
Petunjuk:

Cantumkan tanggapan layar (screenshot) untuk kode yang dituliskan dan hasil yang diperoleh sehingga saya bisa mengetahui bahwa Anda sendiri yang mengerjakan ujian ini.

Apabila ditemukan jawaban yang identik sama dari hasil screenshot-nya, maka hasil ujian dari setiap mahasiswa tersebut akan dibatalkan.

Letakkan file soal beserta jawaban ini pada platform kelas.usu.ac.id dalam format .pdf. Letakkan juga semua kode program berektensi .py hasil pengerjaan UTS ini pada akun GitHub masing-masing peserta ujian, lalu cantumkan link GitHub tersebut pada kelas.usu.ac.id.

1. Write a Python program that reads in a whole number and divides it by number of days this year and displays the result with eleven decimal places if they exist (rounded up).
2. Write a Python program that reads a number (today's test date) and prints the product of all the values from 1 to that number.
3. Write a Python program that reads in a number and prints the date that number of days from now in this format: Monday on 25 March 2024.
4. Write a Python class that calculates and stores the height and weight of a person in metric. The BMI is calculated using this formula:

$$\text{Weight/Height}^2$$

Weight is in pound and height is in feet.

The class should have two properties named: Weight and Height

The class should have two methods:

- BMI_Value – This takes no arguments and returns a decimal value of the BMI;
 - Equals – This should override the equals method from the object class to compare the weight and height of two BMI objects. To override the equal method you should implement this method: `__eq__(self, other)` and return a boolean.
5. Write a program that reads in integer numbers from a text file named input.txt in the same directory as the executing program.
Print the sum of the numbers with comma separators and three digits.

1. Screenshot Jawaban dari Soal ke-1.

The image displays two screenshots of a Visual Studio Code editor window, showing a Python script for calculating the number of days in a year based on leap year rules. The script is named `soal1.py` and is located in the `uts_nada` directory.

Top Screenshot: The script defines a function `hitung_pembagian(bilangan):` that calculates the number of days in a year based on leap year rules. The function uses `datetime.datetime.now().year` to get the current year and checks if it is a leap year. The function then calculates the number of days in the year and returns the result. The script also includes a `main()` function that prompts the user to enter a number and calls the `hitung_pembagian` function. The terminal output shows the execution of the script, displaying the result of the calculation.

```
8
9 def hitung_pembagian(bilangan):
10     # Mendapatkan jumlah hari dalam tahun ini
11     tahun_ini = datetime.datetime.now().year
12     if tahun_ini % 4 == 0 and (tahun_ini % 100 != 0 or tahun_ini % 400 == 0):
13         jumlah_hari = 366 # Tahun kabisat
14     else:
15         jumlah_hari = 365 # Bukan tahun kabisat
16
17     # Melakukan pembagian
18     hasil_pembagian = bilangan / jumlah_hari
19
20     # Menampilkan hasil dengan 11 angka di belakang koma (jika ada)
21     print("Hasil pembagian:", format(hasil_pembagian, ".11f"))
22
23 def main():
24     try:
25         bilangan = int(input("Masukkan bilangan bulat: "))
26         hitung_pembagian(bilangan)
27     except ValueError:
```

Bottom Screenshot: The script is modified to include error handling for invalid input. The `main()` function now includes a `try` block to catch `ValueError` exceptions and print a message if the input is not a valid integer. The terminal output shows the execution of the script, displaying the result of the calculation and the error handling message.

```
17     # Melakukan pembagian
18     hasil_pembagian = bilangan / jumlah_hari
19
20     # Menampilkan hasil dengan 11 angka di belakang koma (jika ada)
21     print("Hasil pembagian:", format(hasil_pembagian, ".11f"))
22
23 def main():
24     try:
25         bilangan = int(input("Masukkan bilangan bulat: "))
26         hitung_pembagian(bilangan)
27     except ValueError:
28         print("Input tidak valid. Masukkan bilangan bulat.")
29
30 if __name__ == "__main__":
31     main()
32
```

2. Soal 2

```
6
7 def hitung_perkalian(angka):
8     # Inisialisasi hasil
9     hasil = 1
10
11     # Perkalian semua nilai dari 1 hingga angka
12     for i in range(1, angka + 1):
13         hasil *= i
14
15     return hasil
16
17 def main():
18     try:
19         angka = int(input("Masukkan angka (tanggal tes hari ini): "))
20         if angka < 0:
21             print("Masukkan angka positif.")
22         else:
23             hasil_perkalian = hitung_perkalian(angka)
24             print("Hasil perkalian dari 1 hingga", angka, "adalah:", hasil_perkalian)
25     except ValueError:
26         print("Input tidak valid. Masukkan angka integer.")
27
28 if __name__ == "__main__":
29     main()
```

