

Nama : Muhamad Daffa Maulana Arrasyi
Kelas : TI.22.A4
NIM : 312210335

Project - UAS Semester 1

Link Github : <https://github.com/daffamaul/projek-uas-semester-1>
Link Youtube : https://youtu.be/huxfXYDo_Jk

1 Penjelasan

Projek ini membuat sebuah daftar mahasiswa dan daftar nilainya dengan menggunakan konsep CRUD (Create, Update, Read, Delete) dan menggunakan bahasa pemrograman Python yang dimana tugas-tugas tersebut dapat dibungkus menjadi sebuah Package (Folder) yang didalamnya terdapat Modul (File). Konsep Package dan Modul memudahkan untuk mengorganisir sebuah program.

1.1 Tampilan Utama Pada Program

Dibawah ini adalah sebuah coding yang merepresentasikan tampilan awal sebuah program ketika user membuka sebuah program. User diharapkan untuk memilih sebuah menu yang telah disediakan pada program, menu tersebut menggunakan program perulangan `_while True_` yang artinya, menu tersebut tidak akan pernah berhenti terkecuali user menginputkan kata kunci "k", maka program akan berhenti. `_Syntax_ `data = daftar_nilai.Data_mahasiswa()`` yang artinya bahwa variabel data tersebut adalah sebuah objek yang ada pada file daftar nilai yang di dalamnya memiliki sebuah `_Class_` yang bernama `_Data_mahasiswa_`

```
```python
import os
from view import input_nilai, view_nilai
from model import daftar_nilai

data = daftar_nilai.Data_mahasiswa()

print("="*20)
print("|PROGRAM INPUT DATA|")
print("="*20)

while True:
 print()
 menu = input("[(T)ambah, (I)nput Nilai, (L)ihat, (C)ari, (H)apus, (U)bah, (K)eluar] : ")
 print("~"*78)
 print()

 if menu.lower() == 't':
 data.tambah()

 elif menu.lower() == 'i':
 input_nilai.nilai()

 elif menu.lower() == 'l':
 if data.nama:
```

```

 view_nilai.lihat()
 else:
 print("BELUM ADA DATA!, pilih [T/t] untuk menambah data")

elif menu.lower() == 'c':
 if data.nama:
 data.cari()
 else:
 print("BELUM ADA DATA!, pilih [T/t] untuk menambah data")

elif menu.lower() == "h":
 data.hapus(data.nama)

elif menu.lower() == "u":
 data.ubah(data.nama)

elif menu.lower() == "k":
 print("Program selesai, Terima Kasih :) ")
 os.system("cls")
 break

else:
 print("\n INPUT {} TIDAK ADA!, Silakan pilih [T/L/I/H/U/K] untuk
menjalankan program!".format(menu))
'''

```

### 1.1.1 Penjelasan Mengenai From dan Import

Syntax dibawah ini saya gunakan untuk meng-clear tumpukan tampilan sebelumnya agar tampilan enak dilihat, pada program tampilan utama ini terdapat syntax `os.system("clr")` yang dapat membersihkan tampilan layar CMD.

```

'''python
import os
'''

```

Syntax dibawah ini menjelaskan bahwa dari folder view masukkan program modul `input_nilai` dan `view_nilai` ke dalam tampilan utama, maksudnya walaupun file `input_nilai` dan `view_nilai` berbeda folder dengan file `main.py`, kedua file tersebut bisa di akses pada file `main.py`

```

'''python
from view import input_nilai, view_nilai
'''

```

Syntax dibawah ini menjelaskan sama seperti penjelasan diatas yang artinya dari folder yang bernama `model`, masukkan file yang bernama `daftar_nilai` ke dalam file yang bernama `main`

```

'''python
from model import daftar_nilai
'''

```

## 1.2 Ketika User Memilih Menu

[(T)ambah, (I)npot Nilai, (L)ihat, (C)ari, (H)apus, (U)bah, (K)eluar] :  
 Jika user tidak memilih menu apapun, maka akan tampil pesan pilihan tidak tersedia

```

```python
else:
    print("\n INPUT {} TIDAK ADA!, Silakan pilih [T/L/I/H/U/K] untuk
menjalankan program!".format(menu))
```

```

### 1.2.1 Tambah

Jika user menginputkan tambah data dengan menekan huruf "t" pada keyboard, maka user akan diarahkan pada pengisian data yang diperlukan yang ada pada folder model yang didalamnya ada daftar\_nilai.

