perbedaan antara kelas abstrak dan interface adalah abstrak merupakan sebuah kelas yang berisi method yang nantinya akan digambarkan pada interface tanpa memunculkan detail penting yang ada pada abstrak

3. Apa itu kelas konkret dan kapan kita perlu memakainya?

Ketika ingin membuat sebuah objek yang atribut dan perilaku nya bisa diubah sesuai dengan keinginan, maka kelas konkret diperlukan untuk memenuhi kebutuhan aplikasi tersebut. Berbeda dengan abstrak kelas konkret tidak memiliki method yang harus diimplementasikan oleh kelas turunannya. Pada kelas konkret terdapat implementasi lengkap mengenai program, seperti metode, property, variable dan perilaku khusus sesuai dengan kebutuhan

```

1  class Mobil:
2      def __init__(self, merk):
3          self.merk = merk
4          self.bensin = 100
5
6      def jalan(self):
7
8          if(self.bensin>0):
9              self.bensin--50
10             print("mobil berjalan")
11         else:
12             print("bensin habis")
13
14     def isi_bensin(self):
15         self.bensin = 100
16         print("bensin terisi full")
17
18     # Membuat objek dari kelas konkret
19     mobil = Mobil("Toyota")
20     mobil.jalan() # Output: Mobil berjalan
21     mobil.jalan() # Output: Mobil berjalan
22     mobil.jalan() # Output: Bensin Habis
23     mobil.isi_bensin() # Output: Bensin terisi full
24

```

**4. (Opsional, nilai tambah jika benar) Apa itu metaclass dan kapan kita perlu memakainya?
Apa bedanya dengan inheritance biasa?**

Metaclass merupakan kelas yang mengatur atau memanipulasi kelas lain, secara implisit dikaitkan dengan meta class bawaan yaitu 'type'

Meta class dapat digunakan untuk memodifikasi pembuatan kelas, validasi, mengatur inheritance, metaprogramming.

Inheritance membuat beberapa kode bisa dipakai Kembali di kelas turunannya [2]

Bibliography

- [1] Genius Education, "CARA MEMBUAT INTERFACE PYTHON: PYSIMPLEGUI DAN BEBERAPA FITUR LAIN," 26 April 2022. [Online]. Available: <https://www.genius.education/blog/cara-membuat-interface-python-pysimplegui-dan-beberapa-fitur-lain>. [Accessed 6 June 2023].
- [2] W. Murphy, "implementing an interface in python," realpython, 2020. [Online]. Available: <https://realpython.com/python-interface/>. [Accessed 2 June 2023].