Masukkan angka (tanggal tes hari ini): 10
Hasil perkalian dari 1 hingga 10 adalah: 3628800
PS C:\pemrogintegrative\uts_nada>

3. Soal 3

```
7
8 from datetime import datetime, timedelta
9 import locale
10
11 # Atur locale untuk Bahasa Indonesia
12 locale.setlocale(locale.LC_TIME, 'id_ID.UTF-8')
13
14 def print_future_date(num_days):
15     # Dapatkan tanggal saat ini
16     current_date = datetime.now()
17
18     # Hitung tanggal di masa depan
19     future_date = current_date + timedelta(days=num_days)
20
21     # Format tanggal di masa depan
22     formatted_date = future_date.strftime("%A, %d %B %Y")
23
24     print("Tanggal {} hari dari sekarang akan menjadi: {}".format(num_days, formatted_date))
25
```

Hasil perkalian dari 1 hingga 10 adalah: 3628800
PS C:\pemrogintegrative\uts_nada> & C:/Users/Asus/AppData/Local/Programs/Python/Python38/python.exe c:/pemrogintegrative/uts_nada/soal3.py
Masukkan jumlah hari dari sekarang: 7
Tanggal 7 hari dari sekarang akan menjadi: Senin, 01 April 2024
PS C:\pemrogintegrative\uts_nada>

```
14 def print_future_date(num_days):
15     # Hitung tanggal di masa depan
16     future_date = current_date + timedelta(days=num_days)
17
18     # Format tanggal di masa depan
19     formatted_date = future_date.strftime("%A, %d %B %Y")
20
21     print("Tanggal {} hari dari sekarang akan menjadi: {}".format(num_days, formatted_date))
22
23
24
25
26 if __name__ == "__main__":
27     try:
28         days = int(input("Masukkan jumlah hari dari sekarang: "))
29         print_future_date(days)
30     except ValueError:
31         print("Silakan masukkan jumlah hari yang valid.")
```

Hasil perkalian dari 1 hingga 10 adalah: 3628800
PS C:\pemrogintegrative\uts_nada> & C:/Users/Asus/AppData/Local/Programs/Python/Python38/python.exe c:/pemrogintegrative/uts_nada/soal3.py
Masukkan jumlah hari dari sekarang: 7
Tanggal 7 hari dari sekarang akan menjadi: Senin, 01 April 2024
PS C:\pemrogintegrative\uts_nada>

4. Soal 4

```
1 class BMI:
2     def __init__(self, berat, tinggi):
3         self.berat = berat # dalam satuan pound
4         self.tinggi = tinggi # dalam satuan kaki
5
6     @property
7     def bmi_value(self):
8         tinggi_meter = self.tinggi * 0.3048 # konversi tinggi dari kaki ke meter
9         berat_kg = self.berat * 0.453592 # konversi berat dari pound ke kg
10        bmi = berat_kg / (tinggi_meter ** 2) # hitung BMI
11        return bmi
12
13    def __eq__(self, other):
14        if isinstance(other, BMI):
15            # Bandingkan berat dan tinggi dari dua objek BMI
16            return self.berat == other.berat and self.tinggi == other.tinggi
17        return False
18
19    # Contoh penggunaan
```