```

```python
if menu.lower() == 't':
    data.tambah()
```

```

Kode dibawah ini merepresentasikan penginputan data oleh user, jika nama sudah ditambahkan, maka akan menampilkan pesan berhasil ditambahkan

```

```python
def tambah(self):
    os.system("cls")
    print("Tambah data\n")
    nama = input("Nama : ")
    self.nama.append(nama)
    nim = int(input("NIM : "))
    self.nim.append(nim)
    uts = 0
    self.uts.append(uts)
    uas = 0
    self.uas.append(uas)
    tugas = 0
    self.tugas.append(tugas)
    akhir = (tugas * .3) + (uts * .35) + (uas * .35)
    self.akhir.append(akhir)

    print("\nData {0} berhasil di tambahkan".format(nama))
```

```

### 1.2.2 Input Nilai

```

```python
elif menu.lower() == "i":
    input_nilai.nilai()
```

```

Jika user menginputkan tambah data dengan menekan huruf "i" pada keyboard, maka user akan diarahkan pada pengisian nama yang ada pada folder view yang didalamnya ada input\_nilai. Pada file tersebut, ada pengambilan sebuah file melalui sebuah folder yang ditunjukkan pada Syntax from model import daftar\_nilai yang artinya file daftar\_nilai dapat diakses di dalam file input\_nilai. Jika nama tersebut diinputkan user sebelumnya sudah ditambahkan, maka selanjutnya menginputkan data yang diperlukan. Jika tidak ada, maka belum menginputkan data tersebut.

```

```python
from model import daftar_nilai

```

```

data = daftar_nilai.Data_mahasiswa()

def nilai():
    input_nama = input("Masukan Nama : ")
    if input_nama in data.nama:
        index = data.nama.index(input_nama)
        data.tugas[index] = int(input("Nilai Tugas : "))
        data.uts[index] = int(input("Nilai UTS : "))
        data.uas[index] = int(input("Nilai UAS : "))
        data.akhir[index] = data.tugas[index] * .3 +
        data.uts[index] * .35 + data.uas[index] * .35

        print("\nData nilai berhasil di input!")
    else:
        print("NAMA {0} TIDAK ADA!".format(input_nama))
    ...

```

1.2.3 Lihat

Jika user menginputkan tambah data dengan menekan huruf "l" pada keyboard, maka user akan ditunjukkan sebuah data yang sudah diinputkan berdasarkan nama. Jika belum, maka menampilkan pesan data belum ditambahkan

```

```python
elif menu.lower() == 'l':
 if data.nama:
 view_nilai.lihat()
 else:
 print("BELUM ADA DATA!, pilih [T/t] untuk menambah
data")
```

```

Syntax from model import daftar_nilai yang artinya file daftar_nilai dapat diakses di dalam file view_nilai. Jika data telah ditambahkan oleh user sebelumnya, maka akan menampilkan data sesuai inputan.

```

```python
import os
from model import daftar_nilai

data = daftar_nilai.Data_mahasiswa()

Menampilkan seluruh data
def lihat():
 os.system("cls")
 for i in range(len(data.nama)):
 print(f"\nData ke -{i+1}")
 print(f>Nama Mahasiswa: {data.nama[i]}")
 print(f"NIM Mahasiswa : {data.nim[i]}")
 print(f"Nilai UTS : {data.uts[i]}")
 print(f"Nilai UAS : {data.uas[i]}")
 print(f"Nilai TUGAS : {data.tugas[i]}")
 print(f"Nilai Akhir : {data.akhir[i]}")

```

