

UJIAN AKHIR SEMESTER
SEMESTER GANJIL 2023/2024

Mata Kuliah : Perograman Dasar
Program Studi : Informatika
Dosen : Diash Firdaus, M.T.
Kelas : AA-GG

Tanggal: 11 Januari 2024
Waktu : 11.00 – 12.30
Sifat : Open Note

A. SUBCPMK 4 Mahasiswa mampu bekerja sama dalam tim menjelaskan rancangan aplikasi dengan mengkomunikasikan hasil pengolahan data IoT (bobot kelulusan 15%)

1. Jelaskan Materi yang menurut anda paling berkesan serta alasannya ! (10 poin)
2. Bagaimana cara mendefinisikan sebuah API endpoint (membuat link get/post) menggunakan FastAPI? (10 point)
 - a. Dengan mendeklarasikan fungsi dan menambahkan decorator `@api_endpoint`
 - b. Dengan membuat file khusus yang berisi daftar endpoint
 - c. Dengan membuat kelas dan mengimplementasikan metode `handle_request`
 - d. Dengan mendeklarasikan fungsi dan menambahkan decorator `@app.get('/namalinknya')`
3. Bagaimana cara mengonsumsi API yang dibuat dengan FastAPI? (10 point)
 - a. Dengan menggunakan modul `requests` dalam Python
 - b. Dengan mengirimkan permintaan HTTP menggunakan browser
 - c. Dengan menginstall ekstensi browser untuk menguji API
 - d. Dengan menambahkan endpoint FastAPI sebagai bookmark pada browser
4. Bagaimana cara menjalankan aplikasi FastAPI? (10 point)
 - a. `python run app.py`
 - b. `uvicorn app:app -- host 127.0.0.1 --port 8080`
 - c. `fastapi uvicorn run app.py -- host 127.0.0.1 --port 8080 -- reload`
 - d. Hanya dapat dijalankan melalui server web eksternal

5. Jika terdapat FastApi pada gambar A, selanjutnya kita jalankan pada localhost dengan port 8080 dan terdapat source code serta hasil input data seperti pada gambar B. Ketika kita consume atau get API dengan alamat <http://127.0.0.1:8080/cobaapi2> maka output yang dihasilkan adalah (30 poin)

```
from fastapi import FastAPI

app = FastAPI()

cobain = ''

# terima
@app.get("/coba_kirim/{data}")
async def set_data(data):
    global cobain
    cobain = data
    return {"flag": cobain}

# kirim
@app.get("/cobaapi")
async def terima():
    return {"flag": cobain }

# kirim
@app.get("/cobaapi2")
async def terima():
    return {"flag": "Informatika Itenas" }
```

Gambar 1

```
import requests

def kirim_API(data):
    response1 = requests.get(f"http://localhost:8080/coba_kirim/{data}")
    if response1.status_code != 200:
        print(f"Gagal mengirim flag {data} ke server FastAPI")
    else :
        print("berhasil dikirim")

input_data = input("masukan data ")
kirim_API(input_data)
masukan data : Itenas Bandung
berhasil dikirim
```

Gambar 2

- a. flag : Informatika Itenas
- b. flag : Itenas Bandung
- c. Error
- d. A dan B ditampilkan

6. Terdapat text , pattern dan regex seperti pada code dibawah ini. Output yang dihasilkan jika program tersebut dijalankan adalah ? (15 poin)

```
1 import re
2
3 # Mencari dan mengekstraksi angka dari sebuah string
4 text = "Tahun 2022 dan 2023 adalah tahun yang baik 2025."
5 pattern = r"\d+"
6 match = re.findall(pattern, text)
7
8 if match:
9     print("Ditemukan:", match)
10 else:
11     print("Tidak ditemukan.")
```

- a. Ditemukan
- b. Ditemukan : 2022
- ☒ c. Ditemukan: ['2022', '2023', '2025']
- d. Semua jawaban salah

