| **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 dạnh chuỗi* (%) là phương pháp định dạnh chuỗi, tương tự như trong ngôn ngữ lập trình C. Định dạnh này có thể sử dụng các ký tự đặc biệt để chèn các giá trị vào chuổi. Cú pháp: formatted\_string = "format string % values"  -Một số các toán tử cơ bản:  +%s: Giá trị của phương thức\_str\_ của đối tượng đó  +%r :Giá trị của phương thức\_repr\_ của đối tượng đó  +%d: Giá trị của một số - Nếu là số thực thì sẽ chỉ lấy phần nguyên (chuyển sang số nguyên  +% <số chữ số phần thập phân> f : Giá trị của một số - Nếu là số sẽ được chuyển sang số thực  -Ví dụ:  1.  name = "nhuy"  age = 18  formatted\_string = "Name: %s, Age: %d" % (name, age)  print(formatted\_string)  2.  user = "Jane"  balance = 1234.567  formatted\_string = "User {0} has a balance of ${1:.2f}".format(user, balance)  print(formatted\_string) # Output: User Jane has a balance of $1234.57  3.  item = "apple"  quantity = 3  price = 1.99  formatted\_string = "Item: %-10s, Quantity: %d, Price: $%.2f" % (item, quantity, price)  print(formatted\_string)  -*Hàm định dạng chuỗi* :Phương thức này cung cấp một cách linh hoạt và hỗ trợ nhiều chức năng hơn. Hàm format() chuỗi Python đã được giới thiệu để xử lý định dạng chuỗi phức tạp hiệu quả hơn. Cú pháp:  -Ví dụ: ten\_nguoi\_dung = "Cao Thi Nhu Y"  cau\_chao = "Chao ban {}!"  cau\_chao\_a = cau\_chao.format(ten\_nguoi\_dung)  print(cau\_chao\_a)  **2. Viết chương trình xuất ra số ngẫu nhiên trong một đoạn bất kỳ bất cho**  **trước?**  import random  print(random.randrange(100))  **3. Khác biệt cơ bản giữa list và tuple?**  Dữ liệu thuộc kiểu list có thể thay đổi được, dữ liệu thuộc kiểu tuple không thể thay đổi được.  **4. Ứng dụng kiểu dữ liệu tuple trong thực tế?** |
| --- |

Tuple giống như list nhưng là bất biến (immutable). Một khi tuple được tạo, bạn không thể thay đổi các giá trị bên trong nó

-Ví dụ:

my\_tuple = (1, "hi", 3.4)

print(type(my\_tuple))