```
...
```

#### 1.2.4 Cari

Jika user menginputkan tambah data dengan menekan huruf "c" pada keyboard, maka user akan ditunjukkan sebuah data yang sudah diinputkan berdasarkan nama. Jika nama tersebut ada pada data, maka menampilkan data-data tersebut. Jika nama tidak ada pada data, maka menampilkan pesan tidak ada data

```
```python
elif menu.lower() == 'c':
    if data.nama:
        data.cari()
    else:
        print("BELUM ADA DATA!, pilih [T/t] untuk menambah
data")
...
```pytohn
def cari(self):
 os.system("cls")
 print("Mencari data")
 print("="*15)
 nama = (input("\nMasukan Nama yg ingin di cari : "))
 if nama in self.nama:
 print("Nama tersedia, berikut datanya")
 index = self.nama.index(nama)
 print(f>Nama Mahasiswa: {self.nama[index]}")
 print(f"NIM Mahasiswa : {self.nim[index]}")
 print(f"Nilai UTS : {self.uts[index]}")
 print(f"Nilai UAS : {self.uas[index]}")
 print(f"Nilai TUGAS : {self.tugas[index]}")
 else:
 os.system("cls")
 print("NAMA {0} TIDAK ADA!".format(nama))
...

```

#### 1.2.5 Hapus

```
```python
elif menu.lower() == "h":
    data.hapus(data.nama)
...

```

Jika user menginputkan tambah data dengan menekan huruf "h" pada keyboard, maka user akan menginputkan data berdasarkan nama. Jika nama tersebut ada pada data, maka menghapus data-data tersebut berdasarkan index. Jika nama tidak ada pada data, maka menampilkan pesan tidak ada data

```
```python
def hapus(self, nama):
 os.system("cls")
 print("Hapus data inputan")
 nama = (input("\nMasukan Namal : "))
 if nama in self.nama:
 print("Data {0} berhasil di hapus".format(nama))
 index = self.nama.index(nama)
 del self.nama[index]
 del self.nim[index]

```

```

 del self.uts[index]
 del self.uas[index]
 del self.tugas[index]
 del self.akhir[index]
 else:
 os.system("cls")
 print("NAMA {0} TIDAK ADA!".format(nama))
 ...

```

### 1.2.6 Ubah

```

```python
elif menu.lower() == "u":
    data.ubah(data.nama)
...

```

Jika user menginputkan tambah data dengan menekan huruf "u" pada keyboard, maka user akan menginputkan data berdasarkan nama. Jika nama tersebut ada pada data, maka menampilkan inputan data-data yg ingin diubah berdasarkan index. Jika nama tidak ada pada data, maka menampilkan pesan tidak ada data

```

```python
def ubah(self, nama):
 os.system("cls")
 input_nama = input("Nama yang ingin di ubah : ")
 if input_nama in nama:
 print(f>Nama {input_nama} tersedia, silahkan ubah.")
 index = nama.index(input_nama)
 self.nim[index] = int(input("NIM : "))
 self.tugas[index] = int(input("Nilai TUGas : "))
 self.uts[index] = int(input("Nilai UTS : "))
 self.uas[index] = int(input("Nilai UAS : "))
 self.akhir[index] = (self.tugas[index] * .3) +
 (self.uts[index] * .35) + (self.uas[index] * .35)

 print("\nData {0} berhasil di
ubah".format(input_nama))
 else:
 os.system("cls")
 print("NAMA {0} TIDAK ADA!".format(nama))
...

```

### 1.2.7 Keluar

```

```python
elif menu.lower() == "k":
    print("Program selesai, Terima Kasih :) ")
    os.system("cls")
    break
...

```

Jika user menginputkan tambah data dengan menekan huruf "k" pada keyboard, maka user akan dikeluarkan dari program.