**CÂU HỎI TỔNG HỢP**

1. Phân biệt toán tử định dạng chuỗi và hàm định dạng chuỗi có sẵn trong gói thư viện chuẩn Python? Cho năm ví dụ minh họa tương ứng?

- Toán tử định dạng chuỗi được sử dụng để chèn các giá trị vào chuỗi định dạng. Đây là cách cũ hơn và ít linh hoạt hơn so với các phương pháp mới hơn.

# Ví dụ 1

name = "Alice"

age = 30

formatted\_string = "My name is %s and I am %d years old." % (name, age)

print(formatted\_string)

# Ví dụ 2

number = 3.14159

formatted\_string = "The value of pi is approximately %.2f." % number

print(formatted\_string)

# Ví dụ 3

data = ("John", "Doe", 53.44)

formatted\_string = "Hello %s %s. Your current balance is $%.2f." % data

print(formatted\_string)

# Ví dụ 4

hex\_value = 255

formatted\_string = "The hex value of %d is %x." % (hex\_value, hex\_value)

print(formatted\_string)

# Ví dụ 5

percentage = 75

formatted\_string = "You have completed %d%% of the task." % percentage

print(formatted\_string)

- Hàm định dạng chuỗi cung cấp một cách linh hoạt hơn để định dạng chuỗi, cho phép sử dụng các chỉ số, vị trí và tên biến.

# Ví dụ 1

name = "Alice"

age = 30

formatted\_string = "My name is {} and I am {} years old.".format(name, age)

print(formatted\_string)

# Ví dụ 2

number = 3.14159

formatted\_string = "The value of pi is approximately {:.2f}.".format(number)

print(formatted\_string)

# Ví dụ 3

data = {"first\_name": "John", "last\_name": "Doe", "balance": 53.44}

formatted\_string = "Hello {first\_name} {last\_name}. Your current balance is ${balance:.2f}.".format(\*\*data)

print(formatted\_string)

# Ví dụ 4

hex\_value = 255

formatted\_string = "The hex value of {0} is {0:x}.".format(hex\_value)

print(formatted\_string)

# Ví dụ 5

percentage = 75

formatted\_string = "You have completed {}% of the task.".format(percentage)

print(formatted\_string)

2. Viết chương trình xuất ra số ngẫu nhiên trong một đoạn bất kỳ bất cho trước?

import random

def generate\_random\_number(start, end):

return random.randint(start, end)

start = int(input("Nhập giá trị bắt đầu: "))

end = int(input("Nhập giá trị kết thúc: "))

random\_number = generate\_random\_number(start, end)

print(f"Số ngẫu nhiên trong đoạn [{start}, {end}] là: {random\_number}")

3. Khác biệt cơ bản giữa list và tuple?

- Tuple Immutable (giá trị bất biến không thể thay đổi).

- So với list, bộ nhớ được sử dụng cho tuple nhỏ hơn. Do đó tải trọng lên máy tính của tuple cũng nhỏ hơn.

- Nhìn chung, tốc độ xử lý của tuple nhanh hơn so với list.

4. Ứng dụng kiểu dữ liệu tuple trong thực tế?

- Lưu trữ dữ liệu không thay đổi

- Trả về nhiều giá trị từ hàm

- Sử dụng làm khóa trong từ điển

- Lưu trữ các bản ghi dữ liệu

- Tối ưu hóa hiệu suất