7. Terdapat text , pattern dan regex seperti pada code dibawah ini. Output yang dihasilkan jika program tersebut dijalankan adalah ? (15 poin)

```
1 import re
2
3 text = "informatika di ITENAS, ITENAS Bandung, itenas Juara"
4 pattern = r"ITENAS"
5 replacement = "institut Teknologi Nasional"
6 result = re.sub(pattern, replacement, text)
7
8 print("Hasil:", result)
```

- ☒ a. Hasil: informatika di institut Teknologi Nasional, institut Teknologi Nasional Bandung, itenas Juara
- b. Hasil: informatika di institut Teknologi Nasional, institut Teknologi Nasional Bandung, institut Teknologi Nasional Juara
- c. Hasil: informatika di ITENAS, ITENAS Bandung, itenas Juara
- d. Semua jawaban salah

B. SUBCPMK 3 Mampu membangun aplikasi yang merupakan solusi permasalahan menggunakan Bahasa pemrograman berbasis GUI (bobot kelulusan 15%)

1. Terdapat 3 orang dengan nama Diash, Firdaus dan Itenas, masing masing memiliki umur dan gaji seperti pada gambar dibawah. Dengan menggunakan function pada python maka Output dari code berikut adalah ... (30 poin)

```
1 import pandas as pd
2
3 data = {'Nama': ['Diash', 'Firdaus', 'Itenas'],
4         'Umur': [25, 20, 51],
5         'Gaji': [230000, 460000, 450000]}
6
7 df = pd.DataFrame(data)
8
9 def kategori_umur(umur):
10     if umur < 25:
11         return 'Muda'
12     elif umur >= 25 and umur < 35:
13         return 'Dewasa'
14     else:
15         return 'Matang'
16
17 df['Kategori_Umur'] = df['Umur'].apply(kategori_umur)
18
19 print(df)
```

2. Terdapat 3 orang dengan nama Ujang, Jajang, Ajang dan Sinjang, masing masing memiliki Usia, gaji dan status seperti pada gambar dibawah. Dengan memanfaatkan fungsi lamda maka Output dari code berikut adalah (30 poin)

```
1 import pandas as pd
2
3 data = {'Nama': ['Ujang', 'Jajang', 'Ajang', 'Sinjang'],
4         'Usia': [25, 35, 30, 28],
5         'Gaji': [50000, 60000, 70000, 55000],
6         'Status': ['Muda', 'Tua', 'Tua', 'Muda']}
7
8 df = pd.DataFrame(data)
9
10 df['Bonus'] = df.apply(lambda x: x['Gaji'] * 1.0 if x['Status'] == 'Tua' else x['Gaji'] * 0.5, axis=1)
11
12 print(df)
```


3. Diketahui terdapat dataframe seperti gambar dibawah, dengan menggunakan perulangan for, maka output yang didapatkan dari source code seperti pada gambar dibawah adalah ? (30 poin)



```
1  import pandas as pd
2
3  data = {'Nama': ['Diash', 'Firdaus', 'Itenas'],
4          'Usia': [25, 30, 35],
5          'Gaji': [50000, 60000, 70000]}
6
7  df = pd.DataFrame(data)
8
9  bonus = 0.5
10 for index, row in df.iterrows():
11     if row['Usia'] > 30:
12         df.at[index, 'Bonus'] = row['Gaji'] * bonus
13     else :
14         df.at[index, 'Bonus'] = 0
15
16 print (df)
```

4. Diketahui terdapat dataframe seperti gambar dibawah, dengan menggunakan lambda, maka output yang didapatkan dari source code seperti pada gambar dibawah adalah ? (10 poin)



```
1  import pandas as pd
2
3  data = {'Nama': ['Diash', 'Firdaus', 'Itenas'],
4          'Umur': [25, 20, 51],
5          'Gaji': [100000, 200000, 300000]}
6
7  df = pd.DataFrame(data)
8  df['Gaji_Bersih'] = df['Gaji'].apply(lambda x: 0.5 * x)
9
10 print(df)
```