**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?
   1. **Toán tử định dạng chuỗi**

Toán tử % được sử dụng để chèn các giá trị vào chuỗi. Đây là cách định dạng chuỗi truyền thống trong Python.

Vd 1:

name = "Alice"

age = 30

print("My name is %s and I am %d years old." % (name, age))

**Vd2**:

pi = 3.14159

print("Value of Pi: %.2f" % pi)

Vd3:

balance = -500

print("Your account balance is %+d dollars." % balance)

Vd4:

language = "Python"

print("I love %-10s programming." % language)

Vd5:

number = 10

print("Binary representation: %b" % number)

* 1. **Hàm định dạng chuỗi**

Phương thức str.format() cung cấp một cách linh hoạt hơn để định dạng chuỗi. Bạn có thể sử dụng các dấu ngoặc nhọn {} để chèn các giá trị vào chuỗi.

Vd 1:

name = "Bob"

age = 25

print("My name is {} and I am {} years old.".format(name, age))

Vd2:

pi = 3.14159

print("Value of Pi: {:.2f}".format(pi))

Vd3:

first = "apple"

second = "banana"

print("First: {0}, Second: {1}".format(first, second))

Vd4:

print("My name is {name} and I am {age} years old.".format(name="Charlie", age=28))

Vd5:

number = 10

print("Binary representation: {:b}".format(number))

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

trước?

import random

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

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

random\_number = random.randint(start, end)

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

3. Khác biệt cơ bản giữa list và tuple?

-Tuple là 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.

+điểm khác nhau lớn nhất và quan trọng nhất giữa List và Tuple trong Python đó chính là, List có thuộc tính Mutable (giá trị có thể thay đổi) còn Tuple lại có thuộc tính ngược lại là Immutable (giá trị bất biến không có khả năng thay đổi)

+do tuple có thuộc tính Immutable (giá trị bất biến không có khả năng thay đổi), do đó tuple thích hợp để lưu trữ những dữ liệu mà một khi đã tạo ra thì rất ít khi thay đổi chúng như là thời khắc, địa chỉ nhà, địa chỉ ip v.v..

+khác với List với thuộc tính Mutable (giá trị có thể thay đổi) có tới 11 phương thức để xử lý, thì Tuple với thuộc tính ngược lại là Immutable (giá trị bất biến không có khả năng thay đổi) chỉ có hai phương thức xử lý không làm thay đổi giá trị của tuple là count() và index() mà thôi:

+Bởi vậy, hãy **sử dụng tuple trong python** nếu bạn muốn lưu trữ các dữ liệu mà bạn hiếm khi thay đổi chúng.  
Và ngược lại, hãy **sử dụng list** nếu bạn lưu trữ các dữ liệu mà bạn có nhu cầu thay đổi chúng sau khi tạo ra.

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

Trình xử lý các tuple là nhanh hơn các List. Làm cho dữ liệu an toàn hơn bởi vì tuple là không thay đổi (immutable) và vì thế nó không thể bị thay đổi. Các tuple được sử dụng để định dạng String.

Kiểu dữ liệu tuple trong Python rất hữu ích khi bạn cần lưu trữ một tập hợp giá trị mà không cần thay đổi nó sau khi đã tạo ra. Dưới đây là một số ứng dụng thực tế của tuple:

1. Lưu trữ các giá trị cố định
2. Sử dụng làm khóa trong từ điển
3. Trả về giá trị từ một hàm
4. **Đóng gói dữ liệu với các chỉ mục có ý nghĩa**
5. **Hiệu suất và bảo mật.**