**CLO 4: Mampu mendesain, menulis, dan melakukan testing dalam rangka membangun projek sederhana menggunakan Python**

**Tugas Membuat LMS Sederhana**

**Informasi umum:**

1. Setiap mahasiswa hanya mengerjakan 1 buah fungsi yang telah ditentukan
2. Mahasiswa dapat mencoba demo contoh program yang diberikan untuk lebih memahami detil cara kerja tiap fungsi
3. Fungsi pada contoh program dibuat dengan kelengkapan minimal, sehingga sangat memungkinkan untuk dikembangkan oleh tiap mahasiswa, misalnya dari segi penanganan Error, estetika tampilan, atau pun penambahan fitur pada fungsi.
4. Mahasiswa **dilarang** mengubah struktur data, nama fungsi, dan parameter fungsi yang telah diberikan.

**Aturan pengerjaan:**

1. Ini adalah tugas Individu, dilarang bekerjasama. Jika ada pertanyaan, dapat ditanyakan ke dosen/asisten melalui channel slack: **#tubes1-lms-sederhana**
2. Setiap mahasiswa wajib mengerjakan fungsi yang ditentukan oleh dosen di bagian terakhir file ini.
3. Mahasiswa pertama-tama mendownload template kode program di: [**https://bit.ly/lms-sederhana**](https://bit.ly/lms-sederhana)
4. Mahasiswa mengisi jawaban dengan mengacu pada cara kerja fungsi seperti di demo contoh program.
5. Mahasiswa **wajib menghandle** segala Error yang mungkin timbul di fungsinya
6. Untuk pengetesan fungsi, cukup menjalankan file yang dikerjakan saja, tidak harus menjalankan lms.py. Tambahkan pemanggilan fungsi yang ingin di test di dalam bagian **if \_\_name\_\_ == "\_\_main\_\_":** seperti contoh pada file lms.py
7. Jawaban yang disubmit berupa 1 buah File .py yang fungsinya telah diisi, disubmit melalui LMS. Contoh: Jika mendapat bagian function nomor 1, maka yang dikirmkan adalah file mahasiswa.py yang telah diisi jawabannya.
8. Waktu pengerjaan sampai **12 Desember 21.00 Waktu LMS**

**Struktur data:**

Berikut ini dijelaskan struktur data yang digunakan pada program LMS Sederhana. Mahasiswa **dilarang** mengubah struktur data yang telah diberikan untuk menghindari Error yang timbul saat fungsi yang dibuat oleh tiap mahasiswa disatukan kembali ke program utuh.

1. dict\_mhs

dict\_mhs adalah dictionary untuk menyimpan data pribadi mahasiswa.

Terdapat 2 key, yaitu:

* ‘field’ memiliki value berupa list yang berisi tuple pasangan Label dan Pola Regex. Label adalah jenis data mahasiswa yang akan disimpan (seperti kolom), kemudian Pola Regex digunakan untuk memvalidasi data yang diinputkan untuk tiap field tersebut.
* ‘data’ menyimpan dictionary yang berisi nim sebagai key dan dictionary detil data mahasiswa sebagai value

Berikut adalah dict\_mhs default berisi data dummy:

|  |
| --- |
| dict\_mhs = {'field' : [('Nama', "^([a-zA-Z]+([ '-]| ')?[a-zA-Z]+){1,4}$"),  ('Email', '^([a-z0-9]+[.\_]?[a-z0-9]+)+[@]\w+[.]\w{2,3}'),  ('Password', '^[a-zA-Z0-9]{8,12}$')],  'data' : {'113': {'Nama': 'Dummy', 'Email': 'dummy@telU.com', 'Password': '12345678'},  '114': {'Nama': 'Joni', 'Email': 'joni@telU.com', 'Password': '12345678'},  '115': {'Nama': 'jeje', 'Email': 'jeje@telU.com', 'Password': '12345678'}  }  } |

1. list\_topic

list\_topic adalah sebuah list yang elemennya berupa dictionary berisi informasi detail tiap topik.

key ‘Activities’ pada dictionary detil topik menyimpan id\_activity, sementara informasi lengkap tentang activity tersebut ada di dict\_activity.