BMI 2: 24.412118268680974
BMI 1 dan BMI 2 memiliki berat dan/atau tinggi yang berbeda.
PS C:\pemrogintegrative\uts_nada> & C:/Users/Asus/AppData/Local/Programs/Python/Python38/python.exe c:/pemrogintegrative/uts_nada/soal4.py
BMI 1: 24.210365225138162
BMI 2: 24.412118268680974
BMI 1 dan BMI 2 memiliki berat dan/atau tinggi yang berbeda.
PS C:\pemrogintegrative\uts_nada>

```
17 |         return False
18 |
19 | # Contoh penggunaan
20 | bmi1 = BMI(150, 5.5) # berat: 150 pound, tinggi: 5.5 kaki
21 | bmi2 = BMI(180, 6.0) # berat: 180 pound, tinggi: 6.0 kaki
22 |
23 | # Cetak nilai BMI
24 | print("BMI 1:", bmi1.bmi_value)
25 | print("BMI 2:", bmi2.bmi_value)
26 |
27 | # Bandingkan dua objek BMI
28 | if bmi1 == bmi2:
29 |     print("BMI 1 dan BMI 2 memiliki berat dan tinggi yang sama.")
30 | else:
31 |     print("BMI 1 dan BMI 2 memiliki berat dan/atau tinggi yang berbeda.")
32 |
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

BMI 2: 24.412118268680974
BMI 1 dan BMI 2 memiliki berat dan/atau tinggi yang berbeda.
PS C:\pemrogintegrative\uts_nada> & C:/Users/Asus/AppData/Local/Programs/Python/Python38/python.exe c:/pemrogintegrative/uts_nada/soal4.py
BMI 1: 24.210365225138162
BMI 2: 24.412118268680974
BMI 1 dan BMI 2 memiliki berat dan/atau tinggi yang berbeda.
PS C:\pemrogintegrative\uts_nada> |

5. Soal 5

```
7 |
8 | def baca_dan_cetak(file_name):
9 |     try:
10 |         # Membuka file untuk membaca
11 |         with open(file_name, 'r') as file:
12 |             # Membaca baris dari file
13 |             lines = file.readlines()
14 |
15 |             # Menginisialisasi daftar untuk menyimpan angka-angka
16 |             angka_list = []
17 |
18 |             # Mengambil bilangan bulat dari setiap baris dan menambahkannya ke daftar
19 |             for line in lines:
20 |                 try:
21 |                     angka = int(line.strip()) # Menghilangkan whitespace dan mengonversi ke integer
22 |                     angka_list.append(angka)
23 |                 except ValueError:
24 |                     print(f"Ignoring non-integer value in file: {line}")
25 |
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

da/soal4.py
BMI 1: 24.210365225138162
BMI 2: 24.412118268680974
BMI 1 dan BMI 2 memiliki berat dan/atau tinggi yang berbeda.
PS C:\pemrogintegrative\uts_nada> & C:/Users/Asus/AppData/Local/Programs/Python/Python38/python.exe c:/pemrogintegrative/uts_nada/soal5.py
Jumlah angka dalam file: 256,33,47,26,275,89
PS C:\pemrogintegrative\uts_nada> |

```
8 def baca_dan_cetak(file_name):
24     print(f"Ignoring non-integer value in file: {line}")
25
26     # Mengubah daftar angka menjadi string dengan pemisah koma setiap tiga angka
27     angka_str = ','.join('{}'.format(angka) for angka in angka_list)
28
29     # Mencetak daftar angka dengan pemisah koma
30     print("Jumlah angka dalam file:", angka_str)
31 except FileNotFoundError:
32     print(f"File {file_name} tidak ditemukan.")
33
34 def main():
35     file_name = "input.txt" # Nama file yang akan dibaca
36
37     # Memanggil fungsi untuk membaca dan mencetak angka-angka dalam file
38     baca_dan_cetak(file_name)
39
40 if __name__ == "__main__":
41     main()
```

da/soal4.py
BMI 1: 24.210365225138162
BMI 2: 24.412118268680974
BMI 1 dan BMI 2 memiliki berat dan/atau tinggi yang berbeda.
PS C:\pemrogintegrative\uts_nada> & C:/Users/Asus/AppData/Local/Programs/Python/Python38/python.exe c:/pemrogintegrative/uts_nada/soal5.py
Jumlah angka dalam file: 256,33,47,26,275,89
PS C:\pemrogintegrative\uts_nada>

```
1 256
2 33
3 47
4 26
5 275
6 89
7
```

da/soal4.py
BMI 1: 24.210365225138162
BMI 2: 24.412118268680974
BMI 1 dan BMI 2 memiliki berat dan/atau tinggi yang berbeda.
PS C:\pemrogintegrative\uts_nada> & C:/Users/Asus/AppData/Local/Programs/Python/Python38/python.exe c:/pemrogintegrative/uts_nada/soal5.py
Jumlah angka dalam file: 256,33,47,26,275,89
PS C:\pemrogintegrative\uts_nada>

6. Screenshot Jawaban dari Soal ke-2.

7. Screenshot Jawaban dari Soal ke-3.

8. Screenshot Jawaban dari Soal ke-4.

9. Screenshot Jawaban dari Soal ke-5.