|  |  |  |
| --- | --- | --- |
| **Nama:**  **(Isi Nama Anda)**  **NIM: (Isi NIM Anda)** | C:\Users\RPL-SI 02\Pictures\288px-Trisakti_Logo.svg.png | **MODUL 11**  **Nama Dosen:**  **Anung B. Ariwibowo, M. Kom** |
| **Hari/Tanggal:**  **Hari, Tanggal Bulan 2022** | **Praktikum Algoritma & Pemrograman** | **Nama Asisten Labratorium:**   1. **Azhar Rizki Zulma 065001900001** |

**Object Oriented Programming pada Python**

1. **Teori Singkat**

Object Oriented Programming atau Pemrograman Berorientasi Objek merupakan paradigma pemrograman berdasarkan konsep "objek", yang dapat berisi data, dalam bentuk field atau dikenal juga sebagai atribut serta kode, dalam bentuk fungsi/prosedur atau dikenal juga sebagai method. Python telah menjadi bahasa berorientasi objek sejak bahasa Python sendiri dibuat.

**Class**

Class adalah prototype, atau blueprint, atau rancangan yang mendefinisikan variable dan method-method pada seluruh objek tertentu. Class berfungsi untuk menampung isi dari program yang akan di jalankan, di dalamnya berisi atribut / type data dan method untuk menjalankan suatu program. Dalam Python sendiri class didefinisikan dengan keyword class dan diikuti oleh penamaan kelas tersebut “*class nama\_kelas”*. Pemanggila kelas sendiri sama seperti pemanggilan sebuah fungsi/method dalam sebuah program yaitu memanggil nama class tersebut beserta parameter classnya. Biasanya class berisi banyak method/fungsi yang merupakan turunan sifat dari kelas tersebut.

Class sendiri memiliki banyak bentuk dalam setiap Bahasa pemrograman yang berbeda, seperti abstrak class, data class dan lain sebagainya. Class juga dapat memiliki keterkaitan dengan class lainnya yang dapat disebut sebagai class turunan atau inheritance, Inheritance merupakan sebuah hubungan Parent Class (Kelas Induk) dengan Child Class (Kelas Anak) yang dimana memiliki pewarisan sifat dan pewarisan variabel turunan yang sama.

Contoh Program OOP sederhana pada Python

|  |
| --- |
|  |

Output

|  |
| --- |
|  |

1. **Alat dan Bahan**

Hardware : Laptop/PC

Software : Spyder (Anaconda Python)

1. **Elemen Kompetensi**
   1. Latihan pertama

Buatlah sebuah program yang mengimplementasikan sebuah class yang memiliki nama class mahasiswa dan memiliki method yang dapat digunakan untuk menampilkan biodata mahasiswa yang diinputkan oleh user.

Source Code

|  |
| --- |
| print(“Isi dengan Source Code kalian ya…”) |

Output

|  |
| --- |
| print(“Screenshot hasil keluaran atau output dari source code kalian ya…”) |

* 1. Latihan Kedua

Buatlah sebuah kelas yang menerapkan method getter dan setter dimana menggunakan implementasi program percabangan serta perulangan seperti pada Latihan sebelumnya. Program menerima deklarasi nilai inputan dari user dan menampungnya dalam sebuah kelas dan variabel didalam kelas tersebut dapat dimanipulasi serta dirubah sesuai keinginan dan perubahan yang diberikan oleh user melalui inputan user itu sendiri serta dapat ditampilkan menggunakan method getter dan setter.

Source Code

|  |
| --- |
| print(“Isi dengan Source Code kalian ya…”) |

Output

|  |
| --- |
| print(“Screenshot hasil keluaran atau output dari source code kalian ya…”) |

1. **File Praktikum**

Github Repository:

|  |
| --- |
| print(“Taruh link github repository praktikum kalian disini yaa…”) |

1. **Soal Latihan**

Soal:

1. Jelaskan apa itu method getter dan setter pada sebuah class dan apa kegunaan serta fungsi method getter dan setter dalam sebuah class?
2. Deskripsikan serta narasikan jalannya alur source code program yang sebelumnya telah kalian buat pada Elemen Kompetensi Latihan Kedua!

Jawaban:

1. (Isi Dengan Jawaban Kalian)
2. (Isi Dengan Jawaban Kalian)
3. **Kesimpulan**
   1. Dalam pengerjaan program dengan bahasa pemrograman Python, kita harus benar-benar teliti dalam menginputkan suatu fungsi untuk menampilkan suatu keluaran pada layar dengan sesuai.
   2. Kita dapat mengetahui… (Tolong Isi lebih dari dua baris!)
4. **Cek List (✓)**

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Elemen Kompetensi** | **Penyelesaian** | |
| **Selesai** | **Tidak Selesai** |
| **1.** | Latihan Pertama |  |  |
| **2.** | Latihan Kedua |  |  |

1. **Formulir Umpan Balik**

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Elemen Kompetensi** | **Waktu Pengerjaan** | **Kriteria** |
| **1.** | Latihan Pertama | … Menit | … |
| **2.** | Latihan Kedua | … Menit | … |

Keterangan:

1. Menarik
2. Baik
3. Cukup
4. Kurang