Berikut adalah list\_topic default berisi data dummy:

|  |
| --- |
| list\_topic = [{'Title': 'Dummy Topic 1', 'Description': 'Ini deskripsi topic 1', 'Activities':[0, 1]},  {'Title': 'Dummy Topic 2', 'Description': 'Ini deskripsi topic 2', 'Activities':[2]}  ] |

1. dict\_activity

dict\_activity adalah sebuah dictionary yang key nya adalah id\_activity, dan value nya berupa detil informasi activity tersebut. Terdapat dua Type activity, yaitu ‘assignment’ atau ‘material’.

Berikut adalah dict\_activity default yang berisi data dummy:

|  |
| --- |
| dict\_activity = {0: {'Title': 'Dummy Assignment 1', 'Type': 'assignment', 'Description': 'buatlah program Game'},  1: {'Title': 'Dummy material', 'Type': 'material', 'Description': 'slide minggu ini'},  2: {'Title': 'Dummy Assignment 2', 'Type': 'assignment', 'Description': 'buatlah program LMS'}  } |

1. dict\_submission

key pada dict\_submission adalah id\_activity, dan yang bisa disubmit jawabannya oleh mahasiswa hanyalah activity bertype ‘assignment’. Value pada dict\_assignment berupa dictionary yang key-nya adalah nim dan value adalah jawaban mahasiswa tersebut.

Berikut adalah dict\_submission default yang berisi data dummy:

|  |
| --- |
| dict\_submission = {0: {'113' : 'Ini jawaban mahasiswa 113',  '114': 'Ini jawaban mahasiswa 114'},  2: {'113' : 'Ini jawaban mahasiswa 113',  '114': 'Ini jawaban mahasiswa 114'}  } |

1. dict\_grade

key pada dict\_grade adalah id\_activity dan valuenya adalah dictionary yang berisi nim sebagai key dan nilai sebagai value.

Berikut adalah dict\_grade default yang berisi data dummy:

|  |
| --- |
| dict\_grade = {0: {'113' : 100}    } |

**List Function:**

Berikut adalah list function yang perlu diimplementasi oleh mahasiswa. Mahasiswa **dilarang** mengubah nama dan parameter fungsi.

|  |  |  |
| --- | --- | --- |
| No | Nama File/Module | Nama Function |
| 1 | mahasiswa.py | add\_one\_mhs |
| 2 | add\_many\_mhs |
| 3 | update\_mhs |
| 4 | delete\_mhs |
| 5 | login\_mhs |
| 6 | print\_to\_file |
| 7 | topic.py | show\_topic |
| 8 | add\_topic |
| 9 | update\_topic |
| 10 | delete\_topic |
| 11 | add\_activity |
| 12 | update\_activity |
| 13 | delete\_activity |
| 14 | print\_topic\_to\_file |
| 15 | submission.py | show\_assignment\_not\_submit |
| 16 | show\_nim\_not\_submit |
| 17 | submit\_assignment |
| 18 | update\_submission |
| 19 | delete\_submission |
| 20 | print\_submissions\_to\_file |
| 21 | grade.py | show\_mhs\_not\_graded |
| 22 | add\_grade\_by\_mhs |
| 23 | show\_assignment\_not\_graded |
| 24 | add\_grade\_by\_assignment |
| 25 | report\_by\_assignment |
| 26 | report\_by\_mhs |
| 27 | print\_grade\_to\_file |

**Penilaian:**

1. 81-100 : Program dapat running sama dengan contoh demo program atau dengan ada tambahan fitur atau perbaikan tampilan, tanpa ada Error
2. 66-80 : Program dapat running sama dengan contoh demo program, namun tidak menghandle Error yang mungkin timbul
3. 41-65 : Program dapat running tetapi tidak sesuai (kurang) dari apa yang ditampilkan pada contoh demo program
4. 0-40 : Program tidak dapat running dan kesesuaian kode di bawah 50% dari seharusnya.

**Pembagian Penugasan Fungsi:**

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **NIM** | **Nama** | **No Fungsi** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |