

**LEMBAR JAWABAN UJIAN AKHIR SEMESTER  
PROGRAM STUDI ILMU KOMPUTER  
UNIVERSITAS DJUANDA**



NIM : I.2210349

Nama Mahasiswa : Muhammad Angga Nugraha

Semester : 4

NILAI :

Jawaban :

1. Fungsi dalam python adalah blok kode yang dirancang untuk melakukan tugas tertentu dan hanya berjalan saat dipanggil. Fungsi dalam Python didefinisikan menggunakan kata kunci "def", diikuti oleh nama fungsi dan tanda kurung "()" yang mungkin berisi parameter.

Contohnya :

```
1 # Mendefinisikan fungsi
2 def tambah(angka1, angka2):
3     # Fungsi ini menambahkan dua angka dan mengembalikan hasilnya.
4     hasil = angka1 + angka2
5     return hasil
6
7 # Memanggil fungsi
8 hasil_penjumlahan = tambah(5, 3)
9 print(f"Hasil penjumlahan adalah {hasil_penjumlahan}")
```

Penjelasan:

- A. Mendefinisikan Fungsi: Fungsi tambah didefinisikan dengan dua parameter angka1 dan angka2. Fungsi ini menambahkan kedua angka tersebut dan mengembalikan hasilnya.
  - B. Memanggil Fungsi: Fungsi tambah dipanggil dengan argumen 5 dan 3, dan hasilnya disimpan dalam variabel hasil\_penjumlahan. Hasil ini kemudian dicetak ke layar.
2. Penjelasan program 1:
    - code ini dimulai dengan mendefinisikan contacts sebagai list kosong.
    - setelah itu Kemudian Membuka File CSV, Menggunakan with open(csv\_filename, mode="r") as csv\_file, program membuka file CSV dalam mode baca ("r"). csv\_filename adalah variabel yang menyimpan nama file CSV yang akan dibaca.
    - Membaca File CSV, csv\_reader = csv.DictReader(csv\_file) membuat objek csv\_reader yang digunakan untuk membaca baris-baris file CSV sebagai dictionary, di mana kunci adalah nama kolom dan nilai adalah data dari baris tersebut.
    - Disini juga terdapat Loop for row in csv\_reader iterasi melalui setiap baris dalam file CSV, dan setiap baris (yang berupa dictionary) ditambahkan ke list contacts menggunakan contacts.append(row) tentunya untuk mengisi list.Fungsi Program ini membaca data dari sebuah file CSV dan menyimpannya ke dalam list contacts, di mana setiap elemen dalam list adalah sebuah dictionary yang mewakili satu baris dari file CSV tersebut. Ini memungkinkan data CSV diakses dengan mudah menggunakan nama kolom sebagai kunci. Jadi intinya Program ini membaca data dari file CSV dan menyimpannya ke dalam list "contacts", di mana setiap elemen dalam list adalah sebuah dictionary yang mewakili satu baris dari file CSV tersebut.

Penjelasan Program 2 :

Program ini akan menjalankan fungsi `show_menu` dalam loop tak terbatas selama code dieksekusi sebagai code utama. Kondisi `"if _name_ == "_main_":` memastikan bahwa code di dalamnya hanya dieksekusi jika file ini dijalankan sebagai code utama, dan tidak ketika diimpor sebagai modul dalam code lain. fungsi `while True`, memulai loop tak terbatas yang akan terus berjalan hingga dihentikan secara eksplisit. Di dalam loop ini, "fungsi `show_menu()`" dipanggil berulang kali, yang diasumsikan menampilkan menu kepada pengguna dan menunggu input mereka. Ini adalah struktur umum dalam aplikasi berbasis menu yang beroperasi di Command Line Interface (CLI), di mana pengguna dapat berinteraksi dengan program melalui menu yang ditampilkan secara terus-menerus. Jadi Program ini menjalankan fungsi `show_menu` secara terus-menerus dalam loop tak terbatas selama skrip dijalankan sebagai skrip utama, biasanya digunakan untuk menampilkan menu interaktif di CLI. Untuk menampilkan data dari database menggunakan Python, Anda perlu menggunakan perintah SQL melalui perintah database yang sesuai. Berikut adalah beberapa perintah umum yang digunakan untuk menampilkan data dari database dengan menggunakan `mysql-connector-python` sebagai contoh.

### 3. Perintah yang digunakan:

`connect()`: Membuat koneksi ke database.

`cursor()`: Membuat objek cursor yang digunakan untuk mengeksekusi perintah SQL.

`execute(sql_query)`: Menjalankan perintah SQL.

`fetchall()`: Mengambil semua baris hasil dari perintah SQL yang dijalankan.

`fetchone()`: Mengambil satu baris hasil dari perintah SQL yang dijalankan.

`fetchmany(size)`: Mengambil sejumlah baris hasil dari perintah SQL yang dijalankan.

`commit()`: Menyimpan semua perubahan yang dilakukan selama sesi koneksi (relevan untuk operasi INSERT, UPDATE, DELETE).

`close()`: Menutup koneksi atau cursor untuk melepaskan sumber daya.

```
1  import mysql.connector
2
3  # Menghubungkan ke database MySQL
4  conn = mysql.connector.connect(
5      host="localhost",
6      user="root",
7      password="password",
8      database="example_db"
9  )
10
11 # Membuat cursor
12 cursor = conn.cursor()
13
14 # Menjalankan perintah SQL untuk menampilkan data
15 cursor.execute("SELECT * FROM users")
16
17 # Mengambil semua baris hasil query
18 rows = cursor.fetchall()
19
20 # Menampilkan data
21 for row in rows:
22     print(row)
23
24 # Menutup cursor dan koneksi
25 cursor.close()
26 conn.close()
```

4.

```
import mysql.connector

# Menghubungkan ke server MySQL
conn = mysql.connector.connect(
    host="localhost",
    user="root",
    password=""
)

# Membuat cursor
cursor = conn.cursor()

# Membuat database
cursor.execute("CREATE DATABASE IF NOT EXISTS database_uas")
cursor.execute("USE database_uas")

# Membuat tabel pertama
cursor.execute("""
    CREATE TABLE IF NOT EXISTS tabel1 (
        id INT AUTO_INCREMENT PRIMARY KEY,
        nama VARCHAR(255),
        usia INT
    )
""")

# Membuat tabel kedua
cursor.execute("""
    CREATE TABLE IF NOT EXISTS tabel2 (
        id INT AUTO_INCREMENT PRIMARY KEY,
        nama VARCHAR(255),
        usia INT
    )
""")

# Mengisi data ke tabel1
data_tabel1 = [
    ("Alice", 25),
    ("Bob", 30),
    ("Charlie", 35),
    ("David", 40)
]
cursor.executemany("INSERT INTO tabel1 (nama, usia) VALUES (%s, %s)", data_tabel1)

# Mengisi data ke tabel2
data_tabel2 = [
    ("Abbas", 20),
    ("Rindan", 21),
    ("Ryan", 23),
    ("Mulyadi", 22)
]
cursor.executemany("INSERT INTO tabel2 (nama, usia) VALUES (%s, %s)", data_tabel2)

# Menyimpan perubahan
conn.commit()

# Menampilkan data dari tabel1
cursor.execute("SELECT * FROM tabel1")
rows = cursor.fetchall()
print("Data dari tabel1:")
for row in rows:
    print(row)

# Menampilkan data dari tabel2
cursor.execute("SELECT * FROM tabel2")
rows = cursor.fetchall()
print("\nData dari tabel2:")
for row in rows:
    print(row)

# Menutup cursor dan koneksi
cursor.close()
conn.close()
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
PS C:\PERKULIAHAN\tugas pak raden\UAS> python -u "c:\PERKULIAHAN\tugas pak raden\UAS\UAS.py"
Data dari tabel1:
(1, 'Alice', 25)
(2, 'Bob', 30)
(3, 'Charlie', 35)
(4, 'David', 40)
(5, 'Alice', 25)
(6, 'Bob', 30)
(7, 'Charlie', 35)
(8, 'David', 40)

Data dari tabel2:
(1, 'Abbas', 20)
(2, 'Rindan', 21)
(3, 'Ryan', 23)
(4, 'Mulyadi', 22)
(5, 'Abbas', 20)
(6, 'Rindan', 21)
(7, 'Ryan', 23)
(8, 'Mulyadi', 22)
PS C:\PERKULIAHAN\tugas pak raden\UAS>
```